Universidad Carlos III de Madrid

Departamento de Informática

Tesis Doctoral

Evaluation in Audio Music Similarity

Autor:

Julián Urbano

Director:

Juan Lloréns

Leganés, Octubre de 2013

Tesis Doctoral

Evaluation in Audio Music Similarity

Autor: Julián Urbano

Director: Juan Lloréns

Firma del Tribunal Calificador:

Firma

Presidente: Manuel Velasco

Vocal: Peter Knees

Secretario: Bob L. Sturm

Calificación:

Leganés, de de .

A Pepita

Acknowledgments

There are two people to "blame" for me ending up in the academic world: Alejandro Calderón and Félix García. They showed it to me around 2002, and now I can't think of myself doing something different. Thank you. Two people are somehow responsible for me liking this so much: Bill Frakes and Gabriella Belli. From them I learned the most important thing a researcher can learn: how to do research. That's something I can not thank enough. Before knowing them I tried to stay as far away from statistics as possible, and here I am now, talking about distributions. Thank you.

Lots of people go successfully through their PhD, but I bet not many can say they did it on Evaluation in Music Information Retrieval while working in a Software Engineering group. For letting me stick to this topic, I have to thank my advisor Juan Lloréns. In that group I found many people over the years, some of which I'll remember now that I leave. Special thanks go to Jorge Morato for all the support and all the random, yet interesting chats; I know you don't think so, but I learned a lot from them. I also have to thank Diego Martín for way too many things to list here, but mainly for being there in the good and the bad moments; I found a great guy, but better friend yet. I can't forget either all the good times spent with the folks at the lab, especially Álvaro, Miguel, Julio and Luis María.

These last years I've had the pleasure to meet some great colleagues out there. I am particularly indebted to Markus Schedl for the work together and the opportunities, and to Justin Salamon for the good time spent together and everything thereafter. I have to thank you both for forcing me to use LaTeX; you were right! It might go unnoticed, but this dissertation wouldn't have been possible without the availability and willingness of Stephen Downie and the IMIRSEL group. Needless to say, it wouldn't have been possible either without the many researchers that participated in MIREX all these years. But if there is something that really made this possible, that's the financial support from all those hard working people who, unknowingly, made my salary possible with their taxes.

It doesn't really matter where you work or who you work with, what makes you want to keep going is what you have out of work. I know I haven't been the best son, brother, uncle or friend these last years. These 172 pages here, and many others elsewhere, are the reason why. Thank you for being there even when I wasn't, and especially now that I am back.

And last, but certainly not least, there is Mónica. I can't thank enough all your support and confidence all along, and the right words at the right moment. You did help me with all the late reviews, the tiny questions that turned into endless discussions, the time spent at the whiteboard, or simply stating the obvious when I couldn't see it, no matter how hard I looked! Thank you for the last 1602 days, and all days to come.

Abstract

Audio Music Similarity is a task within Music Information Retrieval that deals with systems that retrieve songs musically similar to a query song according to their audio content. Evaluation experiments are the main scientific tool in Information Retrieval to determine what systems work better and advance the state of the art accordingly. It is therefore essential that the conclusions drawn from these experiments are both valid and reliable, and that we can reach them at a low cost. This dissertation studies these three aspects of evaluation experiments for the particular case of Audio Music Similarity, with the general goal of improving how these systems are evaluated. The traditional paradigm for Information Retrieval evaluation based on test collections is approached as an statistical estimator of certain probability distributions that characterize how users employ systems. In terms of validity, we study how well the measured system distributions correspond to the target user distributions, and how this correspondence affects the conclusions we draw from an experiment. In terms of reliability, we study the optimal characteristics of test collections and statistical procedures, and in terms of efficiency we study models and methods to greatly reduce the cost of running an evaluation experiment.

Contents

Contents i

List of Figures iii

List of Tables vii

List of Symbols xi

1 Introduction 1
- 1.1 Information Retrieval . . . 1
- 1.2 Information Retrieval Evaluation . . . 2
- 1.3 Importance and Impact of IR Evaluation Research . . . 4
- 1.4 Audio Music Similarity . . . 5
- 1.5 Motivation . . . 6
- 1.6 Organization . . . 7

2 Information Retrieval Evaluation 9
- 2.1 The Cranfield Paradigm . . . 9
- 2.2 Validity . . . 13
- 2.3 Reliability . . . 17
- 2.4 Efficiency . . . 18
- 2.5 Effectiveness Measures . . . 19

3 System Effectiveness and User Satisfaction 25
- 3.1 Effectiveness Measures and Relevance Scales . . . 25
- 3.2 Experimental Design . . . 30
- 3.3 Results . . . 33
- 3.4 Considering Priors . . . 40
- 3.5 Discussion . . . 44
- 3.6 Summary . . . 45

4 Modeling Distributions 47
- 4.1 Mean Probability of User Satisfaction . . . 47
- 4.2 Distribution of the Probability of User Satisfaction . . . 52
- 4.3 Probability of Success . . . 57

4.4 Discussion . . . 59
4.5 Summary . . . 60

5 Optimality of Statistical Significance Tests 63
5.1 Reliable System Comparisons . . . 63
5.2 Effectiveness Measures and Relevance Scales . . . 67
5.3 Data and Methods . . . 69
5.4 Results . . . 71
5.5 Discussion . . . 76
5.6 Statistical Significance and Practical Significance . . . 78
5.7 Summary . . . 82

6 Test Collection Size 85
6.1 Generalizability Theory . . . 85
6.2 The Effect of Query Set Size . . . 88
6.3 The Effect of Evaluation Cutoff . . . 93
6.4 The Effect of Assessor Set Size . . . 94
6.5 Discussion . . . 97
6.6 Summary . . . 98

7 Learning Relevance Distributions 101
7.1 Probabilistic Evaluation . . . 101
7.2 Estimation of Relevance Judgments . . . 102
7.3 Results . . . 105
7.4 Discussion . . . 107
7.5 Summary . . . 108

8 Low-Cost Evaluation 109
8.1 Probabilistic Effectiveness Measures . . . 109
8.2 Evaluation Without Relevance Judgments . . . 114
8.3 Estimating Differences in Effectiveness . . . 117
8.4 Estimating Absolute Effectiveness . . . 121
8.5 Discussion . . . 125
8.6 Summary . . . 126

9 Conclusions and Future Work 129
9.1 Conclusions . . . 129
9.2 Future Work . . . 132

A Models to Estimate Relevance 135

List of Figures

1 Introduction **1**

1.1 The IR Research and Development cycle. 2

1.2 Timeline of Evaluation in Text IR (top) and Music IR (bottom). 3

1.3 Importance (left) of publications related to IR Evaluation in SIGIR and ISMIR proceedings; and their impact (right) along with TREC overview papers. 5

2 Information Retrieval Evaluation **9**

2.1 Validity and Reliability. Adapted from [Trochim and Donnelly 2007]. 12

2.2 Accuracy and Precision. θ is the true value and E$[\hat{\theta}]$ is the estimated value. . 13

3 System Effectiveness and User Satisfaction **25**

3.1 Task template used in the experiment. 31

3.2 Log-scaled distributions of absolute λ scores in all 2,025 examples with nonpreferences. 34

3.3 Probability of user satisfaction given a λ score in all 2,025 examples with nonpreferences. 35

3.4 Probability of positive user preference given a $\Delta\lambda$ score in all 4,115 examples. 39

3.5 Probability of negative user preference given a $\Delta\lambda$ score in all 4,115 examples. 41

3.6 Cumulative distribution functions of prior $\Delta\lambda$ scores. Points mark $\Delta\lambda$ such that $P(Agr = 1|\Delta\Lambda \geq \Delta\lambda) \geq 0.5$ (lower is better). 43

4 Modeling Distributions **47**

4.1 Estimated $\widehat{sat}(\lambda)$ mappings fitted from Figure 3.3. 50

4.2 Distributions of CG_l@5 and corresponding $\hat{P}(Sat)$ for system CL1 in MIREX 2009. 52

4.3 True distribution, histogram of a sample, and sampling distribution of the mean. 53

4.4 Examples of distribution fits for RBP_l@5 with Broad judgments for systems ME and BK2 from MIREX 2007. 54

4.5 Average $\hat{\omega}$ statistics of the fits provided by the Normal, Truncated Normal, Beta and Empirical distributions for different query set sizes. 56

4.6 Distributions of DCG_e@5 scores with $n_{\mathcal{L}} = 5$ (top) and corresponding distributions of $\hat{P}(Sat)$ (bottom) for systems ANO and GT in MIREX 2009. . . 60

4.7 Estimated difference in $P(Sat)$ (left) and in $P(Succ)$ (right) as a function of $\overline{\Delta\lambda}$ for all 40 $(\Lambda, \mathcal{L})$ combinations and all pairs of systems from MIREX 2007, 2009, 2010 and 2011. 61

5 Optimality of Statistical Significance Tests **63**

5.1 Left: non-significance rates (lower is better), success rates (higher is better), and lack of power rates (lower is better). Right: minor conflict rates (lower is better), major conflict rates (lower is better), and global conflict rates ($rate = \alpha$ is better). All rates per significance test. 72

5.2 Non-significance rates (left, lower is better) and success rates (right, higher is better) for all measures and scales. All rates per measure and scale. 75

5.3 Minor conflict rates (left, lower is better) and major conflict rates (right, lower is better) for all measures and scales. All rates per measure and scale. . 76

6 Test Collection Size **85**

6.1 Estimated Eρ^2 scores as a function of query set size n'_q, for all measures and scales of interest and for the MIREX 2007, 2009, 2010 and 2011 AMS test collections (higher is better). 90

6.2 Estimated Φ scores as a function of query set size n'_q, for all measures and scales of interest and for the MIREX 2007, 2009, 2010 and 2011 AMS test collections (higher is better). 92

6.3 Estimated Eρ^2 (top) and Φ (bottom) scores in MIREX 2012 as a function of the query set size n'_q and evaluation cutoff k, for all measures and scales of interest (higher is better). 94

6.4 Estimated Eρ^2 (top) and Φ (bottom) scores in MIREX 2012 as a function of the number of relevance judgments n'_r and evaluation cutoff k, for all measures and scales of interest (higher is better). The vertical solid line marks the usual number of judgments that would have been made in a traditional MIREX setting. 95

6.5 Estimated Eρ^2 (left) and Φ (right) scores in MIREX 2006 as a function of the number of assessors, and as a function of the number of queries n'_q (top) and the number of relevance judgments n'_r (bottom) for RBP_l@5 with Fine judgments (higher is better). 97

7 Learning Relevance Distributions **101**

7.1 Distributions of relevance judgments made in MIREX 2007, 2009, 2010 and 2011. 103

8 Low-Cost Evaluation **109**

8.1 Confidence in the ranking of systems in MIREX 2007, 2009, 2010 and 2011 as the number of judgments increases. 118

8.2 Accuracy of the estimated ranking of systems in MIREX 2007, 2009, 2010 and 2011 as the number of judgments increases. 118

8.3 Accuracy of the absolute effectiveness estimates in MIREX 2007, 2009, 2010 and 2011 as the number of judgments increases. 121

8.4 Estimated vs. actual absolute effectiveness scores in MIREX 2007, 2009, 2010 and 2011 when judging documents until expected error is ±0.05 with an uncorrected (left) or corrected (right) stopping condition. 123
8.5 Rooted variance of estimates needed for absolute errors to be at a certain level. 124

9 Conclusions and Future Work **129**

A Models to Estimate Relevance **135**

List of Tables

1 Introduction **1**

1.1 Summary of MIREX AMS editions. In the 2006 edition three different assessors provided annotations for every query-document pair. The task did not run in 2008 6

2 Information Retrieval Evaluation **9**

3 System Effectiveness and User Satisfaction **25**

3.1 All 95 combinations of effectiveness measures and relevance scales studied (marked with x), and equivalent combinations (e.g. GAP@5 is the same as AP@5 with a binary scale). 30

3.2 Bias in $P(Sat|\lambda)$ at the endpoints $\lambda = 0$ and $\lambda = 1$ as per (3.11) (lower is better). Best per measure in bold, best per scale in italics. 37

3.3 Distance between 1 and $P(Agr = 1|\Delta\lambda)$ (lower is better). Best per measure in bold, best per scale in italics. 38

3.4 Distance between 0 and $P(Agr = -1|\Delta\lambda)$ (lower is better). Best per measure in bold, best per scale in italics. 42

3.5 Expected fraction of observations such that $P(Agr = 1) \geq 0.5$ (higher is better). Best per measure in bold, best per scale in italics. 44

4 Modeling Distributions **47**

4.1 All 40 combinations of effectiveness measures and relevance scales studied (marked with x), and equivalent combinations (e.g. GAP@5 is the same as AP@5 with a binary scale). 48

4.2 RMS residuals of $\widehat{sat}$ predictions (lower is better). Best per measure in bold, best per scale in italics. 49

4.3 Fitted parameters of the $\widehat{sat}(\lambda) = a_0 + a_1\lambda + a_2\lambda^2 + a_3\lambda^3$ models. 51

4.4 Average $\hat{\omega}$ statistics of the fits provided by the Normal, Truncated Normal, Beta and Empirical distributions for $n_{\mathcal{Q}_1} = 50$. Best per measure in bold. . . 58

4.5 Average $\hat{\omega}$ statistics of the fits provided by the Normal, Truncated Normal, Beta and Empirical distributions for $n_{\mathcal{Q}_1} = 20$. Best per measure in bold. . . 59

5 Optimality of Statistical Significance Tests **63**

5.1 Statistical hypothesis testing as a binary decision problem. 65

5.2 All 14 combinations of effectiveness measures and relevance scales studied (marked with x), and equivalent combinations (e.g. Q_l@5 is the same as AP@5 with the $\ell_{min} = 40$ scale). 70
5.3 Left: non-significance rates (lower is better), success rates (higher is better), and lack of power rates (lower is better). Right: minor conflict rates (lower is better), major conflict rates (lower is better), and global conflict rates ($rate = \alpha$ is better). All rates per significance test. Best per α in bold. 73
5.4 Non-significance rates (lower is better), success rates (higher is better), lack of power rates (lower is better), minor conflict rates (lower is better), major conflict rates (lower is better), and global conflict rates ($rate = \alpha$ is better) for all measures and scales at $\alpha = 0.05$. All rates per measure and scale. Best per rate in bold face. 74
5.5 RMS error of all five tests with themselves (lower is better). Best per bin in bold. 78

6 Test Collection Size **85**
6.1 Estimated variance components (over total variance) for all measures and scales of interest and for the MIREX 2007, 2009, 2010 and 2011 AMS test collections. Best measure per component, year and scale in bold. 88
6.2 Estimated $E\rho^2$ scores (higher is better) for all measures and scales of interest and for the MIREX 2007, 2009, 2010 and 2011 AMS test collections, along with required number of queries to reach $E\hat{\rho}^2 = 0.95$ (lower is better). Best per year and scale in bold face. 89
6.3 Estimated Φ scores (higher is better) for all measures and scales of interest and for the MIREX 2007, 2009, 2010 and 2011 AMS test collections, along with required number of queries to reach $\hat{\Phi} = 0.95$ (lower is better). Best per year and scale in bold face. 91
6.4 Estimated variance components (over total variance) for all measures and scales of interest in the MIREX 2006 AMS test collection. Best measure per component and scale in bold. 96

7 Learning Relevance Distributions **101**
7.1 Likelihood-ratio Chi-squared statistics of all effects fitted in each model, along with R^2 score, rooted mean squared error between predicted and actual scores, and average variance of estimates for M_{out} (top) and M_{jud} (bottom) models. Models for year Y are fitted excluding all judgments from Y, and tested against those. 106

8 Low-Cost Evaluation **109**
8.1 Confidence and accuracy of the effectiveness estimates when evaluating systems in MIREX 2007, 2009, 2010 and 2011 without relevance judgments. . . 115
8.2 Accuracy vs. confidence in the sign of estimates when evaluating systems in MIREX 2007, 2009, 2010 and 2011 without relevance judgments. 116
8.3 Confidence and accuracy of estimated differences in MIREX 2007, 2009, 2010 and 2011 when judging documents until 95% average confidence. 119

8.4 Accuracy vs. confidence in the sign of estimates in MIREX 2007, 2009, 2010 and 2011 when judging documents until 95% average confidence. 120
8.5 Accuracy of estimated absolute scores in MIREX 2007, 2009, 2010 and 2011 when judging documents until expected error is ±0.05. 122
8.6 Accuracy of estimated absolute scores in MIREX 2007, 2009, 2010 and 2011 when judging documents until expected error is ±0.05 with corrected thresholds on variance as per Figure 8.5. 125

9 Conclusions and Future Work **129**

A Models to Estimate Relevance **135**
A.1 Parameters fitted for the ordinal logistic regression models using all judgments from MIREX 2007, 2009, 2010 and 2011. 136

List of Symbols

In general, uppercase italic letters (e.g. X, Y) denote random variables and lowercase italic letters (e.g. x, y) denote scalars. Lowercase bold letters (e.g. $\boldsymbol{x}$, $\boldsymbol{y}$) denote vectors, and Greek letters (e.g. μ, σ) represent parameters. Calligraphic letters (e.g. $\mathcal{Q}$, $\mathcal{S}$) represent sets, and sans serif letters (e.g. A, B) represent retrieval systems. When necessary though, this notation is explicitly ignored or simplified.

$\mathcal{D}, n_{\mathcal{D}}$	A corpus of documents and its size.
$\mathcal{Q}, n_{\mathcal{Q}}$	A set of queries and its size.
$\mathcal{R}, n_{\mathcal{R}}$	A set of relevance judgments and its size.
$\mathcal{S}, n_{\mathcal{S}}$	A set of retrieval systems and its size.
$\mathcal{H}, n_{\mathcal{H}}$	A set of human assessors and its size.
$\mathcal{L}, n_{\mathcal{L}}$	A set of possible relevance levels and its size; $\mathcal{L} := \{0, 1, \ldots, n_{\mathcal{L}} - 1\}$
$[\mathcal{S}]^2$	The set of all system pairs; $[\mathcal{S}]^2 := \{\mathcal{S}' \subset \mathcal{S} : \lvert\mathcal{S}'\rvert = 2\}$; $\left\lvert[\mathcal{S}]^2\right\rvert = \binom{n_{\mathcal{S}}}{2}$.
r_d	The relevance judgment for document d; $r_d \in \mathcal{L}$.
R_d	Random variable representing the relevance of document d.
$\mathcal{R}^\ell$	The subset of documents whose relevance is ℓ; $\mathcal{R}^\ell = \{d \in \mathcal{D} : r_d = \ell\}$.
Sat	A random variable that equals 1 if a user is satisfied by the system output and 0 if he is not.
Agr	A random variable that equals 1 if a user agrees with a $\Delta\lambda$ score as to which of two systems is better, -1 if he does not agree, and 0 if he can not decide.
$Succ$	A random variable that equals 1 if at least 50% of users are satisfied by the system output and 0 if not; $P(Succ = 1) = 1 - F_{P(Sat)}(0.5)$.
$g(\ell)$	A monotonically increasing function mapping relevance levels onto gain scores; $g : \mathcal{L} \to \mathbb{R}^{\geq 0}$; $g(0) = 0$; $g(\ell_i) \leq g(\ell_{i+1})$.
$d(i)$	A monotonically increasing function to discount the gain at a certain rank; $d : \mathbb{N}^{>0} \to \mathbb{R}^{>0}$; $d(i) \leq d(i+1)$.
A	The ranking of documents by system A; $\mathsf{A} := \{\mathsf{A}_1, \ldots, \mathsf{A}_{n_{\mathcal{D}}}\}$.
A_i	The document retrieved at rank i by system A; $\mathsf{A}_i \in \mathcal{D}$.
A_d^{-1}	The rank at which document d is retrieved by system A; $\mathsf{A}_{\mathsf{A}_d^{-1}} = d$.
I	An ideal ranking of documents; $\mathsf{I} := \left\langle \mathsf{I}_1, \ldots, \mathsf{I}_{n_{\mathcal{D}}} : \forall i : r_{\mathsf{I}_i} \geq r_{\mathsf{I}_{i+1}} \right\rangle$.

Λ	An arbitrary effectiveness measure or a distribution of effectiveness scores.
λ	The effectiveness score according to some arbitrary measure Λ.
$\lambda_{q,\mathsf{A}}$	The effectiveness score of system A for query q according to measure Λ.
$\overline{\lambda}_{\mathcal{Q},\mathsf{A}}$	The average score of system A with query set $\mathcal{Q}$ according to measure Λ.
$\Delta\lambda_{q,\mathsf{AB}}$	The difference between systems A and B for query q according to measure Λ.
$\overline{\Delta\lambda}_{\mathcal{Q},\mathsf{AB}}$	The average difference between systems A and B with query set $\mathcal{Q}$ according to measure Λ.
f_X	The probability density function of random variable X.
F_X	The cumulative distribution function of random variable X.
Q_X	The quantile or inverse cumulative distribution function of random variable X.
$\mathrm{E}[X]$	The expectation of random variable X.
$\mathrm{Var}[X]$	The variance of random variable X.
$\hat{\theta}$	An estimate of θ.
$\mathbb{1}(e)$	Alternative notation for the Iverson Bracket; $\mathbb{1}(e) = 1 \Leftrightarrow e$ is true.
$\mathcal{A}^{\succ\alpha}$	$\{a \in \mathcal{A} : \mathbb{1}(a \succ \alpha)\}$, where $\succ$ is a binary relation on the set $\mathcal{A}$.

Chapter 1
Introduction

1.1 Information Retrieval

Information Retrieval (IR) is the field concerned with the automatic representation, storage and search of unstructured information [Croft et al. 2009, Buettcher et al. 2010]. In a typical IR scenario, a user has some kind of information need and uses a system that provides her with information that is deemed as relevant or significant to the problem at hand [Baeza-Yates and Ribeiro-Neto 2011, Manning et al. 2008].

Traditionally, these have been activities in which only a few people engaged, such as librarians and professional searchers. But technological developments over the last two decades have made traditional cataloging techniques impractical to cope with the vast amount of information readily available through communication networks, digital libraries, etc. On the other hand, the increasing availability of large computing and storage capacity allowed for a turn in how information is searched and accessed, to the point that these tasks are nowadays ubiquitous and carried out in an automatic fashion with the aid of computers.

Research on IR dates as far back as the 1960s, though the first computer-based search systems go back further to the late 1940s [Sanderson and Croft 2012]. Most IR research has focused on textual information, but other types of information have been gradually studied in the last two decades, such as video, image, audio or music. Information Retrieval systems are based on models that define how documents are represented and how to predict their relevance for some input user query. These models usually work according to some parameters, and they can generally be extended with other techniques to improve their performance. For example, a Text IR system for the Web may be designed to not distinguish between present and past tense, and a Music IR system to recommend songs may be designed to disregard lyrics or focus just on beat patterns. The problem is then to figure out what models, parameters or techniques work better. That is, *what is the best system*?

Most research in Information Retrieval follows a cycle that ultimately leads to the development of better systems thanks to evaluation experiments (see Figure 1.1). First, a research problem is identified and an IR task is defined to evaluate different approaches to solve it. In the Development phase researchers build a new system for that task or adapt a previous one, and to assess how well it performs they then go through an Evaluation phase.

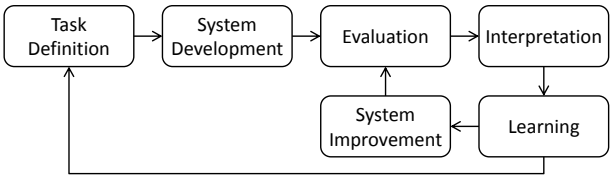

Figure 1.1: The IR Research and Development cycle.

Once experiments are finished the Interpretation of results is carried out, which leads to a phase of Learning why the system worked well or bad and under what circumstances. Finally, with the new knowledge gained researchers go through an Improvement phase to try and make their system better, going back over to the Evaluation phase. In some cases, and especially when the task is new, the first evaluation rounds lead to a re-definition of the task to better capture the real application scenario [Voorhees 2002a].

1.2 Information Retrieval Evaluation

Information Retrieval is thus a highly experimental discipline. Evaluation experiments are the main research tool to scientifically compare IR techniques and advance the state of the art through careful examination and interpretation of their results. Despite being a quite young field of research, Music IR is not an exception. In its early years, the Music IR community mirrored Text IR in terms of evaluation practices, but there has been little research studying whether that mirroring should be fully applied and, when it should not, what alternatives work better. These are very important questions to deal with, because reaching wrong conclusions from evaluation experiments may not only hamper the proper development of our field, but also make us follow completely wrong research directions. Some presentations and discussions at the recent ISMIR (International Society for Music IR) 2012 conference showed the general concern of the Music IR community in this matter, but also the lack of clear views to improve the situation [Peeters et al. 2012].

1.2.1 Evaluation in Text Information Retrieval

Information Retrieval Evaluation has attracted a wealth of research over the years [Harman 2011, Robertson 2008] (see Figure 1.2). The Cranfield 2 experiments [Cleverdon 1991], carried out by Cyril Cleverdon between 1962 and 1966, are often cited as the basis for all modern IR evaluation experiments, and even as the birthplace of the IR field altogether[1] [Harman 2011]. Cleverdon established the so-called *Cranfield paradigm* for IR Evaluation based on test collections (see Chapter 2). From 1966 to 1967, the MEDLARS (Medical Literature Analysis and Retrieval System) study focused on the evaluation of a complete system from a user perspective, taking into consideration the user requirements, response times, required effort, etc. [Lancaster 1968]. The SMART project was directed by Gerard Salton from 1961 until his death in 1995 [Lesk et al. 1997]. One of the results of the project was the development of several test collections, procedures and measures that allowed

[1] He showed that indexing the words in the documents was more effective than indexing terms in a controlled vocabulary.

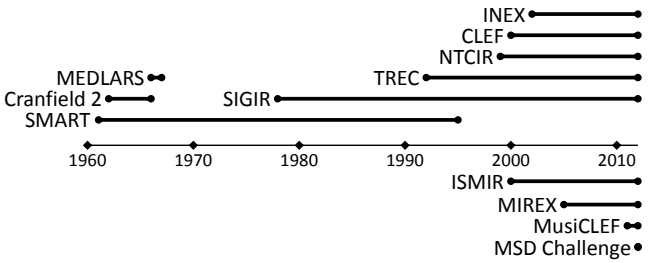

Figure 1.2: Timeline of Evaluation in Text IR (top) and Music IR (bottom).

researchers to perform batch evaluation experiments in a systematic fashion. Meanwhile, the ACM SIGIR conference started in 1978 as the premier venue for Text IR research.

Very successful IR Evaluation forums have followed ever since. TREC[2] (Text REtrieval Conference) started in 1992 to study and provide infrastructure necessary for evaluations based on large-scale test collections [Voorhees and Harman 2005]. NTCIR[3] (National Institute of Informatics–Testbeds and Community for Information access Research) started in 1999 to provide similar infrastructure for Asian languages. CLEF[4] (Conference and Labs of the Evaluation Forum) started in 2000 with an emphasis on multilingual and multimodal information, and INEX[5] (INitiative for the Evaluation of XML retrieval) focuses on structured information since 2002.

1.2.2 Evaluation in Music Information Retrieval

On the Music IR side, the ISMIR conferences started in 2000. Reflecting upon the very long tradition of Text IR Evaluation research, the "ISMIR 2001 resolution on the need to create standardized MIR test collections, tasks, and evaluation metrics for MIR research and development" was drafted during ISMIR 2001, and signed by many members of the Music IR community as a demonstration of the general concern regarding the lack of formal evaluations [Downie 2003b]. A series of three workshops then followed between July 2002 and August 2003, where researches engaged in this long-needed work for evaluation in Music IR [Downie 2003b]. There was some general agreement that evaluation frameworks for Music IR would need to follow the steps of TREC [Voorhees 2002b], although it was clear too that special care had to be taken not to oversimplify the TREC evaluation model [Downie 2002], because Music IR differs greatly from Text IR in many aspects that affect evaluation experiments [Downie 2004].

The general outcome of these workshops and many other meetings was the realization by the Music IR community that a lot of effort and commitment was needed to establish a periodic evaluation forum for Music IR systems. The ISMIR 2004 Audio Description Contest stood up as the first international evaluation project in Music IR [Cano et al. 2006]. Finally, the first edition of the Music Information Retrieval Evaluation eXchange[6]

[2] http://trec.nist.gov
[3] http://research.nii.ac.jp/ntcir/
[4] http://www.clef-initiative.eu
[5] http://inex.mmci.uni-saarland.de
[6] http://www.music-ir.org/mirex/wiki/MIREX_HOME

(MIREX) took place in 2005, organized by IMIRSEL (International Music IR Systems Evaluation Laboratory) [Downie et al. 2010], and ever since it has evaluated over 1,500 Music IR systems for over a dozen different tasks on a yearly basis. More recent evaluation efforts have appeared in the Music IR field, namely the MusiClef[7] campaign in 2011 [Lartillot et al. 2011] (now part of the MediaEval series) and the Million Song Dataset Challenge[8] in 2012 [McFee et al. 2012]. However, these forums cover a much smaller range of tasks than MIREX, usually just one or two, and the MSD Challenge is only scheduled for two years.

1.3 Importance and Impact of IR Evaluation Research

The problem of improving how we evaluate systems is recognized as one of the key areas in Information Retrieval research. In 2002, a workshop gathering world-wide leading IR researchers identified Evaluation as one of the seven grand challenges in the field [Allan and Croft 2003]. This meeting turned into the SWIRL series of workshops, which explore the long-range issues in IR, recognize key challenges and identify past and future research directions. Reflecting upon the history of IR research, the first workshop collected in 2004 a list of 47 recommended readings for IR researchers [Moffat et al. 2005], where as many as 9 (19%) were devoted to analyzing or improving evaluation methods, clearly showing the importance of this topic. The second meeting took place in 2012, and Evaluation was still recognized as one of the six grand challenges in Information Retrieval [Allan et al. 2012]. An updated list of recommended readings included this time 28 (21%) publications related to evaluation. Even the 2012 ACM Computing Classification System[9], which updates the previous 1998 version, reflects the importance of Evaluation by listing it as one of the eight main areas in the IR field.

On the Music IR side, the recent MIReS project (Roadmap for Music Information ReSearch), funded by the 7th Framework Programme of the European Commission, is an international and collective attempt at recognizing the challenges and future directions of the field. Evaluation is also listed here as one of the seven technical-scientific grand challenges in Music IR research [Serra et al. 2013]. This recognition was also explicit during the ISMIR 2012 conference, where a discussion panel on Evaluation in Music IR was held along with a late-breaking discussion session [Peeters et al. 2012]. Even the recent *MIRrors* journal special issue on the future of Music IR research acknowledges this importance by having half the papers devoted to different aspects of Evaluation [Herrera and Gouyon 2013].

To quantitatively measure the importance and impact of evaluation studies in IR, I analyzed the proceedings of the two major conferences on Text IR and Music IR: the ACM SIGIR and ISMIR conferences. The proceedings of each edition since 1998 were examined, counting the number of publications devoted to analyzing or improving evaluation methods. Figure 1.3-left shows that on average Evaluation comprised 11% of research published in SIGIR, while in ISMIR this goes down to 6%. In fact, it is very interesting to see that the relative difference between both trends has been twofold over the years. To measure the impact of that research, the number of citations received by evaluation papers for each

[7] http://www.multimediaeval.org/mediaeval2012/newtasks/music2012/
[8] http://labrosa.ee.columbia.edu/millionsong/challenge
[9] http://www.acm.org/about/class/2012

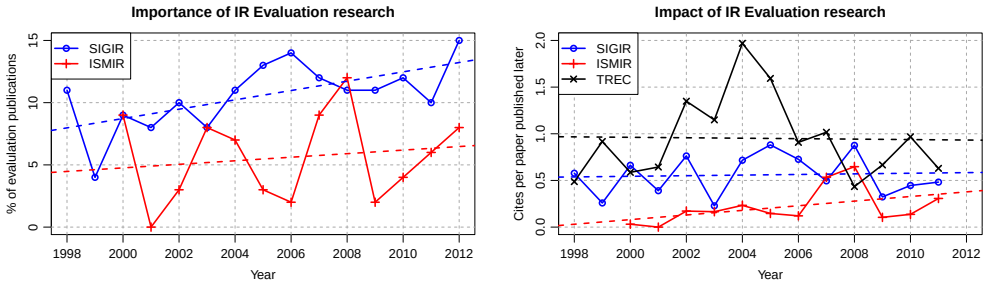

Figure 1.3: Importance (left) of publications related to IR Evaluation in SIGIR and ISMIR proceedings; and their impact (right) along with TREC overview papers.

year was also counted, and then divided the citation counts by the total number of papers (related to evaluation or not) published later and in the same venue. Figure 1.3-right shows that SIGIR papers on evaluation are cited an average of 0.6 times for each paper published later. Impact seems to be much lower in ISMIR, although the positive trend shows that the community is indeed becoming aware of the need for this research. These figures serve as a rough indication that Evaluation is in fact a very important topic of research which might not be receiving enough attention from the Music IR community yet. Another indicator of this mismatch can be found in the best paper awards: from the 17 papers awarded in SIGIR, 4 (24%) are related to evaluation. To the best of my knowledge, this has never been the case in ISMIR.

Therefore, Evaluation is not only a cornerstone in IR for allowing us to quantitatively measure which techniques work and which do not, but also a very active area of research receiving a lot of attention in recent years. We have seen this tendency in Text IR with a series of indicators which, at the same time, show that the Music IR field does not seem to pay as much attention as it probably should.

1.4 Audio Music Similarity

The Audio Music Similarity (AMS) task deals with systems that receive as query input the audio signal of a music piece and in response they have to return a list of songs from a corpus, sorted by their musical similarity to the query [Logan and Salomon 2001, Aucouturier and Pachet 2002, Seyerlehner et al. 2010b, Mcfee et al. 2012]. These systems differ from traditional music search systems in that the query input is an actual music audio signal, not just a textual query containing a section of the lyrics or metadata such as the artist and music genre [Typke et al. 2005b]. They also differ from traditional notational systems in which the query contains a sequence of pitches and durations [Urbano et al. 2011a, Doraisamy and Rüger 2003]; AMS systems work with audio signals rather than with quantized symbolic information, and in the case of MIREX there is no metadata about documents available to systems. AMS is one of the most recognizable tasks in Music IR, with clear application scenarios like music recommendation or plagiarism detection [Downie 2003a].

Besides private evaluations carried out by individuals as part of their research, public evaluation of Audio Music Similarity systems is carried out in a MIREX task with the same

Year	Teams	Systems	Queries	Documents	Judgments
2006	5	6	60	5,000	3x1,629
2007	8	12	100	7,000	4,832
2009	9	15	100	7,000	6,732
2010	5	8	100	7,000	2,737
2011	10	18	100	7,000	6,322
2012	7	10	50	7,000	2,622

Table 1.1: Summary of MIREX AMS editions. In the 2006 edition three different assessors provided annotations for every query-document pair. The task did not run in 2008

name (see Table 1.1). The AMS task ran for the first time in 2006, with five different research teams submitting six systems. The participation level has gone up and down since then, with a grand total of 69 systems evaluated in 6 MIREX editions so far[10]. The same document collection, with 7,000 audio documents, has been used since 2007.

1.5 Motivation

The impact of MIREX has been without doubt positive for the Music IR community [Cunningham et al. 2012], not only for fostering evaluation experiments, but also the study and establishment of specific evaluation frameworks for the Music domain. For some time the Music IR community accepted MIREX as "our TREC", but we are just now becoming aware of its limitations [Urbano 2011, Peeters et al. 2012]. Evaluation experiments in IR are anything but trivial [Harman 2011, Sanderson 2010, Voorhees 2002a, Tague-Sutcliffe 1992, Saracevic 1995]. Section 1.3 showed that for the past fifteen years the Text IR literature has been flooded with studies showing that evaluation experiments have their very own issues, proposing different approaches and techniques to improve the situation. While the Music IR community has inherited good evaluation practices by adopting TREC-like frameworks, some are already outdated, and most still lack appropriate analysis. I agree that not everything from the Text IR community applies to Music IR, but *many evaluation studies do*. In fact, the Music IR evaluation frameworks and body of knowledge are based on research up to the early 2000's, but about 250 evaluation papers have been published in SIGIR alone, and several landmark studies have taken place in the context of TREC since MIREX started in 2005. These studies focused mainly on large-scale evaluation, robustness and reliability, but none of them has even been considered for Music IR. In my view, this is where our community should start to improve how we evaluate systems [Urbano et al. 2013c].

The main goal of this dissertation is to improve evaluation in the Audio Music Similarity task. The approach to achieve this goal is towfold. On the one hand, I analyze the extent to which the knowledge body inherited from Text IR applies to the AMS task, and on the other hand I extend and improve the techniques used in Text IR to assess what evaluation methods work better, therefore extending the general knowledge body on IR Evaluation.

Being a task that closely resembles the ad hoc setting in Text IR, AMS evaluation experiments were designed in MIREX following the principles of other evaluation forums like TREC. The MIREX AMS task has run since 2006, and yet there has been no comprehensive

[10] Some of the MIREX AMS data can be downloaded from http://music-ir.org/mirex/wiki/.

study analyzing the appropriateness of that body of knowledge for the particular case of AMS. This issue is studied from the perspective of experimental validity, reliability and efficiency, with particular emphasis on the relationship between system- and user-measures, the optimal characteristics of test collections and statistical procedures, and the reduction of annotation costs. In doing so, modified versions of the techniques widely used in Text IR evaluation are employed. However, these techniques present some limitations that do not allow researchers to fully describe experimental results, besides theoretical and experimental gaps that make them hard to understand and apply in real situations.

1.6 Organization

Chapter 2 details the Cranfield paradigm followed in IR evaluation experiments, from which we identify the three main objects of research for this dissertation: validity, reliability and efficiency. The chapter also presents previous research on IR Evaluation, categorized according to these three criteria. Three main blocks then follow, taking on each of them.

The first block is concerned with the validity of the evaluation experiments, that is, how well the system-measures correspond to the target user-measures and how this correspondence affects the conclusions we draw from an experiment. Chapter 3 studies the relationship between system effectiveness and user satisfaction, providing an empirical mapping from the former to the latter. This mapping allows researchers to study systems from the perspective of users, it allows us to measure how much room for improvement there is for systems considering personalization, and it shows that seemingly different systems according to effectiveness may not be different according to user satisfaction. Chapter 4 then takes user satisfaction over a sample of queries, discussing the possibilities it offers as opposed to taking just averages, and showing that conclusions based on the distribution of user satisfaction may easily contradict conclusions based on the distribution of effectiveness.

The second block is concerned with the reliability of the evaluation experiments, that is, how confident we can be that our conclusions are correct and not just a random artifact of measuring performance on a sample such as a test collection. Chapter 5 compares various statistical significance tests under different optimality criteria, discussing the usually overlooked difference between practical and statistical significance. Chapter 6 then employes Generalizability Theory to analyze the optimal characteristics of test collections in terms of number of queries, assessors, etc.

The third block is concerned with the efficiency of the evaluation experiments, that is, how to make them inexpensive while still reaching valid and reliable conclusions. Chapter 7 introduces the probabilistic framework for evaluation, and develops two models to predict the relevance of documents under different circumstances. Chapter 8 then shows how the effectiveness of systems can be estimated in this probabilistic setting. The chapter then discusses how to estimate the ranking of systems without relevance judgments, and how to minimize the judging effort when estimating differences between systems or absolute scores.

Finally, Chapter 9 presents the conclusions of this work and outlines topics for further research in this line.

Chapter 2
Information Retrieval Evaluation

Most evaluation experiments in Information Retrieval follow the Cranfield paradigm to a greater or lesser extent. This paradigm is based on test collections, which are used as abstractions of the search process that users undertake in real situations. It is designed to allow rapid development of systems and reproducibility of results, but it is limited in other ways. This chapter formalizes this evaluation paradigm, presenting three aspects that must be considered when designing such evaluation experiments, namely their validity, reliability and efficiency. Past literature on IR Evaluation is then outlined under these three categories.

2.1 The Cranfield Paradigm

Batch evaluation experiments in Information Retrieval usually follow the traditional Cranfield paradigm conceived by Cyril Cleverdon half a century ago for the Cranfield II experiments [Cleverdon 1991]. The main element needed for these evaluations is a test collection, which is made up of three basic components [Sanderson 2010]: a collection of documents $\mathcal{D}$, a set of queries $\mathcal{Q}$ and a set of relevance judgments $\mathcal{R}$, compiled by a set of human assessors $\mathcal{H}$, telling what documents are relevant to what queries (the ground truth or gold standard). These test collections are built within the context of a particular task, which defines the expected behavior of the systems, the users and their information needs, and the characteristics of the documents to be considered relevant. Several effectiveness measures are used to score systems following different criteria, always from the point of view of a user model with assumptions and restrictions as to the potential real users of the systems.

A typical IR research scenario goes as follows [Harman 2011, Voorhees 2002a]. First, the task is identified and defined, normally seeking the agreement of several researchers. Depending on the task, a document collection is either put together or reused from another task, and a set of queries is selected trying to mimic the potential requests of the final users. The set of systems to evaluate return their results for the particular set of queries and document collection, and these results are then evaluated using several effectiveness measures. Doing so, we attempt to assess how well the systems would have satisfied a real user at different levels. This framework promotes rapid development and improvement of systems because it allows researchers to systematically and iteratively evaluate and com-

pare alternative algorithms and parametrizations. In that line, it also allows to reproduce experiments and repeat results across research groups by using the same test collection.

Different tasks define the user information needs in different ways. For instance, in the early TREC Ad Hoc tracks the information need would be "find documents related to some topic", and documents were considered relevant if they could be used as source to write a report on that topic [Voorhees 2002a]. Examples of topics were "language and cultural differences impeding the integration of foreign minorities in Germany" and "counterfeiting of money being done in modern times"; here there is a distinction between a topic (the instance of information need) and the query (the actual data structure provided as input to a system) [Voorhees 2002a]. In a *Named Entity Recognition* task, the information need would be "find all entities of some type", where that type is the actual query (e.g. persons, locations or organizations). In the case of *Audio Music Similarity* the information need is "find songs musically similar to the query song", and the query item given to systems as input is the audio signal itself.

Other Music IR tasks such as *Symbolic Melodic Similarity* or *Query by Humming* clearly fit into this classic retrieval setting. In other cases such as *Audio Melody Extraction* and *Audio Chord Estimation* a slightly different procedure is followed. Instead of retrieving documents in response to a query, systems provide annotations for different segments of this query item, that is, there is no distinction between documents and queries. The ground truth data does not provide information about query-document pairs, but rather about different segments of the queries. Other tasks such as *Audio Mood Classification* and *Audio Genre Recognition* are similar to annotation tasks, but instead of providing annotations for different segments of the query, systems provide tags for the query itself. Therefore, in all IR tasks systems are provided with some kind of query item and they return different output data in response, as dictated by the task.

2.1.1 Formalizing the IR Evaluation Process

User Measures

The ultimate goal of evaluating an IR system is to characterize the usage experience of the users who will employ it. We may consider several facets. For example, given an arbitrary query, we may be interested in knowing how likely it is for a user to be satisfied by the system results, or how long it would take to complete the task defined by the query. In the first case we may characterize the system response as 0 (failure) or 1 (success), and in the second case we may use the total duration in seconds required to complete the task.

We can formalize these user-measures by employing random variables. For example, we can define the discrete random variable U_1 that equals 1 if the user is satisfied by the system, and 0 otherwise. This variable U_1 is defined by a probability distribution function[1] f_{U_1}, specified by a vector of parameters $\boldsymbol{\theta}_{U_1}$. This first facet of the system is defined by f_{U_1}, and whenever a new query is run we can model the expected outcome with a random variable drawn from that distribution. We could consider the second facet with a random

[1] I use the term "probability distribution function" to indistinctly refer to the "probability density function" of a continuous random variable and the "probability mass function" of a discrete random variable.

variable U_2, equal to the task completion time in the interval $(0, \infty)$. Likewise, this variable is defined by a probability distribution function f_{U_2}, parametrized by vector $\boldsymbol{\theta}_{U_2}$.

This multifaceted characterization of the system usage allows researchers to fully assess the performance of the system from different perspectives, such as the probability of user satisfaction, the minimum time needed to complete 50% of the tasks, the probability that at least 80% of users will find the system satisfactory, etc.

Modeling Users

Unfortunately, there are several problems to know what the U_i distributions look like. First, including real users in evaluation experiments is not only expensive but also complex, and there are always ethical issues to consider (e.g. privacy and wages). Second, involving users makes it harder to tune system parameters due to the cost of running an evaluation trial. Third, it is hard to reproduce experiments that involve human subjects, so system comparisons across research groups becomes quite difficult. An example earlier to the Cranfield I experiments can be found in the ASTIA-Uniterm test in 1953: the two participating teams could not agree on the relevance of documents, and so each team produced their own results for the same experiment [Gull 1956]. To minimize these problems, Cleverdon came up with the idea of removing actual users from the evaluation process but including a static user component: the relevance judgments in the ground truth. He was able to control the experiment and reduce all sources of variability to just the systems themselves, making it possible to iteratively compare systems in a systematic, fast and inexpensive way.

Therefore, when evaluating a system following the Cranfield framework we are actually characterizing the system response rather than the user experience. The ground truth provides us with information on how good or accurate that response is, but it does not provide information on the user-system interaction, let alone on user-specific characteristics such as perceived easiness in using the system. Likewise, each of the system-based measures used in the evaluation experiment provides us with a description of the system from different perspectives, each of which can again be modeled with random variables. For instance, when evaluating music similarity systems we may use a random variable S_1 to refer to the average similarity of the items returned by the system, and another variable S_2 might refer to the rank at which the system retrieves the first similar item. These variables are computed with effectiveness measures Λ_i (e.g. *Cumulative Gain* for S_1 and *Reciprocal Rank* for S_2, see Section 2.5), and they are also defined by probability distribution functions f_{S_1} and f_{S_2}, parametrized by vectors $\boldsymbol{\theta}_{S_1}$ and $\boldsymbol{\theta}_{S_2}$, respectively. The assumption underlying Cranfield is that S_i is correlated with U_i, and therefore the distribution defined by f_{S_i} can somehow be used to describe the distribution defined by f_{U_i}.

Parameter Estimation

Computing the parameter vector $\boldsymbol{\theta}_{S_i}$ is clearly impossible; it requires to evaluate a system with the universe of all queries, documents and assessors; we would need all existing queries and all queries yet to exist, which are potentially infinite. Instead, we evaluate with the sample of queries $\mathcal{Q}$ in a test collection. When we evaluate a system according to an effectiveness measure Λ_i, we compute a score λ_{iq} for each query $q \in \mathcal{Q}$. When we repeat

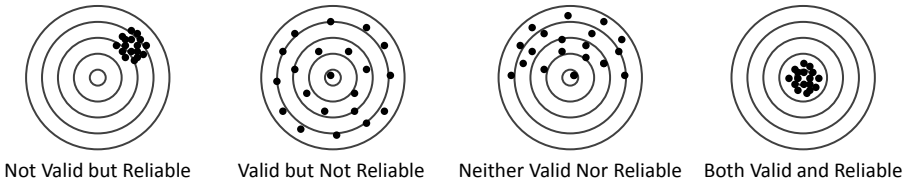

Figure 2.1: Validity and Reliability. Adapted from [Trochim and Donnelly 2007].

the process with all queries in the sample, the empirical distribution is used to estimate the distribution f_{S_i} that defines the random variable S_i associated with the effectiveness measure Λ_i. That is, we are estimating the parameters $\boldsymbol{\theta}_{S_i}$, and because we assume the correlation between S_i and U_i, we treat those system estimates $\hat{\boldsymbol{\theta}}_{S_i}$ as estimates themselves of the $\boldsymbol{\theta}_{U_i}$ parameters of the user-based distribution.

The problem is knowing the distribution family for each facet. For example, the user satisfaction variable U_1 may be modeled with a Bernoulli distribution with parameter p, and then a Binomial distribution can be used to compute the probability that n out of m users will find the system satisfactory. The completion time variable U_2 may be modeled with a Gamma distribution with parameters α and β, but we could use instead a Log-Normal distribution with parameters μ and σ^2. There is really no theoretical basis for using one distribution family or another, so researchers tend to ignore the shape of the distributions and focus just on the first and second moments, the mean $\mathrm{E}[U_i]$ and the variance $\mathrm{Var}[U_i]$, estimated with the sample mean and variance of the empirical distribution.

2.1.2 Validity, Reliability and Efficiency

In summary, we can look at an IR evaluation experiment as just an estimator of the true parameters defining a user-based distribution. An effectiveness measure is our measurement instrument, whose system-based distribution is assumed to perfectly correlate with our target user-based distribution. As such, there are three aspects of these evaluation experiments that must be considered: validity, reliability and efficiency [Tague-Sutcliffe 1992]:

Validity. Do our effectiveness measures and ground truth data really define system-distributions that match the intended user-distributions? We assume there is some function mapping S_i to U_i, and therefore f_{S_i} to f_{U_i}. In fact, researchers somehow assume $U_i = S_i$, or $U_i \propto S_i$ at the very least. In a more relaxed form, validity can be reformulated as: are we really measuring what we want to measure?

Reliability. How many queries are needed in the test collection so that the estimates can be trusted? The more queries we use, the smaller the standard error we have in our estimates, but the higher the cost too. A similar issue is found with the human assessors because the ground truth is subjective, so it is expected that results vary to the extent the assessors and the final users disagree as to the relevance of documents. Therefore, evaluation experiments must find a tradeoff between reliability and effort. In a more relaxed form, reliability can be formulated as: how repeatable are our results?

Efficiency. Creating a ground truth set is usually a very expensive and tedious task, and some forms of ground truth data can be prohibitive for a large number of queries. Therefore,

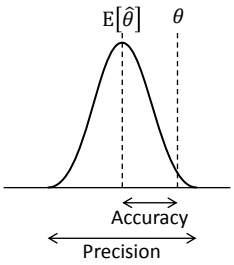

Figure 2.2: Accuracy and Precision. θ is the true value and $\mathrm{E}[\hat{\theta}]$ is the estimated value.

the efficiency of the ground truth annotation process directly impacts the reliability of the evaluation. On the other hand, an efficient annotation process might be inaccurate, lowering the validity of the results. Therefore, evaluation experiments must also find a balance between validity and reliability and the cost of the annotation process. In a relaxed form, efficiency can be formulated as: is there a cheaper way to obtain valid and reliable results?

Figure 2.1 illustrates validity and reliability with the metaphor of a target. Imagine our goal is the center of the target (i.e. $\mathrm{E}[U_i]$), and each shot we take is our measurement with a different test collection. In the first and fourth examples we have an instrument that is very reliable, but in the first case we are clearly off the target. In the second and third examples our instrument is not reliable, but in the second case we still manage to hit around the target so that our measure is correct on average. In this case, efficiency can be thought of as the cost of the weapon: rifle, bow, handgun, slingshot, etc. In Statistics terms, validity refers to the *accuracy* and *bias* of the estimates, and reliability refers to their *precision* or *variance* [Lehmann and Casella 1998]. That is, how close they are to the true parameters and how much uncertainty there is in those estimates (see Figure 2.2). In Machine Learning terms, validity refers to the *bias* of a learner, and reliability refers to its *variance* [Geman et al. 1992]. That is, the average difference over training datasets between the true values and the predictions, and how much they vary across training datasets. They can also be linked to the concepts of *systematic* and *random* error in measurement [Taylor 1997]. Thus, we may describe the IR evaluation process with the following basic equation:

$$U = S + \varepsilon_s + \varepsilon_r \tag{2.1}$$

were ε_s is the systematic error and ε_r is the random error.

2.2 Validity

Validity is the extent to which an experiment actually measures what the experimenter intended to measure [Shadish et al. 2002, Trochim and Donnelly 2007]. Validity is frequently divided in four types that build upon each other, addressing different aspects of an experiment. *Conclusion Validity* relates to the relationship found between our experimental treatments (systems) and our response variables (user-measures). Can we conclude that the systems are different? How much different? *Internal Validity* relates to confounding factors that might cause the differences we attribute to the systems. Are those differences caused by

specific characteristics of the annotators or the queries? *External Validity* relates to the generalization of that difference to other populations. Would system differences remain for the wider range of all genres and artists? *Construct Validity* relates to the actual relationship between the system-measures and the user-measures. Do differences in system-measures directly translate into the same differences in user-measures? How do those differences affect end users?

2.2.1 Conclusion Validity

Effectiveness measures are usually categorized as precision- or recall-oriented. Therefore, it is expected for precision-oriented measures to yield effectiveness scores correlated with other precision-oriented measures, and likewise with recall-oriented ones. However, this does not always happen [Sakai 2007, Kekäläinen 2005], and some measures are even better correlated with others than with themselves [Webber et al. 2008b], evidencing problems when predicting user-measures. In general, system-measures should be correlated with user-measures, but observing a difference between two systems according to some system-measure does not necessarily mean there is a noticeable difference with end users. For example, it can be the case that relatively large differences need to appear between systems for users to actually note them.

At this point it is important to note that in most situations systems are not provided with any kind of user information [Järvelin 2011, Schedl et al. 2013a], and therefore our results should be interpreted as if targeting *arbitrary* users. As such, even if our system-measures corresponded perfectly to user-measures, the system distributions estimated with an evaluation experiment would not correspond perfectly to the expected user distributions because we are not accounting for user factors in the ground truth data [Voorhees 2000].

It is also important to recall that an evaluation experiment provides an estimate of a true population mean, which bears some degree of uncertainty due to sampling. Confidence intervals should always be calculated when drawing conclusions from an experiment, to account for that uncertainty and provide reliable reports of effect sizes [Cormack and Lynam 2006]. Depending on the experimental conditions, it might be the case that the interval is too wide to draw any reliable conclusion regarding the true performance of systems. In this line, it is important to distinguish between *confidence intervals*, used as estimators of distribution parameters such as the true mean; and *prediction intervals*, which serve as estimators of the expected performance on a new query item.

2.2.2 Internal Validity

Ground truth data is a much debated part of IR Evaluation because of the subjectivity component it usually has. Several studies show that documents are judged differently by different people in terms of their relevance to some specific query, even by the same people over time [Schamber 1994]. As such, the validity of evaluation experiments can be questioned because different results are obtained depending on the people that make the annotations. Nevertheless, it is generally assumed that ground truth data is invariable, and user-dependent factors are ignored [Järvelin 2011, Schedl et al. 2013a]. Several studies have shown that absolute scores do indeed change, but that relative differences between systems

stand still for the most part [Voorhees 2000]. For domain-specific tasks though, results may suffer large variations [Bailey et al. 2008], and for very large-scale experiments different assessor behaviors may also have a large impact on the results [Carterette and Soboroff 2010], let alone if the ground truth has inconsistencies.

Likewise, if a low-cost evaluation method were used with an incomplete ground truth (see Section 2.4), systems more alike could reinforce each other, while systems with novel technology might be harmed [Zobel 1998]. In general, making assumptions regarding the missing annotations can affect both the measures [Buckley and Voorhees 2004, Sakai and Kando 2008] and the overall results [Buckley et al. 2007]. This is an obvious problem because the very test collection (documents, queries and ground truth), *which is in its own a product of the experiment*, might not be reusable for subsequent evaluations of new systems [Carterette et al. 2010a,b].

The particular queries used could also be unfair if some systems were not able to fully exploit their characteristics. This is of major importance for tasks where Machine Learning is heavily employed in systems that are first tuned with a training collection: if the query or document characteristics were different between the training and evaluation collections, systems could be misguided, resulting in researchers reaching wrong conclusions from the experiment. On the other hand, if the same collections were used repeatedly, an increase in performance could be just due to overfitting and not to a real improvement [Voorhees 2002a]. Also, some evaluation measures could be unfair to some systems if accounting for information they cannot provide.

2.2.3 External Validity

In IR Evaluation it is very important to clearly define what our *target populations* are. That is, who our final users are, the music corpora they will work with, etc. When we carry out an experiment to evaluate a system, we are interested in the distributions of user-measures for those populations. The problem is that we might not be able to get access to those users (e.g. anonymous users of an online music service, music artists, etc.) or those corpora (e.g. copyrighted material or songs yet to exist). Therefore, we often have access only to restricted and biased subsets of those populations. These are the *accessible populations*. To reduce costs, we draw a *sample* from those accessible populations and carry out the experiment. Our assumption when doing this is that the results obtained with our samples can be generalized back to the target populations. In particular, for an arbitrary system-measure we assume that the sample mean is an unbiased estimator of the true population mean because our sample is *representative* of the target population. This is not necessarily true if the accessible and target populations have different characteristics or the sampling method is not appropriate.

This is probably the weakest point in IR Evaluation [Voorhees 2002a, Cormack and Lynam 2006, Robertson and Kanoulas 2012, Sanderson et al. 2012]. In order to get a sample representative of the accessible population we generally want that sample to be large; the more elements we draw the better our estimates will be. This poses obvious problems in terms of cost. Having large corpora means that the completeness of the ground truth is compromised; it is just not feasible to judge every query-document pair or annotate

every single segment of every query [Buckley and Voorhees 2004, Zobel 1998]. As a result, collections contain too few query items or their corpus is too small to be representative.

In addition, we want the sample to be random in order to eliminate biases. In Text IR, this has been a problem since the early days, because there was no pool of queries to draw a sample from; they were made up on demand for the evaluation experiments [Voorhees and Harman 2005]. Because of this, the Text IR literature has always emphasized that results with a single test collection must be taken with a grain of salt because results are highly dependent on document collections and query sets [Robertson 2011, Voorhees 2002a]; that is, systems may work very well with a test collection but significantly worse with a different one [Poibeau and Kosseim 2001], especially if Machine Learning algorithms are involved. This is also emphasized in that results should be interpreted in terms of relative pairwise system differences rather than absolute scores. That is, comparisons across collections and claims about the state of the art based on a single collection are generally not justified.

To partially overcome this problem with non-random samples, the Text IR community has traditionally sought very large collections. In the last decade though, several sources of information, such as query logs from commercial search engines, have been used to draw random samples and slightly reduce the cost. This has the additional advantage that queries are likely to be representative of the final user needs, although the actual distributions of queries may be hard to sample from because they tend to be highly skewed [Zaragoza et al. 2010]. A similar problem arises in Music IR because the accessible population is hardly representative of the target population (e.g. copyrighted music material is nearly impossible to use in test collections), so even if we have a very large sample we still can not generalize back as we would like. Recent research has studied query selection methods that try to avoid queries that do not provide useful information to differentiate between systems [Guiver et al. 2009, Robertson 2011].

2.2.4 Construct Validity

In IR evaluation experiments, Construct Validity is concerned mainly with the system-measures used, their underlying user model [Carterette 2011], and their correlation with user-measures. Unlike batch experiments where the only user component is the ground truth, there have been some experiments with actual users interacting with IR systems. They found little correlation between system-measures and user-measures, questioning the whole Cranfield paradigm [Hersh et al. 2000, Turpin and Hersh 2001]. But the problem strives in seeking correlations between measures that are not really supposed to be related [Smucker and Clarke 2012a]. For instance, *Precision* is not designed as an indicator of task completion time; *Reciprocal Rank* is. Various alternatives have been studied, such as using different relevance thresholds on a per-assessor basis [Scholer and Turpin 2008], carefully normalizing effectiveness scores [Al-Maskari et al. 2007], or including other factors in the measurement of relevance [Smucker and Clarke 2012b, Yilmaz et al. 2010, Huffman and Hochster 2007]. Later work further explored this issue, finding clear correlations between system effectiveness and user satisfaction [Allan et al. 2005, Sanderson et al. 2010]. A similar study in the Audio Music Similarity task has appeared recently following the principles of Hersh et al. [2000], showing little relationship between measures [Hu and Kando 2012].

The development of appropriate system-measures that closely capture the user experience is thus very important. For instance, in a traditional ad hoc retrieval task binary set-based measures such as *Precision* and *Recall* do not resemble a real user who wants not only relevant documents, but highly relevant ones at the top of the results list [Sanderson et al. 2010]. Instead, measures that take the rank into account [Moffat and Zobel 2008], graded relevance judgments [Voorhees 2001, Kekäläinen 2005], or a combination of them [Järvelin and Kekäläinen 2002, Robertson et al. 2010, Chapelle et al. 2009, Kanoulas and Aslam 2009, Sakai 2004], are more appropriate. Other forms of ground truth can also be studied [Bennett et al. 2008], such as preference judgments [Carterette et al. 2008].

2.3 Reliability

Reliability is the extent to which the results of the experiment can be repeated [Trochim and Donnelly 2007, Tague-Sutcliffe 1992]. Will we obtain similar results if we replicate the experiment with different sets of documents, queries and assessors?

As mentioned, it is very important that our samples are representative of the target populations. It is important not only because we want our estimates to closely correspond to the true population parameters, but also because our results would otherwise be unreliable: with one sample system A is better than system B, but with another sample it is the other way around. That is, we can not repeat the results. There are three main factors that influence reliability: the effectiveness measures, the size of our samples and the agreement between human annotators.

Two characteristics of the effectiveness measures used in IR evaluation experiments are their stability and sensitivity. The results should be stable under different annotators and query sets, so they do not vary significantly and alter the conclusions as to what systems are better [Buckley and Voorhees 2000]. They are also desired to discriminate between systems if they actually perform differently [Voorhees and Buckley 2002, Sakai 2007], and to do so with the minimum effort [Sanderson and Zobel 2005]. However, they are desired to not discriminate between systems that actually perform very similarly. These performance differences must always be considered in the context of the task and its user model.

In general, the more queries we use the more stable the results and therefore the more reliable, because we compute estimates closer to the true values and their variance is reduced. Estimating how many queries are enough to reach some level of reliability is a quite tedious process if following a data-based approach such as system swap rates [Buckley and Voorhees 2000, Voorhees and Buckley 2002, Sakai 2007, Sanderson and Zobel 2005]. A simpler yet more powerful approach based on statistical theory can be followed with *Generalizability Theory* [Bodoff and Li 2007, Urbano et al. 2013b, Salamon and Urbano 2012]. It allows to measure how much variability is due to facets like queries or annotators, so it can be decided where to spend more resources to increase reliability. It can be used to measure the stability of a test collection *while* it is being developed, but it can also be used to estimate the stability of a different experimental design, or to estimate the point at which it is preferable to employ more annotations and the current query set rather than just including more queries.

Given a set of systems and the resulting effectiveness distributions obtained with different queries according to some system-measure, they are usually compared in terms of their mean

effectiveness score. This can be problematic, because those means are just estimates of the true population means, and are therefore subject to random error due to sampling. Not until relatively recently, statistical methods have been systematically employed to compare systems by their score distribution rather than just their sample mean score [Carterette 2012, Sakai 2006, Carterette and Smucker 2007, Webber et al. 2008a]. It is also very important to study which statistical methods are more appropriate, because their assumptions are known to be violated in IR evaluation experiments [Urbano et al. 2013a, Smucker et al. 2007, Zobel 1998]. At this point, it is important to correctly interpret the results and understand the very issues of hypothesis testing and, most importantly, distinguish between *statistical* and *practical* significance [Ioannidis 2005, Ziliak and McCloskey 2008]. Even if one system is found to be statistically significantly better than another one, the difference might be extremely small; too small to even be noticed by users. On the other hand, the tiniest practical difference will turn out statistically significant with a sufficiently large collection.

2.4 Efficiency

Efficiency is the extent to which the experimenter gets to a valid and reliable conclusion at a low cost [Trochim and Donnelly 2007, Tague-Sutcliffe 1992]. Are there other annotation procedures or alternative evaluation methods that result in a more cost-effective experiment?

Annotations for test collections are usually made by experts, which increases the cost of building large datasets. Some recent work examined the use of non-experts for relevance judging [Bailey et al. 2008], and found that in general there are no noticeable differences in the evaluation results, although clear differences exist when the task is very specialized. Others explore the use of paid crowdsourcing platforms such as Amazon Mechanical Turk [Alonso and Mizzaro 2012, Lease and Yilmaz 2011, Kittur et al. 2013] to gather annotations for a very low cost. The problem in these cases is the potential low quality of the results. Some quality control techniques are based on known answers [Sanderson et al. 2010], redundant answers to compute consensus [Ipeirotis et al. 2010, Snow et al. 2008] or detection of neglecting behavior [Kittur et al. 2008, Urbano et al. 2011b, Rzeszotarski and Kittur 2011].

Other research focused on the use of *incomplete* ground truth data where not all annotations are present in the test collections. A first approach to reduce the number of annotations in test collections was the pooling technique [Buckley and Voorhees 2004]. Instead of annotating all documents retrieved by all systems, a pool with the top-k results from all systems is formed, and only those are annotated. All documents outside the pool are then assumed to be non-relevant. This technique has been used in Text IR for many years, and it has been repeatedly shown to be reliable despite the non-relevance assumption, permitting the use of large collections by reducing the annotation cost to about 35%. With very large collections though, it is shown to have problems [Buckley et al. 2007]. Different modifications of the basic pooling technique have been proposed via interactive annotation processes [Zobel 1998, Cormack et al. 1998], meta-search models [Aslam et al. 2003], intelligent selection of documents to judge [Moffat et al. 2007] or ignoring them altogether [Buckley and Voorhees 2004, Sakai and Kando 2008]. Other alternatives studied the evaluation of systems even when annotations are not available at all [Soboroff et al. 2001], which es useful as a lower bound on evaluation reliability.

More recent work has focused on the inference of annotations based on a *very* incomplete set of previous annotations, using a more probabilistic view of evaluation. Some techniques focus on sampling theory [Aslam and Yilmaz 2007], document similarities [Carterette and Allan 2007] or meta-search [Carterette 2007]. The inferred data are then used to estimate effectiveness scores based on random samples of annotations [Yilmaz and Aslam 2006, Yilmaz et al. 2008]; or to estimate the ranking of systems by annotating only those documents that are more informative to tell the difference between systems [Carterette et al. 2006, Carterette 2007]. These low-cost techniques have been studied mainly in the TREC Million Query Track between 2007 and 2009, offering very reliable results for a very low cost of annotation. In fact, they allowed a dramatic increase in the number of queries from a few dozens to over a thousand [Carterette et al. 2009].

2.5 Effectiveness Measures

Given a ranked list of documents returned by a system for some query and a set of relevance judgments, a measure Λ is used to assess the effectiveness of the system for the query. In this section I review several measures as used in the IR literature. For simplicity, I will refer to an effectiveness score $\lambda_{q,\mathsf{A}}$ simply as λ, assuming some arbitrary system A and query q.

2.5.1 Binary Relevance Scale

Traditionally, the relevance of a document d has been assessed with a binary relevance scale $\mathcal{L} = \{0, 1\}$, that is, $n_{\mathcal{L}} = 2$. If a document d is deemed as relevant to the query then $r_d = 1$, and if it is not then $r_d = 0$.

Precision

A simple effectiveness measure to evaluate a retrieval run A when $n_{\mathcal{L}} = 2$ is Precision at k documents retrieved ($P@k$). Its purpose is to measure the noise introduced by the system in the top k documents retrieved:

$$P@k = \frac{1}{k} \sum_{i=1}^{k} r_{\mathsf{A}_i} \tag{2.2}$$

that is, it computes the average relevance of the top k documents retrieved by A. If $P@k = 0$ it means that the system was not able to retrieve any document relevant to the query, and $P@k = 1$ means all documents were indeed relevant.

Average Precision

$P@k$ does not provide by itself any information on the ordering of documents. If a system retrieved relevant documents at the top of the ranking it would have the same $P@k$ score as a system retrieving the same documents at the very bottom. It is clear that a user inspecting results from the first system will find useful information quicker than with the second one. Average Precision at k documents retrieved ($AP@k$) does account for the ordering of documents as follows [Harman 1993]:

$$AP@k = \frac{1}{|\mathcal{R}^1|} \sum_{i=1}^{k} r_{\mathsf{A}_i} \cdot P@i \tag{2.3}$$

that is, the average of precisions computed at ranks where relevant documents are retrieved. If a pivot document A_i is not relevant, the $r_{\mathsf{A}_i} = 0$ and therefore $P@i$ does not contribute to the summation. The final score is normalized dividing by the total number of relevant documents $\left|\mathcal{R}^1\right|$. In the example above, the first system would have a higher $AP@k$ score.

Reciprocal Rank

In several situations, the retrieval task aims at finding one relevant document. This is the case of Question Answering systems or Known-Item search. In these cases we are interested in measuring the ability of the system to return a relevant document at a high rank. Reciprocal Rank (RR) is defined as [Kantor and Voorhees 1996]:

$$RR = \frac{1}{\min\{i : r_{\mathsf{A}_i} = 1\}} \tag{2.4}$$

that is, the inverse of the rank at which the first relevant document is retrieved.

2.5.2 Graded Relevance Scale

Despite the binary $n_{\mathcal{L}} = 2$ relevance scale has been the standard in TREC and other forums, Cleverdon [1991] already used three levels of relevance in the Cranfield I experiments and as many as five in Cranfield II. In the early 2000's it became apparent that some systems worked well for retrieving highly relevant documents, while others were better at retrieving all relevant documents Voorhees [2001]. In addition, there is the notion of *perfect* documents for some types of queries, such as navigational queries (e.g. the homepage of *Youtube*), while other documents may still be somewhat relevant. Consequently, graded $n_{\mathcal{L}} > 2$ relevance scales were increasingly adopted [Järvelin and Kekäläinen 2000].

Binary Measures with Graded Relevance Scales

One option to handle graded relevance judgments is to use binary measures and conflate relevance levels with a threshold $\ell_{min} \in \mathcal{L}$ such that if the relevance r_d of a document d is $r_d < \ell_{min}$ it is considered not relevant, and if it is $r_d \geq \ell_{min}$ it is considered relevant. This way, we can easily redefine $P@k$, $AP@k$ and RR as follows:

$$P@k = \frac{1}{k} \sum_{i=1}^{k} \mathbb{1}(r_{\mathsf{A}_i} \geq \ell_{min}) \tag{2.5}$$

$$AP@k = \frac{1}{|\mathcal{R}^{\geq \ell_{min}}|} \sum_{i=1}^{k} \mathbb{1}(r_{\mathsf{A}_i} \geq \ell_{min}) \cdot P@i \tag{2.6}$$

$$RR = \frac{1}{\min\{i : r_{\mathsf{A}_i} \geq \ell_{min}\}} \tag{2.7}$$

The drawback of using binary measures and a threshold ℓ_{min} is that effectiveness scores have to be reported several times for different values of ℓ_{min}.

Cumulative Gain

Järvelin and Kekäläinen [2002] proposed a family of measures that directly handle graded relevance scales. These measures are based on the concept of utility provided by a returned document. Let $g(\ell)$ be a gain function that maps relevance level ℓ to a utility score: $g : \mathcal{L} \to \mathbb{R}^{\geq 0}$. Additionally, the following two restrictions are imposed: $g(0) = 0$ and $\ell_i > \ell_j \Rightarrow g(\ell_i) \geq g(\ell_j)$ (i.e. it is monotonically increasing). The gain is larger as the relevance score is larger, and a relevance score of 0 is always mapped to a gain of 0.

This way, we can define the Cumulative Gain at k documents retrieved ($CG@k$) as the total gain provided by the top k documents:

$$CG@k = \sum_{i=1}^{k} g(r_{\mathsf{A}_i}) \tag{2.8}$$

In the initial definition of these measures Järvelin and Kekäläinen [2002] used the straightforward gain function $g(\ell) = \ell$, so that the gain provided by a document could range from 0 to $n_{\mathcal{L}} - 1$. However, the gain function can be arbitrary.

Discounted Cumulative Gain

$CG@k$ has the same problem as $P@k$, that is, that it ignores the rank of documents. Similarly, if a system returns highly relevant documents towards the top of the list or towards the end, does not make any difference for the computation of $CG@k$. To account for the fact that documents retrieved at lower ranks are less useful than those retrieved at higher ranks, Järvelin and Kekäläinen [2002] also introduced a discount function $d : \mathbb{N}^{>0} \to \mathbb{R}^{>0}$ to reduce the gain of a document depending on the rank at which it is retrieved. Likewise, the following restriction is imposed to a discount function: $i > j \Rightarrow d(i) \geq d(j)$ (i.e. it is monotonically increasing).

Discounted Cumulative Gain after k documents retrieved ($DCG@k$) is thus a measure like $CG@k$ that also discounts the gain of documents as they are returned down the list:

$$DCG@k = \sum_{i=1}^{k} \frac{g(r_{\mathsf{A}_i})}{d(i)} \tag{2.9}$$

In the original definition of $DCG@k$ Järvelin and Kekäläinen [2002] used a logarithmic discount function as follows:

$$d(i) = \begin{cases} 1 & i < b \\ \log_b i & i \geq b \end{cases}$$

The logarithm base b is a parameter to model user persistence; the larger it is the lower the discount for a given rank and therefore the larger the utility at that rank. Järvelin and Kekäläinen [2002] suggested the base $b = 2$, so their original formulation can be written simply as $d(i) = \max(1, \log_2 i)$.

In a later paper, Burges et al. [2005] proposed an alternative definition for the gain and discount functions: $g(\ell) = 2^\ell - 1$ and $d(i) = \log_2(i + 1)$. These definitions can still be generalized to an arbitrary base b: $g(\ell) = b^\ell - 1$ and $d(i) = \log_b(i + b - 1)$. This way, they strongly emphasized highly relevant documents and also discounted *all* documents according to their rank, not only those where $i \geq b$. This definition with $b = 2$ is the de-facto standard used by the community, sometimes known as *Microsoft DCG*.

Normalized Discounted Cumulative Gain

The last measure proposed by Järvelin and Kekäläinen [2002] accounts for the ideal ranking of documents sorted by relevance $\mathsf{I} := \langle \mathsf{I}_1, \ldots, \mathsf{I}_{n_\mathcal{D}} : \forall i : r_{\mathsf{I}_i} \geq r_{\mathsf{I}_{i+1}} \rangle$ to normalize a $DCG@k$ score. They defined the Normalized Discounted Cumulative Gain at k documents retrieved as the $DCG@k$ of the system A divided by the $DCG@k$ of the ideal ranking I:

$$nDCG@k = \frac{\sum_{i=1}^{k} g(r_{\mathsf{A}_i}) / d(i)}{\sum_{i=1}^{k} g(r_{\mathsf{I}_i}) / d(i)} \tag{2.10}$$

Similarly, Burges et al. [2005] defined the so-called *Microsoft nDCG* measure by using the alternative definitions for the gain and discount functions that we saw above.

Q-Measure

Sakai [2004] proposed a new effectiveness measure to handle graded relevance judgments. It was designed to cope with the late discount problem in the original definition of Järvelin and Kekäläinen [2002] for large b, and to generalize to Average Precision when the relevance scale is binary. The Q-measure at k documents retrieved ($Q@k$) can be defined as:

$$Q@k = \frac{1}{|\mathcal{R}^{>0}|} \sum_{i=1}^{k} \mathbb{1}(r_{\mathsf{A}_i} > 0) \frac{\sum_{j=1}^{i} \mathbb{1}\big(r_{\mathsf{A}_j} > 0\big) + \beta \cdot \sum_{j=1}^{i} g\big(r_{\mathsf{A}_j}\big)}{i + \beta \cdot \sum_{j=1}^{i} g\big(r_{\mathsf{I}_j}\big)} \tag{2.11}$$

where β is a tuning parameter usually set to $\beta = 1$. When $\beta = 0$ it is easy to see that $Q@k$ in (2.11) reduces to $AP@k$ in (2.3) if using a binary $n_\mathcal{L} = 2$ scale, and to (2.6) with a graded scale and a threshold ℓ_{min} instead of 0.

Rank-Biased Precision

Moffat and Zobel [2008] argued against the discount function in $(n)DCG$ and proposed a model of user behavior based on a persistence parameter p: with probability p the user moves to the next document in the ranking, and with probability $1-p$ she does not. The probability that the user reaches the document at rank i and stops there is therefore $p^{i-1} \cdot (1-p)$.

Moffat and Zobel [2008] originally defined the Rank-Biased Precision (RBP) measure for a binary $n_\mathcal{L} = 2$ relevance scale:

$$RBP = (1-p) \sum_{i=1}^{n_\mathcal{D}} r_{\mathsf{A}_i} \cdot p^{i-1}$$

RBP can easily be generalized to a graded $n_\mathcal{L} > 2$ scale by substituting r_{A_i} with $g(r_{\mathsf{A}_i})$:

$$RBP = \frac{1-p}{g(n_\mathcal{L} - 1)} \sum_{i=1}^{n_\mathcal{D}} g(r_{\mathsf{A}_i}) \cdot p^{i-1} \tag{2.12}$$

where the $g(n_\mathcal{L} - 1)$ factor is used to normalize the score between 0 and 1. This formulation is implemented for example in the `ntcireval` evaluation package used in NTCIR.

Expected Reciprocal Rank

Both $(n)DCG$ and RBP incorporate a form of discount that penalizes the need of the user to traverse the ranked list of documents. However, this discount function depends only on the rank of a document, ignoring the relevance of all documents ranked above it. Chapelle et al. [2009] extended the discount notion to incorporate this factor. Let us assume that the user inspects results from top to bottom, and let p_d be the probability that the user is satisfied with document d. Chapelle et al. [2009] defined this probability as:

$$p_d = \frac{2^{r_d} - 1}{2^{n_{\mathcal{L}}-1}} \tag{2.13}$$

With probability $1 - p_d$ the user is not satisfied with document d and moves on to the next one in the list. Therefore, the probability that the user starts from the top of the list and stops at rank i is:

$$p_{\mathsf{A}_i} \cdot \prod_{j=1}^{i-1} 1 - p_{\mathsf{A}_j}$$

Under this model of user behavior, we can define the Expected Reciprocal Rank (ERR) by fixing the contribution of the i-th document to be $1/i$:

$$ERR = \sum_{i=1}^{n_{\mathcal{D}}} \frac{1}{i} \left(p_{\mathsf{A}_i} \prod_{j=1}^{i-1} 1 - p_{\mathsf{A}_j} \right) \tag{2.14}$$

If we chose the contribution to be $g(r_{\mathsf{A}_i})$ rather than $1/i$, we would obtain a redefinition of DCG with a more elaborated discount function. If we further consider a gain function to define the probability of satisfaction p_d, we may formulate it as

$$p_d = \frac{g(r_d)}{g(n_{\mathcal{L}} - 1) + 1} \tag{2.15}$$

which is identical to (2.13) when $g(\ell) = 2^{\ell} - 1$ as defined by Burges et al. [2005]. Therefore, we can generalize the ERR definition in (2.14) with an arbitrary gain function as in (2.15).

Graded Average Precision

Robertson et al. [2010] proposed a generalization of $AP@k$ to graded $n_{\mathcal{L}} > 2$ relevance scales that similarly assumes a relevance threshold $\ell_{min} \in \mathcal{L}$ such that a document d is considered relevant if $r_d \geq \ell_{min}$ and not relevant if $r_d < \ell_{min}$. However, this threshold is not fixed as in (2.6). Instead, Robertson et al. [2010] made it probabilistic, defining p_ℓ as the probability that an arbitrary user implicitly sets $\ell_{min} = \ell$. By defining p_ℓ over the space of users, we have $\sum_{\ell=1}^{n_{\mathcal{L}}-1} p_\ell = 1$. Note that p_0 is always assumed to be 0.

For arbitrary rank k and threshold ℓ, the binary precision is defined similar to (2.5):

$$P@k_\ell = \frac{1}{k} \sum_{i=1}^{k} \mathbb{1}(r_{\mathsf{A}_i} \geq \ell)$$

Taking into account that a document with relevance r_d only contributes to the binary precisions of levels $\{1, \ldots, r_d\}$, the expected precision over the space of users is:

$$\mathrm{E}[P@k] = \frac{1}{k} \sum_{\ell=1}^{r_{\mathsf{A}_k}} p_\ell \cdot \sum_{i=1}^{k} \mathbb{1}(r_{\mathsf{A}_i} \geq \ell)$$

Computing these precisions along the ranking and normalizing by the maximum possible, the Graded Average Precision at k documents retrieved ($GAP@k$) is:

$$GAP@k = \frac{\sum_{i=1}^{k} \mathrm{E}[P@i]}{\sum_{\ell=1}^{n_{\mathcal{L}}-1} |\mathcal{R}^{\ell}| \sum_{s=1}^{\ell} p_{\ell}} \tag{2.16}$$

The computation of a $GAP@k$ score still depends on the probability distribution of p_{ℓ}. Without any specific data regarding actual users, Robertson et al. [2010] recommend the uniform distribution $p_{\ell} = 1/(n_{\mathcal{L}} - 1)$ for being the most informative.

Average Dynamic Recall

Typke et al. [2006] proposed a measure specifically for Symbolic Melodic Similarity, designed to handle ground truth data in the form of partially ordered lists [Typke et al. 2005a, Urbano et al. 2010a]. These lists do not contain a relevance judgment for a query-document pair as usual, but groups of documents equally relevant to the query. Some groups are more relevant than others, but the magnitude of this difference is not defined because there is no pre-fixed relevance scale. Nonetheless, it can be extended to handle a prefixed scale $\mathcal{L}$, so we can define Average Dynamic Recall at k documents retrieved ($ADR@k$) as:

$$ADR@k = \frac{1}{k} \sum_{i=1}^{k} \frac{|\{d \in \mathcal{D} : r_d \geq r_{\mathsf{l}_i}\} \cap \{\mathsf{A}_1, \ldots, \mathsf{A}_i\}|}{i} \tag{2.17}$$

At each rank i it computes the fraction of documents in an ideal ranking up to that point that are indeed retrieved by the system, averaging across ranks. Because this proportion is 1 almost surely for a sufficiently large rank, it is recommended to compute $ADR@k$ at cut-off $k = \left|\mathcal{R}^{>0}\right|$.

Chapter 3

System Effectiveness and User Satisfaction

The construct validity of an Information Retrieval evaluation experiment was identified in the previous chapter as the extent to which the system-oriented effectiveness measures correspond to the target user-oriented measures. Evaluation experiments following the Cranfield paradigm consider users from a static point of view so that we can reproduce experiments and compare systems in a systematic and iterative way. The underlying assumption is that systems with better scores are actually perceived as more useful by the users and therefore are expected to bring more satisfaction, but it is unknown the extent to which this is true. This is an important gap to fill because the ultimate goal of a researcher is to figure out whether final users will be satisfied or not, or which system is better from that viewpoint.

In this chapter I empirically establish the relationship between system effectiveness and user satisfaction for an array of measures and relevance scales. This allows us not only to interpret evaluation results in practical terms, but also to assess which effectiveness measures and relevance scales are better correlated with user satisfaction. As a side result, this chapter also allows us to quantify the extent to which users agree as to the performance of a system, setting the practical limits of purely system-based evaluations that do not account for user-specific information.

3.1 Effectiveness Measures and Relevance Scales

In the MIREX Audio Music Similarity evaluation experiments the relevance of a document to a query is assessed by human experts and based on two different relevance scales. The Broad scale is an $n_{\mathcal{L}} = 3$ scale where the relevance of a document d is $r_d = 0$ if the document is *not similar* to the query, $r_d = 1$ if it is *somewhat similar* and $r_d = 2$ if it is *very similar* [Jones et al. 2007, Downie et al. 2010]. The Fine scale is an $n_{\mathcal{L}} = 101$ relevance scale, where r_d goes from 0 (not similar at all) to 100 (identical to the query)[1]. In terms of effectiveness

[1] In early editions of MIREX it was defined from 0 to 10, with one decimal digit. With the appropriate normalization, both definitions are equivalent

measures, the official MIREX evaluations only report Cumulative Gain after 5 documents retrieved ($CG@5$) with the $g(\ell) = \ell$ linear gain function (2.8).

3.1.1 User-Oriented Effectiveness Measures

Only two effectiveness scores are reported in MIREX: $CG@5$ with Broad judgments and $CG@5$ with Fine judgments. The assumptions are therefore that the higher the relevance judgment for a document the more likely for it to be perceived as satisfactory; that this perception is independent of the rank at which the document is retrieved and of the relevance of the previous documents; and that the user will actually listen to all five documents in the list. Section 2.5 described a wealth of effectiveness measures based on different user models and making different assumptions. For this chapter I consider all these measures and study their suitability for the AMS task.

However, there are some aspects that we must consider before trying to map system effectiveness onto user satisfaction. Al-Maskari et al. [2007] described a counterintuitive behavior of the $nDCG$ formulation in (2.10). Consider a graded $n_{\mathcal{L}} = 5$ relevance scale and a query for which the set of judgments is $\mathcal{R} = \{1, 0, 0, 0, 0\}$. A system returning the ranking $\mathsf{A} = \langle 1, 2, 3, 4, 5\rangle$ would obtain $nDCG@5 = 1$ because it retrieves the only relevant document at the top. However, only one single document of the lowest possible relevance has been returned, which probably will not help the final user. If the set of judgments were $\mathcal{R} = \{1, 4, 4, 4, 4\}$ then the system would obtain $nDCG@5 < 1$ because it failed to retrieve the highly relevant documents first. Nonetheless, it managed to retrieve all four highly relevant documents after the first rank, which most likely will help the user. In the first case the system was given the highest possible score and was therefore assumed to be perfect, but it is clear that it would have been worse for the user than in the second case.

This counterintuitive behavior of $nDCG$ can be found in other measures; the root of the problem is the consideration of all known judgments for the calculation of the effectiveness score. Consider for example Average Precision at cutoff k as in (2.3). The only way for a system to obtain $AP@k = 1$ is if it only retrieves relevant documents and $k \geq \left|\mathcal{R}^1\right|$. That is, $AP@k$ scores are likely to be low for queries where there are many relevant documents.

For this chapter some of the formulations in Section 2.5 are modified to avoid this behavior. In addition, all λ scores are normalized between 0 and 1 such that a system would obtain $\lambda = 1$ only if it returned a perfect ranking according to the measure's user model. These modifications are discussed below.

Binary Relevance Scale

Precision. No modification is needed in this case because the score depends only on the retrieved documents and it is 1 only when all documents retrieved are relevant. The formulation used is (2.2).

Average Precision. We force the score to be 1 only when all k documents retrieved are relevant, regardless of how many relevant documents there are in the ground truth. Instead of (2.3), this formulation is used:

$$AP@k = \frac{1}{k} \sum_{i=1}^{k} r_{\mathsf{A}_i} \cdot P@i \tag{3.1}$$

Reciprocal Rank. The original formulation in (2.4) is defined for the full list of retrieved documents, so we need to define a variant for a cutoff k. The $RR@k$ score is set to be 0 if none of the top k retrieved documents is relevant:

$$RR@k = \begin{cases} \dfrac{1}{\min\{i : r_{\mathsf{A}_i} = 1\}} & \text{if } \min\{i : r_{\mathsf{A}_i} = 1\} \leq k \\ 0 & \text{otherwise} \end{cases} \tag{3.2}$$

Graded Relevance Scale

Cumulative Gain. This measure only considers the relevance of the retrieved documents, but its upper bound is $k \cdot g(n_{\mathcal{L}} - 1)$. The formulation in (2.8) is therefore normalized to:

$$CG@k = \frac{1}{k} \sum_{i=1}^{k} \frac{g(r_{\mathsf{A}_i})}{g(n_{\mathcal{L}} - 1)} \tag{3.3}$$

Discounted Cumulative Gain. Similarly, no other than the retrieved documents are considered here, but the score needs to be normalized between 0 and 1. Instead of (2.9), the following formulation is used:

$$DCG@k = \frac{\sum_{i=1}^{k} g(r_{\mathsf{A}_i}) \,/ d(i)}{\sum_{i=1}^{k} g(n_{\mathcal{L}} - 1) \,/ d(i)} \tag{3.4}$$

As for the discount function, I use $d(i) = \log_2(i + 1)$ as defined by Burges et al. [2005].

Normalized Discounted Cumulative Gain. This measure is naturally bounded between 0 and 1 but, as mentioned before, it is counterintuitive because it considers all relevance judgments in the ground truth [Al-Maskari et al. 2007]. I do not use it in this chapter, but it should be noted that the above formulation for $DCG@k$ in (3.4) would be the equivalent to a user-oriented $nDCG@k$: the definition of the ideal ranking is changed to be $\langle n_{\mathcal{L}} - 1, n_{\mathcal{L}} - 1, \ldots, n_{\mathcal{L}} - 1 \rangle$, that is, independent of the documents not retrieved.

Q-Measure. Similar to the modifications for $AP@k$ and $DCG@k$, I divide by k rather than by $\left|\mathcal{R}^{>0}\right|$ and define the ideal ranking as $\langle n_{\mathcal{L}} - 1, n_{\mathcal{L}} - 1, \ldots, n_{\mathcal{L}} - 1 \rangle$. Instead of (2.11), the following formulation is used:

$$Q@k = \frac{1}{k} \sum_{i=1}^{k} \mathbb{1}(r_{\mathsf{A}_i} > 0) \, \frac{\sum_{j=1}^{i} \mathbb{1}\left(r_{\mathsf{A}_j} > 0\right) + \beta \cdot \sum_{j=1}^{i} g\left(r_{\mathsf{A}_j}\right)}{i + \beta \cdot \sum_{j=1}^{i} g(n_{\mathcal{L}} - 1)} \tag{3.5}$$

Rank-Biased Precision. The first modification necessary for RBP is that we have to compute it for a cutoff $k = 5$ rather than for all $n_{\mathcal{D}}$ documents as in (2.12). The score is then divided by the $RBP@k$ obtained with the ideal ranking as defined above, such that a system retrieving all documents with maximum relevance would obtain $RBP@k = 1$. The final formulation is as follows:

$$RBP@k = \frac{\sum_{i=1}^{k} g(r_{\mathsf{A}_i}) \cdot p^{i-1}}{\sum_{i=1}^{k} g(n_{\mathcal{L}} - 1) \cdot p^{i-1}} \tag{3.6}$$

Note that this formulation is the same as $DCG@k$ as defined in (3.4) but with the geometric discount function $d(i) = 1/p^{i-1}$ instead of the logarithmic $d(i) = \log_2(i + 1)$. Next, we consider the expected number of documents seen by the user [Moffat and Zobel 2008]:

$$(1-p)\sum_{i=1}^{\infty} i \cdot p^{i-1} = \frac{1}{1-p}$$

Because the MIREX AMS evaluation assumes the user sees as many as 5 documents, I set the persistence parameter to $p = 0.8$ so that we have $1/(1-p) = 5$.

Expected Reciprocal Rank. Similarly, ERR is defined in (2.14) for the full list of documents, but we need to compute the score at the cutoff $k = 5$. In this case I also divide by the $ERR@k$ score obtained by a fully ideal ranking $\langle n_{\mathcal{L}} - 1, n_{\mathcal{L}} - 1, \ldots, n_{\mathcal{L}} - 1\rangle$. As in (2.13), the probability that the user is satisfied with a document in the ideal ranking is:

$$p' = \frac{g(n_{\mathcal{L}} - 1)}{g(n_{\mathcal{L}} - 1) + 1}$$

The probability that the user starts from the top of the ideal ranking and stops at rank i is:

$$p' \cdot \prod_{j=1}^{i-1} 1 - p' = p'(1-p')^{i-1}$$

Plugged into (2.14) this gives us the $ERR@k$ score of the ideal ranking. Using it to normalize the score of an arbitrary system, the final formulation of $ERR@k$ used in this chapter is:

$$ERR@k = \frac{\sum_{i=1}^{k} \frac{1}{i}\left(p_{\mathsf{A}_i} \prod_{j=1}^{i-1} 1 - p_{\mathsf{A}_j}\right)}{p' \sum_{i=1}^{k} \frac{1}{i}(1-p')^{i-1}} \tag{3.7}$$

Graded Average Precision. The $GAP@k$ formulation in (2.16) is already bounded between 0 and 1, but similarly to $AP@k$ and $Q@k$ it uses the full set of judgments to normalize. In this case I also consider a fully ideal ranking with k documents of relevance $n_{\mathcal{L}} - 1$. The denominator in (2.16) is changed accordingly to reflect the fact that the best an arbitrary system can do is retrieve those k documents:

$$GAP@k = \frac{\sum_{i=1}^{k} \mathrm{E}[P@i]}{|\mathcal{R}^{n_{\mathcal{L}}-1}| \sum_{\ell=1}^{n_{\mathcal{L}}-1} p_\ell} = \frac{1}{k}\sum_{i=1}^{k} \mathrm{E}[P@i] \tag{3.8}$$

In the absence of data, a uniform distribution $p_\ell = 1/(n_{\mathcal{L}} - 1)$ is used as suggested by Robertson et al. [2010].

Average Dynamic Recall. The case of $ADR@k$ is more restrictive than in the other measures. We need to consider again a fully ideal ranking with k documents of relevance $n_{\mathcal{L}} - 1$ to obtain a score of 1. The problem is that when we compute how many of the top i retrieved documents are in the ideal ranking, we only accept those with maximum relevance $n_{\mathcal{L}} - 1$, and any other document with relevance $r_d < n_{\mathcal{L}} - 1$ will not contribute to the final score. That is, we are forcing systems to retrieve only highly relevant documents, so this user-oriented variation becomes very restrictive. The formulation used in this chapter is:

$$ADR@k = \frac{1}{k}\sum_{i=1}^{k} \frac{\left|\left\{\mathsf{A}_j \in \mathsf{A} : j \le i \wedge r_{\mathsf{A}_j} = n_{\mathcal{L}} - 1\right\}\right|}{i} = \frac{1}{k}\sum_{i=1}^{k} \frac{1}{i}\sum_{j=1}^{i} \mathbb{1}\left(r_{\mathsf{A}_j} = n_{\mathcal{L}} - 1\right) \tag{3.9}$$

Expected Discounted Cumulative Gain. The final measure considered in this chapter is a variation of $ERR@k$ as in (3.7) where a document at rank i contributes $g(r_{\mathsf{A}_i})$ rather than $1/i$. This is therefore a version of $DCG@k$ where the discount applied to a document actually depends on the relevance of all documents retrieved before rather than just on its rank. The exact formulation is:

$$EDCG@k = \frac{\sum_{i=1}^{k} g(r_{\mathsf{A}_i}) \left(p_{\mathsf{A}_i} \prod_{j=1}^{i-1} 1 - p_{\mathsf{A}_j} \right)}{p' \sum_{i=1}^{k} g(n_{\mathcal{L}} - 1) \, (1 - p')^{i-1}} \tag{3.10}$$

3.1.2 Relevance Scales

This chapter studies several relevance scales combined with all eleven user-oriented effectiveness measures described above. Because they are the scales annually used in MIREX, the original $n_{\mathcal{L}} = 3$ Broad and $n_{\mathcal{L}} = 101$ Fine scales are employed. I further consider three graded scales and four binary scales by using thresholds. The judgments for these seven scales were artificially created from the original Fine judgments made for MIREX. For the artificial graded scales, the $[0, 100]$ range is evenly split in $n_{\mathcal{L}}$ intervals of length $101/n_{\mathcal{L}}$. For instance, in the $n_{\mathcal{L}} = 3$ case, a document d has relevance 0 if $r_d < 34$, relevance 1 if $34 \leq r_d < 67$ and relevance 2 if $r_d \geq 67$. For the artificial binary scale, the $[0, 100]$ range is split in $n_{\mathcal{L}} = 2$ intervals based on a threshold ℓ_{min} so that an arbitrary document d is considered relevant only if $r_d \geq \ell_{min}$. Although both the original Broad and the artificial $n_{\mathcal{L}} = 3$ scales have 3 possible levels of relevance, I should note that the final judgments are not necessarily the same because the latter is computed from the Fine scale and the former is independent of it (see Figure 2 in [Downie et al. 2010]).

Table 3.1 shows all combinations of effectiveness measures and relevance scales considered in this chapter (marked with x). The binary measures $P@5$ (2.2), $AP@5$ (3.1) and $RR@5$ (3.2) are combined with all four artificial binary scales. All graded measures based on a gain function are combined with both the linear $g(\ell) = \ell$ and the exponential $g(\ell) = 2^\ell - 1$ gain functions, named Λ_l and Λ_e respectively. All these measures are combined with the original Broad scale and with the three artificial graded scales. The original Fine scale is combined only with the Λ_l versions, because the maximum gain score in the Λ_e versions is extremely high and the measure has virtually no sensitivity for relatively high levels (e.g. consider $g(90) = 2^{90} - 1 \approx 1.24 \cdot 10^{27}$ versus $g(85) = 2^{85} - 1 \approx 3.87 \cdot 10^{25}$). The $ADR@5$ (3.9) measure is not combined with the Fine scale either, as it would only accept retrieved documents with relevance $r_d = 100$, which is very hardly ever the case because assessors seldom assign such an extreme relevance level to begin with (see Figure 7.1).

Finally, some combinations are ignored because they are equivalent to others. All Λ_e measures with binary scales are equivalent to their Λ_l counterparts because $g(1) = 2^1 - 1 = 1$. Similarly, under a binary relevance scale $CG@5$ (3.3) is equivalent to $P@5$ (2.2); and both $Q@5$ (3.5) and $GAP@5$ (3.8) are equivalent to $AP@5$ (3.1). In summary, as many as 95 different combinations are studied.

Measure	Original		Artificial Graded			Artificial Binary			
	Broad	Fine	$n_{\mathcal{L}}=3$	$n_{\mathcal{L}}=4$	$n_{\mathcal{L}}=5$	$\ell_{min}=20$	$\ell_{min}=40$	$\ell_{min}=60$	$\ell_{min}=80$
P@5						x	x	x	x
AP@5						x	x	x	x
RR@5						x	x	x	x
CG_l@5	x	x	x	x	x	P@5	P@5	P@5	P@5
CG_e@5	x		x	x	x	P@5	P@5	P@5	P@5
DCG_l@5	x	x	x	x	x	x	x	x	x
DCG_e@5	x		x	x	x	DCG_l@5	DCG_l@5	DCG_l@5	DCG_l@5
$EDCG_l$@5	x	x	x	x	x	x	x	x	x
$EDCG_e$@5	x		x	x	x	$EDCG_l$@5	$EDCG_l$@5	$EDCG_l$@5	$EDCG_l$@5
Q_l@5	x	x	x	x	x	AP@5	AP@5	AP@5	AP@5
Q_e@5	x		x	x	x	AP@5	AP@5	AP@5	AP@5
RBP_l@5	x	x	x	x	x	x	x	x	x
RBP_e@5	x		x	x	x	RBP_l@5	RBP_l@5	RBP_l@5	RBP_l@5
ERR_l@5	x	x	x	x	x	x	x	x	x
ERR_e@5	x		x	x	x	ERR_l@5	ERR_l@5	ERR_l@5	ERR_l@5
GAP@5	x	x	x	x	x	AP@5	AP@5	AP@5	AP@5
ADR@5	x		x	x	x	x	x	x	x

Table 3.1: All 95 combinations of effectiveness measures and relevance scales studied (marked with x), and equivalent combinations (e.g. GAP@5 is the same as AP@5 with a binary scale).

3.2 Experimental Design

An experiment with actual users was designed such that it allows us to map system effectiveness onto user satisfaction. In this context, I consider the two situations an IR researcher is often faced with. In the first scenario we want to evaluate a single system (i.e. absolute λ scores) to assess how well it will satisfy users. In the second scenario we want to compare two systems (i.e. relative $\Delta\lambda$ scores) to assess which one will provide more user satisfaction.

Subjects were presented with different examples, each containing a query clip q and two ranked lists of five results each, as if retrieved by two different AMS systems A and B. The effectiveness scores $\lambda_{q,\mathsf{A}}$ and $\lambda_{q,\mathsf{B}}$ were known but withheld to subjects [Sanderson et al. 2010, Allan et al. 2005]. They had to listen to the clips and then select one of the following options: system A provided better results, system B did, they both provided *good* results, or they both returned *bad* results (see Figure 3.1). From these options we can differentiate four user preferences:

- **Positive preference.** The subject selected the system with *larger* effectiveness.
- **Negative preference.** The subject selected the system with *smaller* effectiveness.
- **Good nonpreference.** The subject indicated both systems are equally *good*.
- **Bad nonpreference.** The subject indicated both systems are equally *bad*.

This design allows us to analyze the results from two different perspectives: the evaluation of a single system and the comparison of two systems. Subjects indicating that both systems are *good* suggest that they are satisfied with both ranked lists. That is, their answer serves as an indication that the effectiveness scores $\lambda_{q,\mathsf{A}}$ and $\lambda_{q,\mathsf{B}}$ measured for those systems translates into user satisfaction. If, on the other hand, they indicate that both systems are *bad*, we can infer that those effectiveness scores do not translate into user satisfaction.

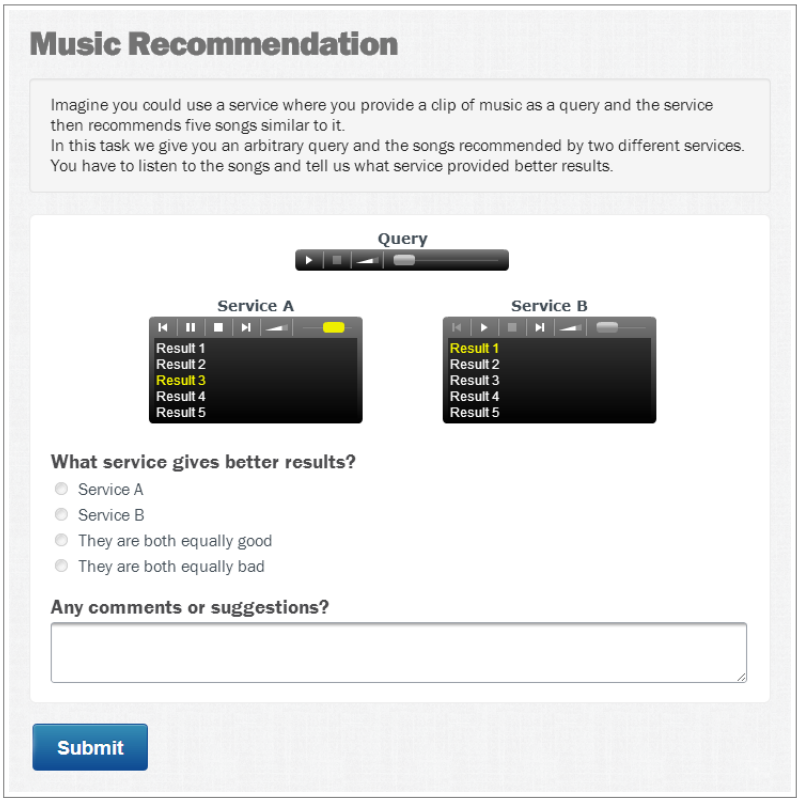

Figure 3.1: Task template used in the experiment.

Subjects indicating a preference for one ranked list over the other one suggest that there is a difference between them large enough to be noted. That is, their answer serves as an indication that the difference in effectiveness $\Delta\lambda_{q,\mathsf{AB}}$ between the systems translates into users being more satisfied with one system than with the other. Whether they agree with the evaluation or not depends on which of the two systems they prefer.

3.2.1 Data

The relevance judgments collected for the MIREX AMS task in 2007, 2009, 2010, 2011 and 2012 were used. They comprise a total of 22,074 relevance judgments across 439 queries[2]. Each of these judgments consists of the Broad and Fine labels assigned to a document for a particular query. The full $[0, 1]$ range for effectiveness scores was split in 10 equally-sized bins $\beta \in \{[0, 0.1), [0.1, 0.2), \ldots, [0.9, 1]\}$, such that the $|\Delta\lambda_{q,\mathsf{AB}}|$ score of an arbitrary example falls under one of these bins.

Next, we need to come up with examples such that for every $(\Lambda, \mathcal{L}, \beta)$ combination we have a sufficiently large number of examples to compute a realiable mapping. Using the documents retrieved by the actual MIREX AMS systems is unfeasible because it would limit our chances of having enough data for all cases. Instead, artificial examples were created

[2] I excluded queries and documents for which I did not have the actual audio files. Among others, this effectively excluded all judgments from MIREX 2006.

using the known judgments to get at least 200 examples per bin [Sanderson et al. 2010, Allan et al. 2005]. Because there are 10 bins, we need at least 2,000 examples in the best case. The easiest method to accomplish this is a random sampling algorithm that randomly creates 2,000 examples. However, this method will most likely generate biased distributions for $|\Delta\lambda|$ in some $(\Lambda, \mathcal{L})$ combinations where data is rather scarce. For example, human assessors very rarely assign $r_d < 10$ and $r_d > 90$ with the Fine scale, so it is very hard to find examples for which $|\Delta\lambda| \in [0.9, 1]$ [Urbano et al. 2012]. Instead, I used an iterative greedy algorithm that at each iteration selects the $(\Lambda, \mathcal{L}, \beta)$ combination with the least number of examples so far. If a new artificial example can be created for that combination, then that example is saved and added to the respective bin it falls under for all other $(\Lambda, \mathcal{L})$ combinations. If the available data is not enough to create another such example, then the $(\Lambda, \mathcal{L}, \beta)$ combination is ignored from that point on. This algorithm iterates until there are at least 200 examples per combination or there are no more possibilities to achieve that with the known judgments at our disposal.

This algorithm was run with all 22,074 judgments, and a total of 4,115 examples were artificially created to accommodate 200 examples per $(\Lambda, \mathcal{L}, \beta)$ combination. However, in 5 of the 950 combinations there was not enough data to create 200 examples: $(CG_e@k, \text{Broad}, [0.5, 0.6))$, $(CG_e@k, \text{Broad}, [0.7, 0.8))$, $(CG_e@k, n_{\mathcal{L}} = 3, [0.5, 0.6))$, $(CG_e@k, n_{\mathcal{L}} = 3, [0.7, 0.8))$ and $(CG_l@k, n_{\mathcal{L}} = 4, [0.7, 0.8))$ ended up with 165, 178, 166, 141 and 178 examples respectively. Additionally, $P@k$ can not accommodate examples for some bins because of its very formulation: all $P@5$ scores are a multiple of 0.2, so it is impossible to create an example such that $\Delta P@5$ is 0.3, for instance. Similarly, it is impossible to create examples such that $\Delta RR@5 \in [0.4, 0.5)$.

In summary we have 4,115 examples, and all but five $(\Lambda, \mathcal{L}, \beta)$ possible combinations contained at least 200 examples, with a final average of over 400 examples per combination. Across all 4,155 examples we find 432 unique queries and 5,636 unique documents, covering the wide range of genres and artists in the MIREX document set. All 4,115 examples were different from each other.

3.2.2 Procedure

Preferences for all 4,115 examples were collected via crowdsourcing. Previous work by Lee [2010] and Urbano et al. [2010b] demonstrated that music similarity judgments gathered with crowdsourcing platforms are very similar to those collected with experts, with fast turnaround and at a low cost. Another advantage of crowdsourcing for our experiment is that it offers a large and diverse pool of subjects around the globe. Using a controlled group of students or experts would probably bias our results, but using a diverse pool of workers allows us to draw conclusions that should generalize to the wider population of users.

However, using crowdsourcing has other issues. The quality of judgments via crowdsourcing can be questioned because some workers are known to produce spam answers and others provide careless answers seeking profit without actually doing the task. I decided to use the platform Crowdflower[3] to gather preferences, which delegates the work to other platforms such as Amazon Mechanical Turk. It also provides a quality control layer at the

[3] http://www.crowdflower.com

process level [Urbano et al. 2011b] that separates good from bad workers by means of trap examples [Le et al. 2010, Sanderson et al. 2010]: some of the examples shown to workers have known answers (provided by us) that are used to estimate worker quality. Workers that show low quality on the trap examples are rejected, and those that show high agreement are allowed to participate. Crowdflower was provided with 20 such trap examples (5 for each of the four answers), assigning each of them a subjective level of difficulty based on the answers by two experts. Most of them were fairly easy to answer.

3.2.3 Task Design

Figure 3.1 shows the task template used. A first section listed the task instructions, and then a Flash player permitted subjects to listen to the query clip. Below, they could find the two ranked lists of 5 results each, followed by radio buttons to select their answer. Finally, a textbox was provided for subjects to optionally leave feedback. All audio clips were uploaded to a private server, and served upon request. The order in which examples are shown to workers is random, as is the assignment of the ranked lists as system A or system B. Also, the maximum number of answers by a single worker was limited to 100, minimizing the possible bias due to super-workers that do most of the work.

All answers were collected in nine batches of nearly 500 examples each. Lee [2010] collected similarity judgments paying $0.20 for 15 query-document pairs, while Urbano et al. [2010b] collected preference judgments paying $0.02 for each query-document-document. In both studies workers were therefore paid approximately $0.007 per audio clip. Music-related tasks are known to be enjoyable by workers, and given that quality does not significantly degrade when decreasing wages [Mason and Watts 2009], I decided to pay $0.03 for each example, leading to approximately $0.003 per clip. The total spent was approximately $250 after fees to Crowdflower.

3.3 Results

The nine batches were completed in less than 24 hours. A total of 547 unique workers from 21 different crowdsourcing platforms participated in the experiment. These workers provided a grand total of 11,042 answers, from which Crowdflower accepted 9,373 (85%) as trustworthy; the extra answers are due to repeatedly showing trap examples to workers. Only 175 workers were responsible for these trusted answers, so 372 workers (68%) were rejected. The average trust on these 175 workers, as computed by Crowdflower [Le et al. 2010], ranges from 73% to 100%, with an average of 90%. Discarding answers to the trap questions, the final 4,115 answers were given by 113 unique workers, with an average of 36 answers per worker.

3.3.1 Evaluating a Single System

For 2,025 of the 4,115 examples (49%) we received a nonpreference (i.e. subjects judged both systems as equally good or bad). Therefore, we have 4,050 ranked lists that subjects judged as satisfactory or unsatisfactory. Figure 3.2 shows the log-scaled distributions of absolute λ scores for these lists. As can be seen, the wide range of scores is covered, following somewhat

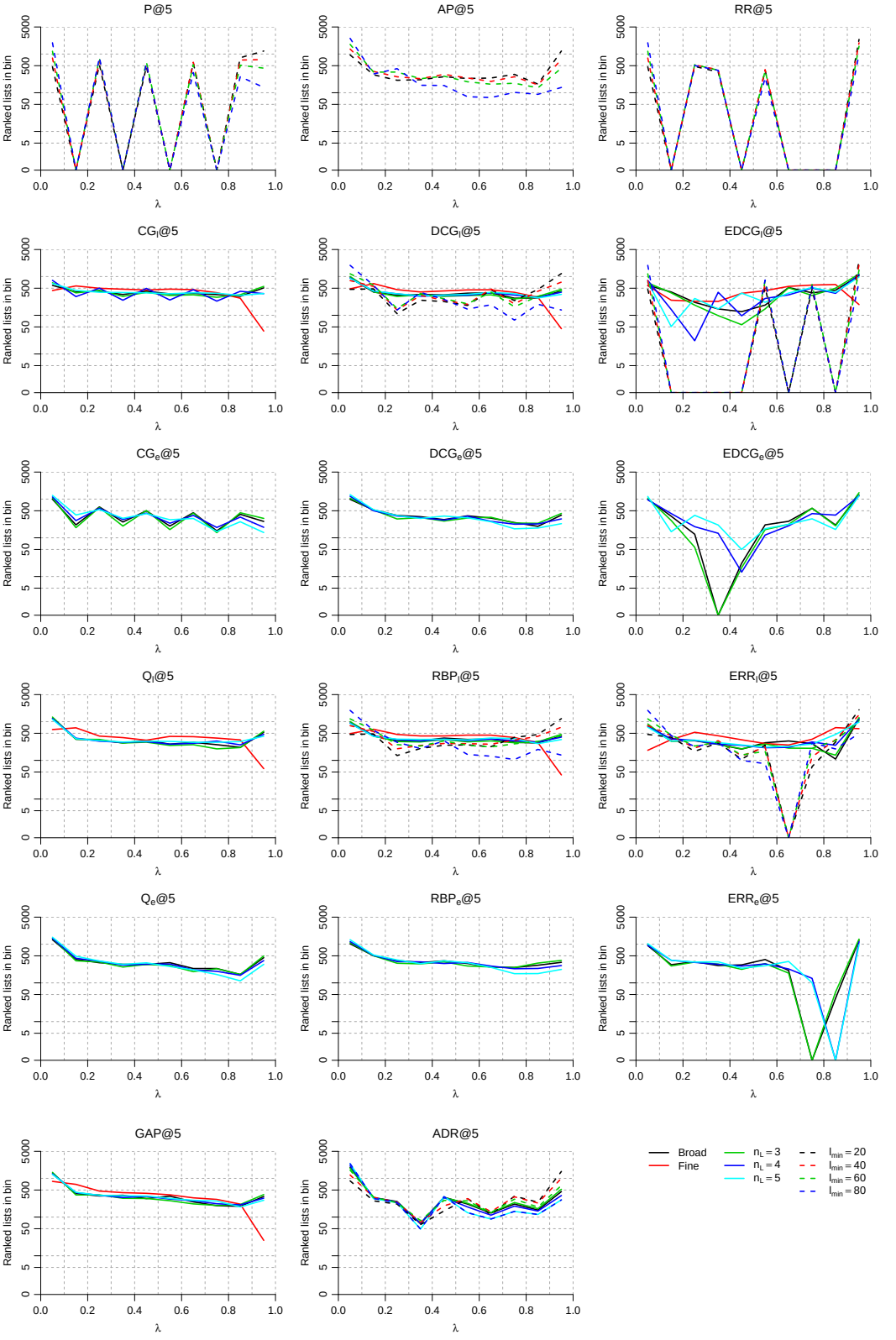

Figure 3.2: Log-scaled distributions of absolute λ scores in all 2,025 examples with nonpreferences.

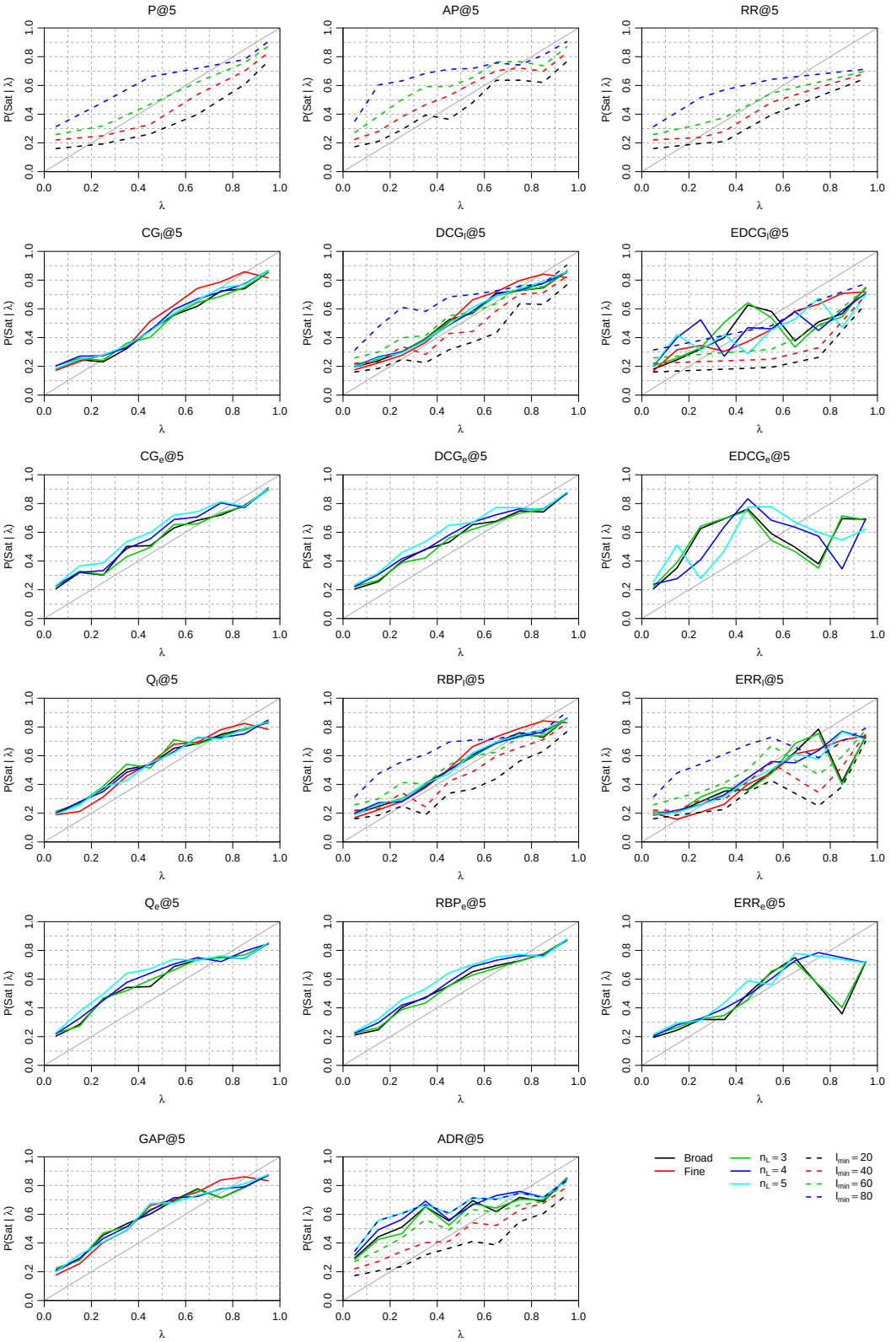

Figure 3.3: Probability of user satisfaction given a λ score in all 2,025 examples with nonpreferences.

uniform distributions with about 400 ranked lists per bin; again, there are some measures for which it is not possible to produce examples that fall under some particular bins. The number of good and bad nonpreferences was similar too: 1,056 and 969.

Figure 3.3 shows the fraction of examples within bins that were judged as satisfactory. Let Sat be a random variable that equals 1 if the user is satisfied and 0 if not. The proportions in the figure allow us to take a frequentist approach to the probability that $Sat = 1$ conditional on the observed λ score: $P(Sat = 1|\Lambda = \lambda)$, or simply $P(Sat|\lambda)$. For instance, if a system obtains $P@5 = 0.4$ when using the $\ell_{min} = 80$ scale, there is a 0.6 probability that an arbitrary user finds the results satisfactory. Similarly, $AP@5 \geq 0.1$ is required for half the users to be satisfied.

As expected, there is a very tight positive correlation between effectiveness and user satisfaction; the relationship appears to be nearly linear in many cases. The effect of the threshold ℓ_{min} is clear in the binary measures: the larger ℓ_{min} the larger $P(Sat)$, because documents need to be very similar to the query to be considered relevant and λ only increases with very similar documents. Comparing the linear Λ_l and exponential Λ_e measures, it is clear that the Λ_e measures tend to underestimate satisfaction (they are quite above the diagonal). This is an artifact of the exponential gain function: highly relevant documents are assumed to be much more satisfactory relative to the others than in the linear gain function. For example, in a linear gain function two documents with relevance $r_d = 2$ and $r_{d'} = 4$ contribute 2 and 4 to $CG_l@5$, but they contribute $2^2 - 1 = 3$ and $2^4 - 1 = 15$ to $CG_e@5$. That is, the relative contribution is an order of magnitude larger. The measure ends up overestimating the contribution of highly relevant documents and underestimating the contribution of the rest. As a result, the expected satisfaction due to mid-relevant documents is underestimated because it is not as large as it could *supposedly* be. This effect is also clear comparing the same Λ_e measure with the different graded scales: the larger $n_{\mathcal{L}}$ the larger the underestimation, because documents with the highest relevance within the scale leave much more room for improvement to the other documents.

In general, the measures that best adhere to the diagonal are $CG_l@5$, $DCG_l@5$ and $RBP_l@5$ for the graded scales. However, it is not really a problem that the curves are far apart from the diagonal; it just means that the interpretation of a λ score is not as immediate as we expected it to be. I get to this issue in Chapter 4. For the time being, we are interested in measures and scales that are closer to the expected $P(Sat|0) = 0$ and $P(Sat|1) = 1$. Given the highly subjective notion of similarity, it is expected that different users perceive results differently. If a human assessor makes some relevance judgments and a system gets an effectiveness score $\lambda = 1$, it means that the system provided ideal results, and therefore all users should find them satisfactory. In practice though, that is not the case; some users will disagree to some extent. Some measures account for more information than others and make different assumptions as to how users behave so that effectiveness scores are better correlated with user satisfaction. It does not really matter if the relationship is linear or not, as long as the measure is not biased towards the endpoints and it gets closer to the expected 0 and 1. Measures that better achieve this are the ones that we can trust the most when generalizing results to the wider population of users because they are less sensitive to user variations.

Measure	Original		Artificial Graded			Artificial Binary			
	Broad	Fine	$n_{\mathcal{L}}=3$	$n_{\mathcal{L}}=4$	$n_{\mathcal{L}}=5$	$\ell_{min}=20$	$\ell_{min}=40$	$\ell_{min}=60$	$\ell_{min}=80$
P@5						*0.1515*	***0.1494***	*0.158*	*0.1893*
AP@5						0.1562	**0.1513**	0.1671	0.2154
RR@5						0.2268	**0.2229**	0.2305	0.2507
CG_l@5	*0.1159*	0.1284	0.1293	*0.1232*	***0.111***				
CG_e@5	**0.1175**		*0.1283*	0.1297	0.13				
DCG_l@5	**0.1202**	0.1293	0.1314	0.1248	0.1215	*0.1515*	*0.1494*	*0.158*	*0.1893*
DCG_e@5	**0.1226**		0.1335	0.1347	0.14				
$EDCG_l$@5	**0.1726**	0.1843	0.1784	0.203	0.2097	0.2371	0.2171	0.2085	0.2237
$EDCG_e$@5	**0.2145**		0.2237	0.226	0.2741				
Q_l@5	0.1327	0.1535	0.141	**0.1296**	0.1304				
Q_e@5	**0.1306**		0.1448	0.1409	0.1425				
RBP_l@5	0.1202	*0.1202*	0.1314	0.1239	**0.1195**	*0.1515*	*0.1494*	*0.158*	*0.1893*
RBP_e@5	**0.127**		0.1347	0.1355	0.1397				
ERR_l@5	**0.1812**	0.1831	0.1857	0.1905	0.1953	0.1927	0.1859	0.1916	0.2176
ERR_e@5	**0.193**		0.1979	0.1978	0.2043				
GAP@5	0.1224	0.1221	0.1346	0.128	**0.1189**				
ADR@5	0.1874		0.181	0.2042	0.2226	0.1747	**0.1654**	0.1781	0.2226

Table 3.2: Bias in $P(Sat|\lambda)$ at the endpoints $\lambda = 0$ and $\lambda = 1$ as per (3.11) (lower is better). Best per measure in bold, best per scale in italics.

The bias between the expected and actual behavior at the endpoints was measured by computing a rooted mean squared distance as follows:

$$\sqrt{\frac{P(Sat|0)^2 + (1 - P(Sat|1))^2}{2}} \tag{3.11}$$

Table 3.2 shows the results for all measure-scale combinations. The $\ell_{min} = 40$ scale performs the best among the binary scales, and the Broad scale dominates among the graded scales. Similarly, the measures that behave the best are CG@5, DCG@5, RBP@5 and GAP@5. In general, $P(Sat|0) \approx 0.2$, with some measures having a bias as high as 0.4. This means that about 20% of users will find the results satisfactory even when the result of the evaluation is $\lambda = 0$. On the other hand, when $\lambda = 1$ it is expected that between 10% and 20% of users are not satisfied despite the evaluation suggested ideal results.

3.3.2 Comparing Two Systems

For 2,090 of the 4,115 examples (51%) we did receive a preference (i.e. subjects indicated that one system provided better results than the other one). Subjects preferred system A 1,019 times and system B 1,071 times, that is, about the same as expected. Whether those user preferences were positive or negative (i.e. agreeing with the sign of $\Delta\lambda_{q,\mathsf{AB}}$ or not), depends on the combination of measure and scale used. Let Agr be a random variable that equals 1 if the subject agrees with the evaluation and does prefer the system with higher λ score as measured with a test collection (i.e. a positive preference), -1 if she disagrees (i.e. a negative preference), and 0 if she can not decide which one is better (i.e. a nonpreference).

Measure	Original		Artificial Graded			Artificial Binary			
	Broad	Fine	$n_{\mathcal{L}}=3$	$n_{\mathcal{L}}=4$	$n_{\mathcal{L}}=5$	$\ell_{min}=20$	$\ell_{min}=40$	$\ell_{min}=60$	$\ell_{min}=80$
P@5						0.5842	0.5945	0.5999	**0.5733**
AP@5						0.5696	0.562	*0.5586*	***0.5101***
RR@5						0.6483	**0.6382**	0.6508	0.6513
CG_l@5	0.5688	***0.4653***	0.5814	0.5348	0.5265				
CG_e@5	0.5309		0.5438	0.5241	***0.5014***				
DCG_l@5	0.531	**0.4733**	0.542	0.5304	0.5203	0.5529	*0.5579*	0.5756	0.5533
DCG_e@5	0.5313		0.544	0.5287	**0.512**				
$EDCG_l$@5	0.563	**0.5172**	0.5877	0.566	0.5527	0.6139	0.5999	0.623	0.6424
$EDCG_e$@5	**0.5942**		0.6148	0.614	0.5963				
Q_l@5	0.5432	**0.4904**	0.5486	0.5387	0.5303				
Q_e@5	0.5387		0.5473	0.5325	**0.5186**				
RBP_l@5	0.5274	**0.4719**	0.539	0.5276	0.5176	0.551	0.5614	0.5731	0.5489
RBP_e@5	0.5272		0.5387	0.5247	**0.5075**				
ERR_l@5	0.5672	**0.515**	0.5783	0.5683	0.5595	0.5587	0.5743	0.6081	0.6044
ERR_e@5	0.5741		0.5906	0.585	**0.5725**				
GAP@5	*0.5229*	**0.4665**	*0.5333*	*0.5166*	0.5105				
ADR@5	0.5827		0.5933	0.5837	0.5649	**0.5622**	0.5756	0.5924	0.5649

Table 3.3: Distance between 1 and $P(Agr = 1|\Delta\lambda)$ (lower is better). Best per measure in bold, best per scale in italics.

Positive Preferences

Figure 3.4 shows the fraction of examples within bins for which users preferred the supposedly better system according to $\Delta\lambda_{q,\mathsf{AB}}$. The proportions allow us again to follow a frequentist approach to the probability that $Agr = 1$ conditional on the observed $\Delta\lambda$ score: $P(Agr = 1|\Delta\Lambda = \Delta\lambda)$, or simply $P(Agr = 1|\Delta\lambda)$. For instance, if comparing two systems we get $\Delta RBP_e@5 = 0.2$ there is a 0.3 probability that an arbitrary user will agree as to which system is better. Similarly, when using the Fine scale $\Delta ERR_l@5 > 0.4$ is required for half the users to agree.

Ideally we would want users to show a preference for the better system whenever we observe an effectiveness difference in the evaluation, regardless of how large this difference is. That is, we always expect $P(Agr = 1) = 1$ regardless of $\Delta\lambda$. Figure 3.3 and Table 3.2 showed that there is some level of disagreement κ among users, so we should actually expect $P(Agr = 1) = 1 - \kappa$. But there is a very tight positive correlation with $\Delta\lambda$ instead: the larger the difference in effectiveness the more likely for users to prefer the supposedly better system. The relationship is nearly linear again, but we can observe very clear differences among scales. To quantitatively assess which measures and scales are closer to the ideal $P(Agr = 1) = 1$, the rooted mean squared distance between the distributions and the top $y = 1$ axis was measured:

$$\sqrt{\frac{1}{10}\sum_{\beta}\left(1 - P(Agr = 1|\Delta\Lambda \in \beta)\right)^2} \tag{3.12}$$

Table 3.3 shows the results for all measure-scale combinations. The $\ell_{min} = 80$ scale is slightly better than the other binary scales except for $\Delta RR@5 > 0.5$. In the graded case, the Fine scale is clearly superior for all Λ_l measures, and the artificial $n_{\mathcal{L}} = 5$ scale is

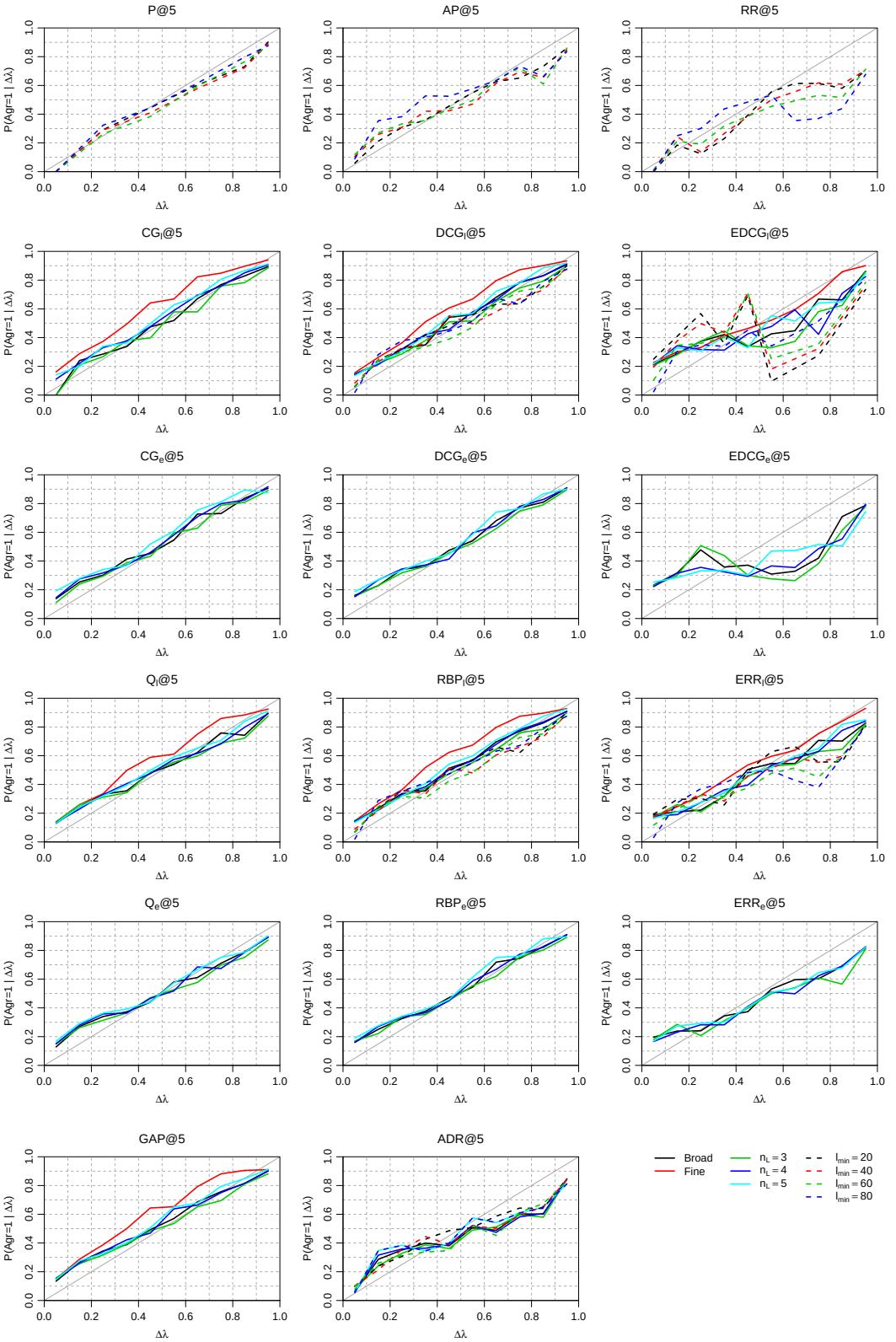

Figure 3.4: Probability of positive user preference given a $\Delta\lambda$ score in all 4,115 examples.

superior for the Λ_e measures. The measures that behave best overall are again $CG@5$, $DCG@5$, $RBP@5$ and $GAP@5$.

Negative Preferences

Figure 3.5 shows the fraction of examples within bins for which users preferred the supposedly worse system according to $\Delta\lambda$. Once again these proportions allow us to follow a frequentist approach to the probability that $Agr = -1$ conditional on the observed $\Delta\lambda$ score: $P(Agr = -1|\Delta\Lambda = \Delta\lambda)$, or simply $P(Agr = -1|\Delta\lambda)$. As an example, if the effectiveness difference between two systems is $\Delta ERR_l@5 = 0.4$ with the Fine scale, 5% of the users disagree and prefer the supposedly least effective system. Similarly, when using the $n_{\mathcal{L}} = 5$ graded scale we need $\Delta GAP@5 > 0.3$ to have less than 5% of users prefer the worse system. Considering both Figure 3.4 and Figure 3.5 together, we see that users are increasingly undecided as differences in effectiveness get smaller. In general, the probability that the user can not decide for one system or the other is $P(Agr = 0|\Delta\lambda) = 1 - P(Agr = 1|\Delta\lambda) - P(Agr = -1|\Delta\lambda)$.

Ideally we would want users to never show a preference for the supposedly worse system, no matter how small the effectiveness difference in the evaluation is. That is, we always expect $P(Agr = -1) = 0$ regardless of $\Delta\lambda$. But as we discussed before, there is some level of disagreement among users, so we should actually expect $P(Agr = -1) = \kappa$. Instead we find a slight negative correlation with $\Delta\lambda$: the smaller the difference in effectiveness the more likely for users to prefer the supposedly worse system.

To quantitatively assess which measures and scales are closer to the ideal $P(Agr = -1) = 0$, the rooted mean squared distance between the distributions and the bottom $y = 0$ axis is measured:

$$\sqrt{\frac{1}{10}\sum_{\beta} P(Agr = 1|\Delta\Lambda \in \beta)^2} \tag{3.13}$$

Table 3.4 shows the results for all measure-scale combinations. The Fine scale seems again superior for the Λ_l measures, while the Broad scale is generally the best for the Λ_e measures. Across scales, the measure that behaves the best overall is $CG@5$, followed by $GAP@5$, $DCG@5$ and $RBP@5$.

3.4 Considering Priors

Section 3.3 provides a guide for the interpretation of evaluation results from the point of view of user satisfaction. It allows researchers to assess how well their systems are expected to satisfy users and what to expect from them when comparing two systems. From Figure 3.3 researchers who pursue a goal such as satisfying over 80% of the users are now able to set a threshold in λ. For example, when using the Fine scale we see that it is required $GAP@5 > 0.7$ or $Q_l@5 > 0.8$. Intuitively then, we would pay more attention to $GAP@5$ because it requires our systems to be less effective. Similarly, from Figure 3.4 we may set a threshold in $\Delta\lambda$ to meet the goal of having 50% or more users agree with the evaluation result. For example, $\Delta Q_l@5 > 0.35$ with the Fine scale is required, while $\Delta Q_l@5 > 0.5$ is with the Broad scale. Intuitively then, we would prefer to use the Fine scale because it requires smaller differences and it should therefore be easier to achieve our goal.

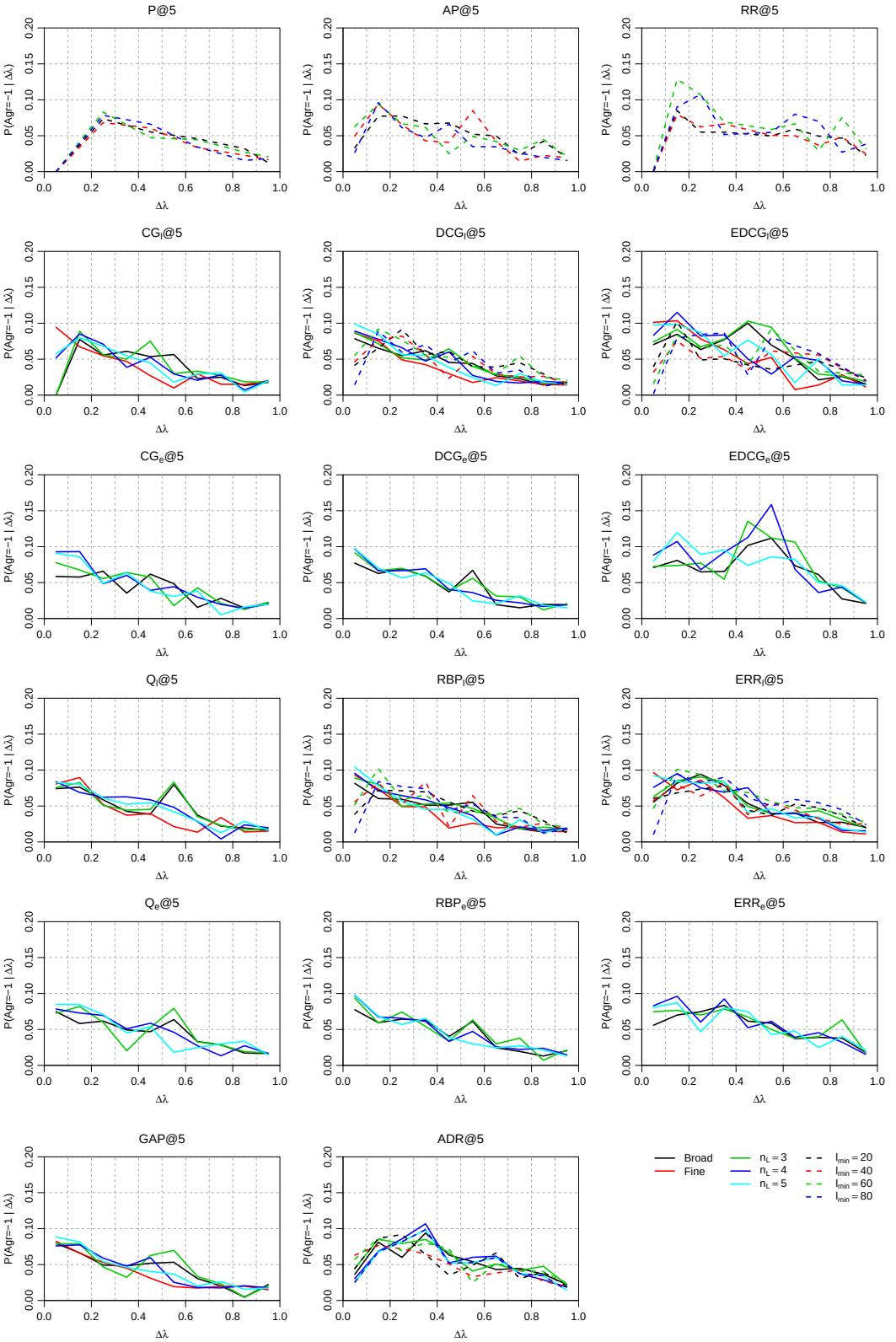

Figure 3.5: Probability of negative user preference given a $\Delta\lambda$ score in all 4,115 examples.

Measure	Original		Artificial Graded			Artificial Binary			
	Broad	Fine	$n_{\mathcal{L}}=3$	$n_{\mathcal{L}}=4$	$n_{\mathcal{L}}=5$	$\ell_{min}=20$	$\ell_{min}=40$	$\ell_{min}=60$	$\ell_{min}=80$
P@5						*0.0443*	***0.0414***	*0.0452*	*0.0452*
AP@5						0.0547	0.054	0.054	**0.0491**
RR@5						0.0521	**0.0516**	0.0732	0.0657
CG_l@5	**0.0453**	0.046	*0.0475*	*0.0466*	*0.0467*				
CG_e@5	***0.045***		0.0494	0.0535	0.0516				
DCG_l@5	0.0479	**0.0446**	0.0505	0.0512	0.0518	0.0523	0.0489	0.0547	0.0501
DCG_e@5	**0.0503**		0.0531	0.0526	0.0515				
$EDCG_l$@5	0.0639	0.0607	0.0696	0.0657	0.0648	0.0535	**0.0509**	0.0596	0.0631
$EDCG_e$@5	**0.073**		0.0818	0.0887	0.079				
Q_l@5	0.0519	**0.0473**	0.0534	0.0521	0.0516				
Q_e@5	**0.049**		0.0525	0.0514	0.0524				
RBP_l@5	0.0486	**0.046**	0.0518	0.0516	0.0518	0.0508	0.0513	0.0538	0.0512
RBP_e@5	**0.0495**		0.0535	0.0523	0.0511				
ERR_l@5	0.0573	0.0547	0.0594	0.0597	0.0599	0.0547	**0.0541**	0.0639	0.0628
ERR_e@5	**0.057**		0.0607	0.063	0.0593				
GAP@5	0.0478	***0.0432***	0.0515	0.0481	0.0492				
ADR@5	0.0574		0.06	0.0609	0.0578	0.0573	**0.0522**	0.0587	0.0578

Table 3.4: Distance between 0 and $P(Agr = -1|\Delta\lambda)$ (lower is better). Best per measure in bold, best per scale in italics.

Intuition fails at this point. When making the decision of using the Fine scale instead of the Broad scale because smaller differences are required, we are assuming that both scales are equally likely to produce these ΔQ_l@5 scores. Imagine that ΔQ_l@5 > 0.35 is achieved only 20% of the times with Fine judgments, while ΔQ_e@5 > 0.5 is achieved 40% of the times with Broad: even though the Fine scale requires smaller differences it is less likely to observe those differences to begin with. That is, we must consider the prior probability of observing differences that large.

For each of the nine relevance scales considered throughout this chapter the prior distributions were computed for each measure of interest. A way to proceed would be to compute all possible assignments of relevance that can be made to two lists of five results each, then computing the distribution of $\Delta\lambda$ scores for each measure. However, priors computed this way would not be informative because they would be too unrealistic. It could be the case that we randomly consider a hypothetical system that retrieves ideal results compared with another system that does not retrieve any relevant document at all. While possible in theory, situations like that hardly ever happen in practice; λ scores are generally correlated across systems, so $\Delta\lambda$ is usually small. Instead, the prior distributions were computed by comparing all pairs of actual systems from the MIREX AMS 2007, 2009, 2010, 2011 and 2012 editions. For every pair of systems evaluated for every query, the corresponding $\Delta\lambda$ score was calculated for every $(\Lambda, \mathcal{L})$ combination. This makes a total of 37,450 datapoints per combination of measure and scale.

Figure 3.6 shows the cumulative distribution functions $F_{\Delta\Lambda}(\Delta\lambda) = P(\Delta\Lambda \leq \Delta\lambda)$ for all $(\Lambda, \mathcal{L})$ combinations. For example, with Fine judgments $F_{\Delta CG_l@5}(0.3) = 0.8$, meaning that in 80% of the observations we get ΔCG_l@5 ≤ 0.3 and only 20% of the times we get ΔCG_l@5 > 0.3. As mentioned, it can be seen that $\Delta\lambda$ scores are generally small.

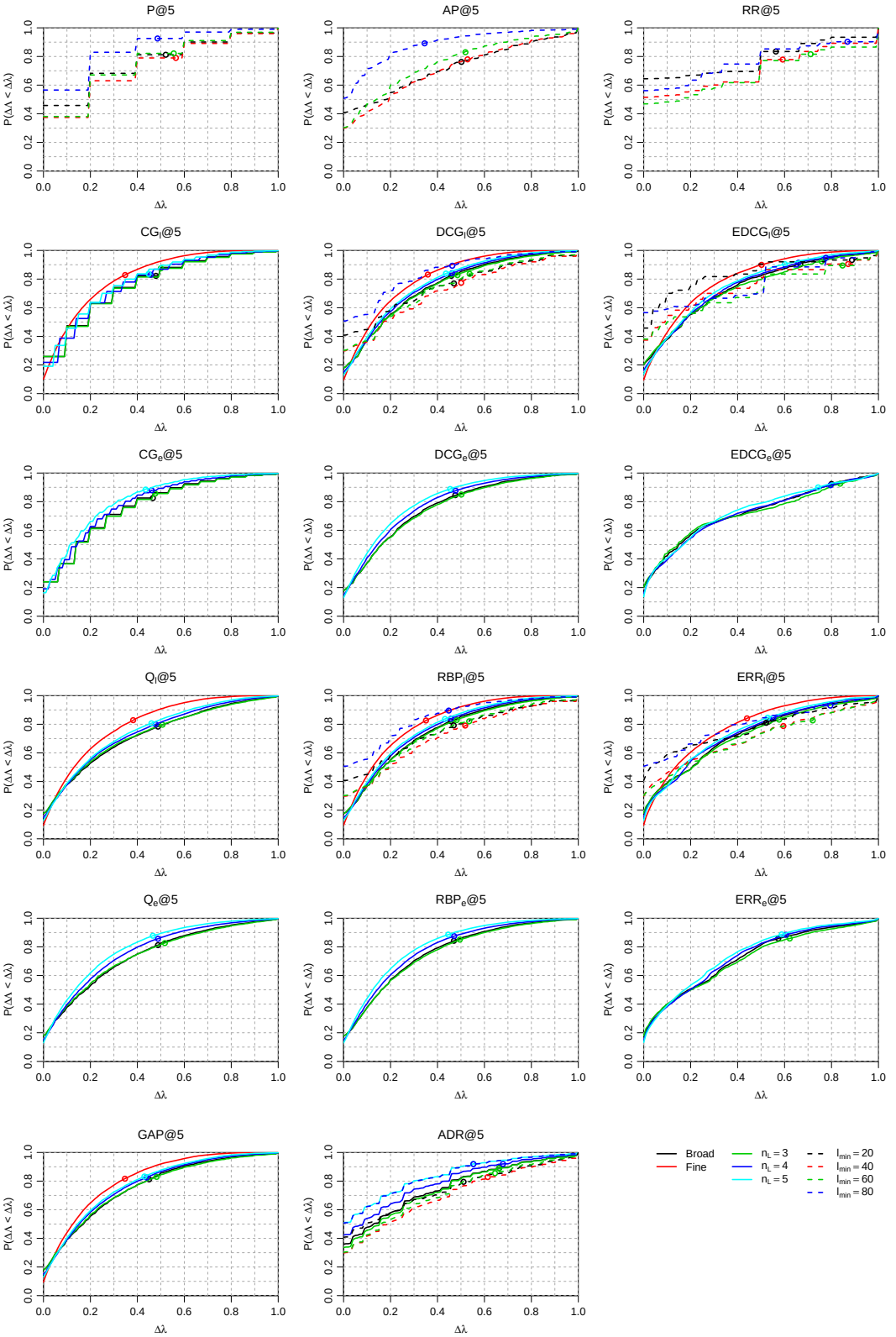

Figure 3.6: Cumulative distribution functions of prior $\Delta\lambda$ scores. Points mark $\Delta\lambda$ such that $P(Agr = 1|\Delta\Lambda \geq \Delta\lambda) \geq 0.5$ (lower is better).

Measure	Original		Artificial Graded			Artificial Binary			
	Broad	Fine	$n_{\mathcal{L}}=3$	$n_{\mathcal{L}}=4$	$n_{\mathcal{L}}=5$	$\ell_{min}=20$	$\ell_{min}=40$	$\ell_{min}=60$	$\ell_{min}=80$
P@5						0.1881	**0.2103**	0.1785	0.0743
AP@5						***0.238***	0.2201	0.1702	*0.1082*
RR@5						0.1654	**0.2216**	*0.1842*	0.0971
CG_l@5	**0.1774**	0.1718	0.1249	0.1698	0.1634				
CG_e@5	**0.1746**		0.1433	0.1205	0.1157				
DCG_l@5	0.1769	0.1685	0.1724	0.1587	0.1615	**0.2303**	*0.2262*	0.1642	0.107
DCG_e@5	**0.1543**		0.1501	0.123	0.1118				
$EDCG_l$@5	0.0972	0.1008	0.0763	0.0861	0.0954	0.0669	0.0951	**0.1054**	0.0518
$EDCG_e$@5	0.0763		0.0743	0.0853	**0.1004**				
Q_l@5	***0.2163***	0.1715	*0.2028*	*0.2003*	*0.1933*				
Q_e@5	**0.1878**		0.1737	0.1434	0.1224				
RBP_l@5	0.1788	0.1735	0.1705	0.1606	0.163	0.2075	**0.2091**	0.1793	0.1044
RBP_e@5	**0.1573**		0.1493	0.1251	0.1142				
ERR_l@5	0.1803	0.1587	0.1676	0.1549	0.1541	0.1895	**0.2122**	0.1733	0.0695
ERR_e@5	**0.1417**		**0.1417**	0.1159	0.1129				
GAP@5	**0.1878**	*0.1832*	0.1706	0.1749	0.1677				
ADR@5	0.1084		0.1112	0.0799	0.0803	**0.2052**	0.1717	0.1448	0.081

Table 3.5: Expected fraction of observations such that $P(Agr = 1) \geq 0.5$ (higher is better). Best per measure in bold, best per scale in italics.

The points in the plots mark the interpolated minimum $\Delta\lambda$ from Figure 3.4 such that $P(Agr = 1|\Delta\Lambda \geq \Delta\lambda) \geq 0.5$, that is, the minimum difference that needs to be observed for the standard goal of having over 50% of users agree as to which system is better. For instance, in Figure 3.4 we saw that the minimum ΔGAP@5 required ranged from 0.35 with the Fine scale to 0.5 with the $n_{\mathcal{L}} = 3$ scale, suggesting the use of the Fine scale. However, Figure 3.6 suggests that if we consider priors all scales are virtually the same: about 15% of the times we will meet our goal. Similarly, with Fine judgments we find that about 83% of the ΔQ_l@5 observations are below the $\Delta\lambda$ threshold, so we are expected to have over 50% of users agreeing only in about 17% of the cases. Surprisingly, the Broad scale, that required a larger $\Delta\lambda$ score, is successful about 22% of the times. In the case of RBP_l@5 we find that the $\ell_{min} = 20$ and $\ell_{min} = 80$ binary scales have almost the same $\Delta\lambda = 0.45$ threshold, but the latter is successful twice as often as the former to meet our goal.

In general, we want the marks to be as close to the bottom $y = 0$ axis as possible. That means that in most of the actual observations we have a $\Delta\lambda$ score sufficiently large to expect over 50% of users preferring the better system. Table 3.5 shows the fraction of observations that are expected to meet the $P(Agr = 1) \geq 0.5$ criterion. Even though differences are generally small, the Broad scale appears to be superior, and the $\ell_{min} = 40$ scale works exceptionally well. In terms of measures, Q_l@5 is clearly the best of all, followed by a mix of combinations in CG_l@5, DCG_l@5, RBP_l@5 and GAP@5.

3.5 Discussion

In terms of measures, the Λ_l versions worked better than the Λ_e versions to predict user satisfaction (good nonpreferences). They also proved to perform better to predict user-

evaluation agreement (positive preferences), although they resulted in slightly more disagreements (negative preferences) too. Both $RR@k$ and $ERR@k$ follow a user model in which the goal is to retrieve one single highly relevant document. These measures showed poor performance in all aspects, evidencing that this user model is not appropriate for the AMS task as presented to users. A music recommendation scenario was suggested, in which users are expected to just consume results pretty much as they would listen to a playlist. If the scenario were that of identifying versions or plagiarized music, then finding that one highly relevant document would definitely be more appropriate. On the other hand, $CG@k$ is a measure assuming that all documents contribute to the user and independently of other documents, and $DCG@k$ further considers a positional user model were the contribution of a document depends on the rank at which it is retrieved; these measures are therefore expected to perform better in the assumed music recommendation scenario. Indeed, they generally obtained the best or next to best results. $RBP@k$ is a variation that accounts differently for the document ranks, and it is generally among the top three measures, especially in its $RBP_l@k$ version. $EDCG@k$ is a mix of $DCG@k$ and $ERR@k$ in which all documents contribute but depending on the other documents. This measure has the worst performance overall, further suggesting that the cascade user model in $ERR@k$ is not appropriate. $ADR@k$ also showed quite poor performance as expected, because it is extremely demanding in the user-oriented formulation employed here. This can be seen in that quite many users considered the results satisfactory even when the $ADR@5$ scores were very low. The other binary measures $P@k$ and $AP@k$, together with $Q@k$, showed average performance. Finally, $GAP@k$ was also among the best measures overall, especially when focusing on extremely low and large scores.

In terms of relevance scales, the original Broad and Fine scales performed best overall. The Broad scale proved to be particularly good to predict user satisfaction, while the Fine scale worked better to differentiate between systems thanks to its greater resolution. The artificial graded scales appear to be next in general, especially $n_{\mathcal{L}} = 4$ and $n_{\mathcal{L}} = 5$. The binary scales were worse overall, although some sporadic combinations were particularly good with $\ell_{min} = 40$. For a task like AMS these results are rather expected. The use of relevance scales with a pre-fixed number of levels is inherited from evaluation in Text IR, where there often are clear guidelines describing the characteristics of documents to assign one or another relevance level (e.g. documents discussing some topic are relevant, and if it is the main topic of the documents they are very relevant). Such guidelines seem unrealistic for music similarity, where relevance is rather continuous: a song can be increasingly modified and still resemble the original song, but similarity is gradually weaker as more and more changes are made. Level-based relevance scales do not seem suitable for similarity tasks because there is no accepted criterion to distinguish between levels. In fact, sometimes assessors go back to a document to change its judgment, after seeing a different song that makes them reconsider the boundaries between relevance levels [Jones et al. 2007].

3.6 Summary

Intuition tells us that if the effectiveness of a system for some query is $\lambda = 1$ any user should be satisfied by the system, and if it is $\lambda = 0$ then no user should. In general, we

expect $100\lambda\%$ of users to like the system. Similarly, if system A obtained a score larger than system B, we expect users to agree and actually prefer A. By choosing one or another effectiveness measure researchers make different assumptions as to the behavior and needs of the final users, and by choosing one or another relevance scale they follow different criteria to differentiate satisfactory from unsatisfactory results. Section 3.3.1 empirically provides the mapping from system effectiveness onto user satisfaction for a wide range of effectiveness measures and relevance scales, and Section 3.3.2 provides the mapping from differences in effectiveness onto user-evaluation agreement, allowing us to validate or not these assumptions.

Figure 3.3 allows researchers to interpret effectiveness scores in practical terms and answer the larger question of whether a system satisfies users or not. These results also allow us to quantify user disagreement and how much room for improvement there is if we implemented personalization of results [Järvelin 2011]. The figure shows that about 15%-20% of users contradict the effectiveness measures, which is consistent with disagreements found between AMS assessors [Jones et al. 2007, Schedl et al. 2013a] and in related tasks such as Genre Classification [Lippens et al. 2004, Seyerlehner et al. 2010a]. This implies a practical lower and upper bound on both user satisfaction and system effectiveness, and also suggests the need for a larger user component beyond mere relevance judgments to consider user properties and user context in evaluation experiments [Schedl et al. 2013a]. The incorporation of user context features such as mood or location is problematic because they are not static. But user properties such as musical background or taste should be fairly easy to incorporate in the Cranfield framework by making them a static part of the query itself. That is, the problem would go from retrieving similar music to a given song to retrieving songs similar to a query and targeting a user with certain characteristics. This is the scenario of the recent Million Song Dataset Challenge [McFee et al. 2012], that incorporates a user-specific listening history as user property to predict songs that the user should also like.

Figure 3.4 and Figure 3.5 allow researchers to interpret differences in effectiveness in practical terms, answering the question of whether users will actually prefer one system or another. These results show that effectiveness differences need to be quite large for the majority of users to actually prefer one system. Although there are variations across measures and scales, it is generally required to observe $\Delta\lambda > 0.4$. Historically though, only about 20% of system comparisons in MIREX have resulted in such large differences. For smaller differences users generally can not decide between one system or another; they seem equally good or bad. This does not mean that incrementally implementing slight improvements in a system will not have practical implications for users; at some point they may all add up to a difference sufficiently large for users to note it.

Chapter 4
Modeling Distributions

Previous chapter provided the tools to map the effectiveness score for one query onto the arguably more meaningful probability of user satisfaction for that query. But the description of evaluation results usually focuses just on the average effectiveness of systems over some sample of queries. In this chapter effectiveness-satisfaction relation is modeled to provide good estimates of average user satisfaction as well.

In addition, I discuss how to move beyond simple means and consider the full score distributions. Doing so we are able to describe the performance of systems from new perspectives. First, it allows us to analyze the variability of scores so that we can better study the expected user satisfaction and predict how extreme systems can be. Second, it allows us to easily categorize system results as successful or unsuccessful, so researchers can focus on problematic types of queries. It is shown that, considering the full distribution of scores, apparently straightforward comparisons between systems can be more complex than it may seem. In fact, conclusions based on simple averages of effectiveness can easily be contradicted by conclusions based on full distributions of user satisfaction.

4.1 Mean Probability of User Satisfaction

Section 3.3.1 provided an empirical mapping between system effectiveness and user satisfaction. In particular, we were able to map an effectiveness score λ onto a probability of user satisfaction $P(Sat|\lambda)$. For simplicity, let us refer to this mapping as a function:

$$sat(\lambda) := P(Sat = 1|\Lambda = \lambda) \tag{4.1}$$

This mapping allows us to interpret the results of an evaluation experiment in terms of user satisfaction. If for an arbitrary query q a system obtains λ_q, the probability that an arbitrary user finds the system satisfactory for that query is $sat(\lambda_q)$. We can further consider user satisfaction as a random variable following a Bernoulli distribution with probability of success $sat(\lambda_q)$; doing so we can define the random variable Sat_n that equals the number of satisfied users from a total of n users. This variable follows a Binomial distribution:

Measure	Original		Artificial Graded		Artificial Binary	
	Broad	Fine	$n_{\mathcal{L}}=4$	$n_{\mathcal{L}}=5$	$\ell_{min}=20$	$\ell_{min}=40$
P@5					x	x
AP@5					x	x
CG_l@5	x	x	x	x	P@5	P@5
CG_e@5	x		x	x	P@5	P@5
DCG_l@5	x	x	x	x	x	x
DCG_e@5	x		x	x	DCG_l@5	DCG_l@5
Q_l@5	x	x	x	x	AP@5	AP@5
Q_e@5	x		x	x	AP@5	AP@5
RBP_l@5	x	x	x	x	x	x
RBP_e@5	x		x	x	RBP_l@5	RBP_l@5
GAP@5	x	x	x	x	AP@5	AP@5

Table 4.1: All 40 combinations of effectiveness measures and relevance scales studied (marked with x), and equivalent combinations (e.g. GAP@5 is the same as AP@5 with a binary scale).

$$\begin{aligned} P(Sat_n = k|\Lambda = \lambda) &= \binom{n}{k} sat(\lambda)^k \left(1 - sat(\lambda)\right)^{n-k} \\ &= \frac{n!}{k!(n-k)!} sat(\lambda)^k \left(1 - sat(\lambda)\right)^{n-k} \end{aligned} \tag{4.2}$$

As an example, let us consider system STBD1 from MIREX 2011 and query d007449. Using the Fine judgments, the effectiveness obtained was Q_l@5 $= 0.6095$, which according to Figure 3.3 corresponds to $P(Sat) = sat(0.6095) \approx 0.7$. It is therefore expected that about 70% of users will find the results for that query satisfactory; if 15 different users were asked, the probability that 10 will be satisfied is $P(Sat_{15} = 10) \approx 0.2061$.

4.1.1 User Satisfaction over a Sample of Queries

Equations (4.1) and (4.2) can be used to compute the expected user satisfaction for a single query, but the larger question relates to the expected user satisfaction on the universe of all queries, that is, the mean probability of satisfaction $\mu_{P(Sat)}$. If the sat mapping functions were linear we could just compute the mapping of the mean effectiveness μ_λ, but judging by Figure 3.3 they are not. We therefore have to integrate the sample space to get the expected probability of user satisfaction as:

$$\mu_{P(Sat)} = \int sat(\lambda) P(\Lambda = \lambda)\, \mathrm{d}\lambda$$

However, because all effectiveness scores are computed for a cutoff k and the set of possible relevance judgments is $\mathcal{L}$, the sample space for effectiveness scores is a finite and countable set. That is, the distribution of effectiveness is discrete, and the expected probability of user satisfaction is therefore calculated as:

$$\mu_{P(Sat)} = \sum sat(\lambda) P(\Lambda = \lambda)$$

This would be the expected probability of satisfaction for an arbitrary user and query. The problem at this point is that the actual distribution $P(\Lambda = \lambda)$ is unknown. We would need

Measure	Original		Artificial Graded		Artificial Binary	
	Broad	Fine	$n_{\mathcal{L}}=4$	$n_{\mathcal{L}}=5$	$\ell_{min}=20$	$\ell_{min}=40$
P@5					***0.0034***	0.0141
AP@5					0.0402	***0.0267***
CG_l@5	0.0254	0.0206	0.0254	**0.0183**		
CG_e@5	0.0334		0.0339	**0.0315**		
DCG_l@5	0.0265	**0.0154**	*0.0176*	*0.0155*	0.0325	0.0346
DCG_e@5	0.0265		**0.02**	0.0281		
Q_l@5	***0.0155***	0.0217	0.0229	0.0205		
Q_e@5	0.0302		**0.0206**	0.0233		
RBP_l@5	0.0272	***0.0141***	0.0229	0.0188	0.0275	0.038
RBP_e@5	**0.0229**		0.0242	0.0252		
GAP@5	0.0302	**0.0205**	0.0208	0.0262		

Table 4.2: RMS residuals of $\widehat{sat}$ predictions (lower is better). Best per measure in bold, best per scale in italics.

to evaluate the system for the universe of all queries and all users to know this distribution, but in reality we only use a sample of queries $\mathcal{Q}$ and a sample of assessors $\mathcal{H}$. By the Law of Large Numbers, the sample mean $\overline{\mu}_{P(Sat)}$ converges to the true mean almost surely as $n_{\mathcal{Q}} \to \infty$, so we use it as an estimator of the true population mean:

$$\hat{\mu}_{P(Sat)} = \frac{1}{n_{\mathcal{Q}}} \sum_{q \in \mathcal{Q}} sat(\lambda_q)$$

4.1.2 Interpolated Probability of User Satisfaction

Note that the true *sat* mapping function is also unknown. The mapping from Figure 3.3 was also obtained empirically with a sample of users, and not for the actual λ scores but for intervals $\{[0, 0.1), [0.1, 0.2), \ldots, [0.9, 1]\}$. This means that we assigned the same probability of satisfaction to any two effectiveness scores in the same interval, which most certainly is not true. To better estimate the *sat* mapping function I proceeded to interpolate the known estimates in Figure 3.3. Judging by the plots, a cubic polynomial fit should be sufficiently powerful to describe the data, so the following model was fitted using least squares:

$$\widehat{sat}(\lambda) = a_0 + a_1\lambda + a_2\lambda^2 + a_3\lambda^3 \tag{4.3}$$

Based on the results from Chapter 3 some $(\Lambda, \mathcal{L})$ combinations are discarded. In particular, I discard the RR@5, $EDCG$@5, ERR@5 and ADR@5 measures, as well as the $n_{\mathcal{L}} = 3$, $\ell_{min} = 60$ and $\ell_{min} = 80$ relevance scales. This leaves us with 40 combinations from this point on (see Table 4.1). Figure 4.1 plots the fits for the $(\Lambda, \mathcal{L})$ combinations of interest. The explained variance ranged from $R^2 = 0.9561$ to $R^2 = 0.9998$, with an average of $R^2 = 0.9858$. Table 4.2 lists the rooted mean squared residuals, ranging from 0.0034 (P@5 with $\ell_{min} = 20$) to 0.0402 (AP@5 with $\ell_{min} = 20$), with an average of 0.0241. The cubic model in (4.3) thus resulted in a quite good fit on the actual data; the average error of $\widehat{sat}$ predictions is about 2%. In terms of measures, P@5 was fitted particularly well, followed by the Λ_l graded measures and GAP@5. In terms of relevance scales, fits were similarly good, but the Fine and $n_{\mathcal{L}} = 4$ scales were fitted slightly

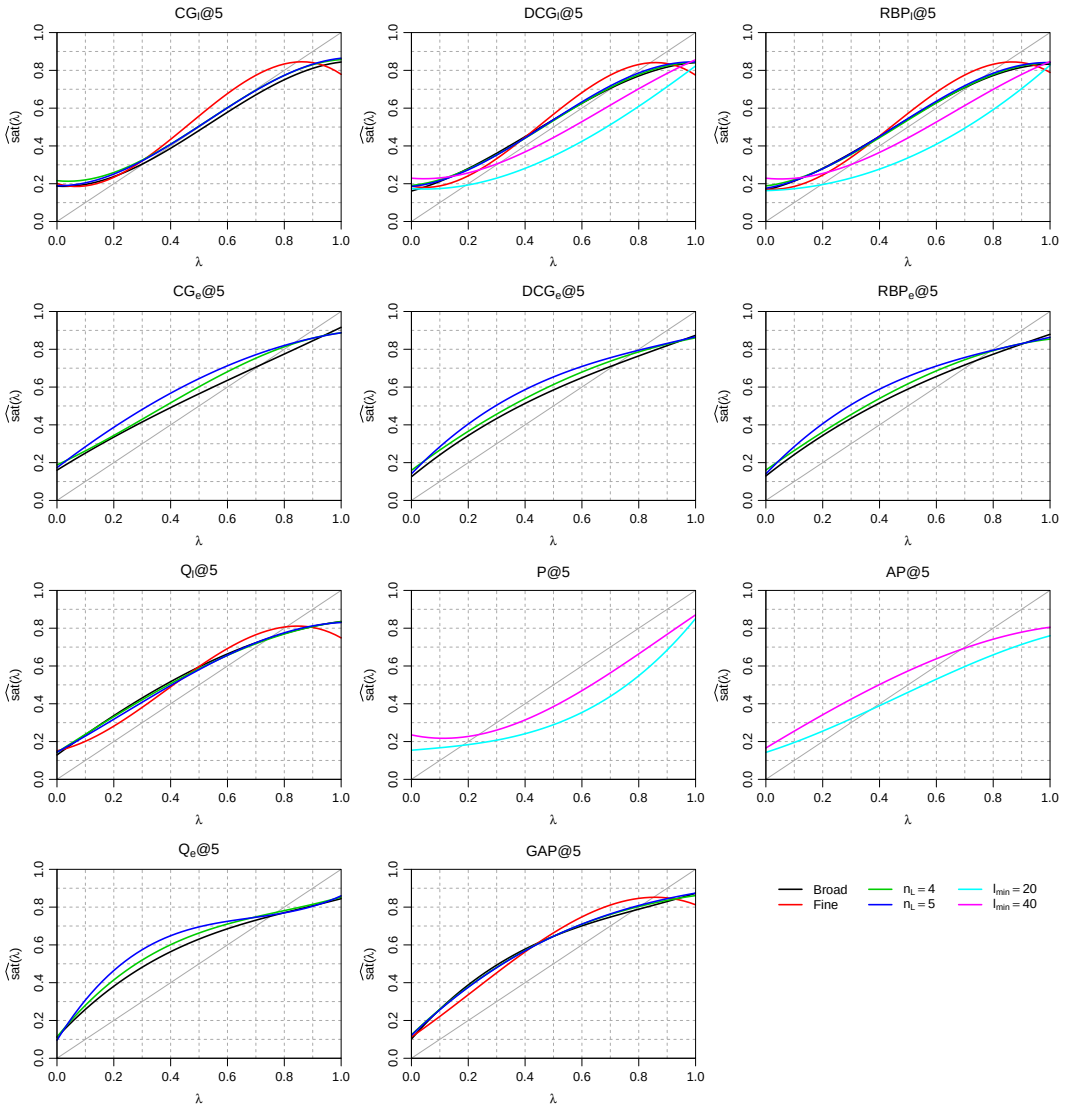

Figure 4.1: Estimated $\widehat{sat}(\lambda)$ mappings fitted from Figure 3.3.

better. Table 4.3 lists the fitted parameters for all $(\Lambda, \mathcal{L})$ combinations of interest. As an example, an effectiveness score $GAP@5 = 0.32$ with Broad judgments corresponds to $\hat{P}(Sat) = 0.1018 + 1.7272 \cdot 0.32 - 1.6028 \cdot 0.32^2 + 0.6471 \cdot 0.32^3 = 0.5116$.

Finally, the expected probability of user satisfaction for an arbitrary query and user can be estimated with a test collection as:

$$\mathrm{E}\left[\hat{\mu}_{\hat{P}(Sat)}\right] = \frac{1}{n_{\mathcal{Q}}} \sum_{q \in \mathcal{Q}} \widehat{sat}(\lambda_q) \tag{4.4}$$

As an example, let us consider the CL1 system from MIREX 2009; Figure 4.2 shows the histograms of effectiveness ($CG_l@5$ with Fine judgments) and estimated probability of user satisfaction. The sample mean effectiveness is $\overline{CG_l@5} = 0.2525$ which, intuitively, would be

Measure	Broad				Fine			
	a_0	a_1	a_2	a_3	a_0	a_1	a_2	a_3
CG_l@5	0.1872	-0.0969	1.9908	-1.237	0.2007	-0.4632	3.6754	-2.6338
CG_e@5	0.1601	0.9345	-0.3245	0.1463				
DCG_l@5	0.1614	0.4043	1.1288	-0.8535	0.1873	-0.3	3.3552	-2.4675
DCG_e@5	0.1253	1.2334	-0.7733	0.2884				
Q_l@5	0.1291	1.0993	-0.2774	-0.1157	0.1509	0.3214	2.0057	-1.7292
Q_e@5	0.1117	1.6064	-1.4001	0.5267				
RBP_l@5	0.1666	0.3591	1.2609	-0.9536	0.1722	-0.1443	3.0142	-2.253
RBP_e@5	0.1297	1.1906	-0.6452	0.204				
GAP@5	0.1018	1.7272	-1.6028	0.6471	0.1131	1.0327	0.6077	-0.9409

Measure	$n_{\mathcal{L}} = 4$				$n_{\mathcal{L}} = 5$			
	a_0	a_1	a_2	a_3	a_0	a_1	a_2	a_3
CG_l@5	0.2162	-0.1609	2.151	-1.3492	0.1895	-0.0282	1.9266	-1.2236
CG_e@5	0.1879	0.6952	0.5243	-0.5204	0.1734	1.1467	-0.3836	-0.0491
DCG_l@5	0.1908	0.1522	1.637	-1.1374	0.1853	0.1434	1.737	-1.2215
DCG_e@5	0.1592	1.1177	-0.4211	0.0049	0.141	1.5581	-1.2976	0.465
Q_l@5	0.143	0.9574	-0.0258	-0.2404	0.1438	0.8224	0.3413	-0.4773
Q_e@5	0.1125	1.8645	-1.93	0.8061	0.0956	2.4332	-3.2653	1.5964
RBP_l@5	0.1884	0.1822	1.5873	-1.1176	0.1761	0.2562	1.5186	-1.1098
RBP_e@5	0.1605	1.0565	-0.2106	-0.1524	0.1406	1.5719	-1.3141	0.4653
GAP@5	0.124	1.4399	-0.883	0.1802	0.1209	1.4456	-0.8938	0.2014

Measure	$\ell_{min} = 20$				$\ell_{min} = 40$			
	a_0	a_1	a_2	a_3	a_0	a_1	a_2	a_3
P@5	0.1541	0.1227	0.0152	0.5589	0.2352	-0.3261	1.5421	-0.5807
AP@5	0.1428	0.4791	0.4859	-0.3479	0.1659	0.9044	-0.0926	-0.1725
DCG_l@5	0.1742	-0.0972	1.0231	-0.2768	0.2292	-0.1256	1.4722	-0.7195
RBP_l@5	0.1635	0.0401	0.6029	0.0208	0.2291	-0.1532	1.5484	-0.7779

Table 4.3: Fitted parameters of the $\widehat{sat}(\lambda) = a_0 + a_1\lambda + a_2\lambda^2 + a_3\lambda^3$ models.

interpreted as roughly 25% of user satisfaction. However, the sample mean probability of satisfaction is $\hat{P}(Sat) = 0.3526$, indicating that on average about 35% of users are expected to find the system satisfactory. This is a clear example that system effectiveness and user satisfaction do not have an equality relation as intuition dictates. In fact, in this case we underestimated user satisfaction by about 10%.

4.1.3 Sampling Distribution of the Mean Probability of User Satisfaction

Equation (4.4) estimates the mean probability of user satisfaction based on a sample of queries. As there is random error due to the sampling process, it is customary to provide a measure of confidence on the estimates. The variance of the estimate is:

$$\mathrm{Var}\left[\hat{\mu}_{\hat{P}(Sat)}\right] = \frac{sd^2_{\hat{P}(Sat)}}{n_{\mathcal{Q}}} \tag{4.5}$$

where $sd_{\hat{P}(Sat)}$ is the sample standard deviation. By the Central Limit Theorem, we know that the sampling distribution of $\hat{\mu}_{\hat{P}(Sat)}$ is approximately normal as $n_{\mathcal{Q}} \to \infty$, so we can

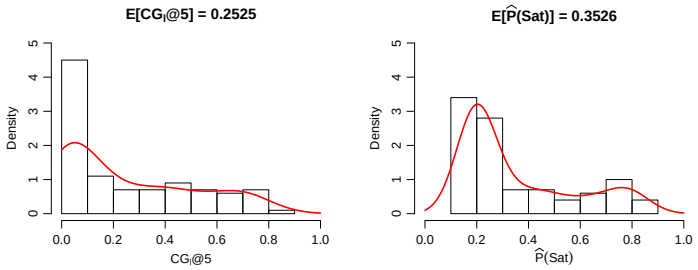

Figure 4.2: Distributions of $CG_l@5$ and corresponding $\hat{P}(Sat)$ for system CL1 in MIREX 2009.

compute a $100(1-2\alpha)\%$ interval estimate for the mean as:

$$\mathrm{E}\left[\hat{\mu}_{\hat{P}(Sat)}\right] \pm t_\alpha \sqrt{\mathrm{Var}\left[\hat{\mu}_{\hat{P}(Sat)}\right]} = \overline{\hat{P}(Sat)} \pm t_\alpha \frac{sd_{\hat{P}(Sat)}}{\sqrt{n_{\mathcal{Q}}}} \tag{4.6}$$

where t_α is the quantile function of the t-distribution with $n_{\mathcal{Q}} - 1$ degrees of freedom. In the previous example with system CL1, the sample standard deviation is $sd_{\hat{P}(Sat)} = 0.2187$, so a 95% confidence interval would be $\hat{\mu}_{\hat{P}(Sat)} = 0.3526 \pm 1.9842 \cdot 0.2187/10 = [0.3092, 0.3960]$. The same procedure can of course be followed to better describe effectiveness. In this case $sd_{CG_l@5} = 0.2557$, so a 95% confidence interval would be $\hat{\mu}_{CG_l@5} = [0.2018, 0.3033]$.

4.2 Distribution of the Probability of User Satisfaction

Equations (4.4) and (4.5) allow researchers to calculate the point and interval estimates for the average probability of user satisfaction. However, this only tells us about average behavior in the long run, not about what to actually expect for an arbitrary new query and user. That is, in the example above it is expected that if we evaluate different samples of queries $\mathcal{Q}_1, \mathcal{Q}_2, \ldots, \mathcal{Q}_m$ and compute the sample average for each of them, 95% of those sample averages will be in the estimated $[0.3092, 0.3960]$ interval. However, given a new and arbitrary query q, what is the range in which we can expect the probability of satisfaction to be? It will of course be between 0 and 1[1], but we are similarly interested in computing intervals up to some confidence level. These are prediction intervals for *new* observations, as opposed to confidence intervals for the *average* over these observations.

As an example, let us consider the distribution in Figure 4.3. On average, 95% of all new observations lie in the red interval (prediction interval), and the sample mean of 95% of independent samples lie in the blue interval (confidence interval on the mean). With larger samples the empirical distribution (histogram) converges to the true distribution (red), and the sampling distribution (blue) gets narrower (more precision) around the true mean μ (more accuracy).

In the ideal case of knowing the distribution of $\hat{P}(Sat)$ we could compute a $100(1-2\alpha)\%$ prediction interval as:

$$\left[Q_{\hat{P}(Sat)}(\alpha), Q_{\hat{P}(Sat)}(1-\alpha)\right]$$

[1] Actually, Section 4.1 showed that 0 and 1 are never expected because of the inherent user disagreement.

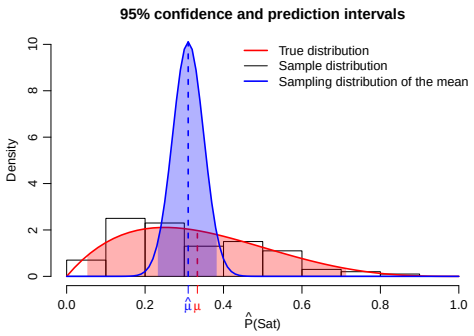

Figure 4.3: True distribution, histogram of a sample, and sampling distribution of the mean.

where Q is the quantile function or inverse cumulative distribution function. That is, we compute the interval covering the central $100(1-2\alpha)\%$ of the probability mass function. We can also compute 1-tailed intervals to describe the distribution. For example, the bottom $100\alpha\%$ of the observations have $\hat{P}(Sat) \leq Q_{\hat{P}(Sat)}(1-\alpha)$.

4.2.1 Probability Distributions

But then again, we do not know the true distribution because we use a sample of queries, so we have to estimate it. Four probability distributions are considered for this purpose.

Empirical Distribution

The easiest way to do so is to use the Empirical distribution. The empirical cumulative distribution function is defined as:

$$ecdf(s) = \frac{1}{n_{\mathcal{Q}}} \sum_{q \in \mathcal{Q}} \mathbb{1}\Big(\widehat{sat}(\lambda_q) \leq s\Big)$$

that is, the fraction of queries for which the predicted probability of satisfaction is less than or equal to s. The *ecdf* is a step function with increments of $1/n_{\mathcal{Q}}$ at each of the observations in the sample. This means that the resolution of the empirical quantile function is $1/n_{\mathcal{Q}}$, and consequently it needs to compute estimates at discontinuities [Hyndman and Fan 1996]. For example, let $\langle \widehat{sat}(\lambda_1), \ldots, \widehat{sat}(\lambda_{n_{\mathcal{Q}}}) \rangle$ be the sequence of observations sorted in ascending order. Each of these observations $\widehat{sat}(\lambda_i)$ corresponds to the $100(i-1)/(n_{\mathcal{Q}}-1)$-quantile, but the quantile between two consecutive observations needs to be estimated because there is no data to calculate it.

By the Strong Law of Large Numbers, the quantile function converges to the true F function as $n_{\mathcal{Q}} \to \infty$ almost surely, and therefore *eqf* converges to Q too. But we do not know how many queries are sufficient, or how sufficient is sufficient enough for that matter. In addition, our objective in Chapter 6 is actually to reduce $n_{\mathcal{Q}}$ as a means to reduce general evaluation cost, so we need to explore other ways to estimate F that work reasonably well for smaller test collections. At the very least, we need to get an idea of how good our estimates are.

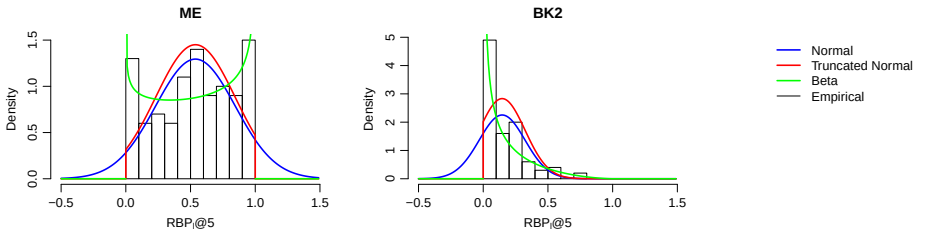

Figure 4.4: Examples of distribution fits for RBP_l@5 with Broad judgments for systems ME and BK2 from MIREX 2007.

Normal Distribution

The second alternative considered is the familiar Normal distribution with parameters mean and standard deviation: $\mathcal{N}(\mu, \sigma)$. The cumulative distribution function is:

$$\Phi(s; \mu, \sigma) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{(s-\mu)/\sigma} e^{\frac{-t^2}{2}} \, \mathrm{d}t$$

The benefit of using the Normal distribution is that it can be easily estimated with mean $\mu = \hat{\mu}_{\hat{P}(Sat)}$ and standard deviation $\sigma = sd_{\hat{P}(Sat)}$ from equations (4.4) and (4.5). The downside is that it is supported on the interval $[-\infty, +\infty]$, while both Λ and $\hat{P}(Sat)$ are supported on the $[0, 1]$ interval. This means that according to this distribution there is some probability, however small, that a new observation falls outside the $[0, 1]$ interval. Therefore, it is expected that Φ overestimates close to 0 and underestimates close to 1.

Truncated Normal Distribution

One way to solve this problem is to use the Truncated Normal distribution $\mathcal{N}'(\mu, \sigma, a, b)$ in the interval $[a, b]$. The cumulative distribution function is:

$$\Phi'(s; \mu, \sigma, a, b) = \begin{cases} 0 & \text{if } s < a \\ 1 & \text{if } s > b \\ \dfrac{\Phi\left(\frac{s-\mu}{\sigma}\right) - \Phi\left(\frac{a-\mu}{\sigma}\right)}{\Phi\left(\frac{b-\mu}{\sigma}\right) - \Phi\left(\frac{a-\mu}{\sigma}\right)} & \text{otherwise} \end{cases}$$

which effectively removes all density to the left of a and to the right of b, and then normalizes by the density left between a and b; in our case $a = 0$ and $b = 1$. This distribution can be easily estimated again with $\mu = \hat{\mu}_{\hat{P}(Sat)}$ and standard deviation $\sigma = sd_{\hat{P}(Sat)}$ from equations (4.4) and (4.5).

Beta Distribution

Finally, the Beta distribution $Beta(\alpha, \beta)$ with shape parameters α and β is also considered. It is supported on the $[0, 1]$ interval by definition. The cumulative distribution function is:

$$B(s;\alpha,\beta) = \frac{\int_0^s t^{\alpha-1}(1-t)^{\beta-1}\,\mathrm{d}t}{\int_0^1 t^{\alpha-1}(1-t)^{\beta-1}\,\mathrm{d}t}$$

When $sd^2 < \overline{\mu}(1-\overline{\mu})$, the shape parameters can be estimated by the method of moments:

$$\hat{\alpha} = \overline{\mu}\left(\frac{\overline{\mu}(1-\overline{\mu})}{sd^2} - 1\right)$$
$$\hat{\beta} = (1-\overline{\mu})\left(\frac{\overline{\mu}(1-\overline{\mu})}{sd^2} - 1\right)$$

If not, they can be estimated by Maximum Likelihood. Unfortunately, the maximum likelihood estimates for α and β do not have a closed form, though they can be solved by numerical methods [Beckman and Tietjen 1978]. The drawback of using the Beta distribution is that in some cases it requires numerical methods to be estimated; on the other hand, they are readily available in virtually any statistical software. Also, under certain combinations of the shape parameters $B(s;\alpha,\beta)$ can be extremely large for $s \to 0$ and $s \to 1$. The upside is that it is much more versatile than the Normal distribution: while the latter is restricted to its well-known bell shape, the Beta distribution can take several other shapes.

In summary, we compare the Normal, Truncated Normal, Beta and Empirical distributions, defined by their cumulative distribution functions Φ, Φ', B and *ecdf* (see Figure 4.4 for a graphical comparison). Instead of fitting the $\hat{P}(Sat)$ distributions, I fitted the Λ distributions. This is because the former is expected to be very similar across effectiveness measures and relevance scales, and it is actually estimated from the latter distribution.

4.2.2 Goodness of Fit

Given a full query set $\mathcal{Q}$, a sample $\mathcal{Q}_1$ of size $n_{\mathcal{Q}_1}$ queries is randomly selected. The sample of $\lambda_{\mathcal{Q}_1}$ scores is then computed, and all four cumulative distribution functions are fitted to it. Ideally, we would measure the goodness of fit of each distribution by comparing their cumulative distribution functions to the true F_Λ function, but this is unfortunately unknown. Instead, they are compared with the *ecdf* of the distribution of scores from the disjoint subset of leftover queries; let these be $\mathcal{Q}_2 = \mathcal{Q} - \mathcal{Q}_1$ with size $n_{\mathcal{Q}_2} = n_{\mathcal{Q}} - n_{\mathcal{Q}_1}$ queries. That is, we are measuring the predictive power of the four distribution fits.

The question now is how to measure the goodness of fit between each of the estimated $\hat{F}$ functions and the true F (estimated themselves by $ecdf_{\mathcal{Q}_2}$). The often used Kolmogorov-Smirnov D statistic measures the maximum absolute difference between the two functions [Kolmogorov 1933]:

$$D = \sup_\lambda \left|F(\lambda) - \hat{F}(\lambda)\right|$$

By the Glivenko-Cantelli theorem, when $\hat{F} = ecdf$ then D converges to 0 almost surely as $n_{\mathcal{Q}_1} \to \infty$, suggesting that for large collections the Empirical distribution will work better. But this is not necessarily true for small samples, nor for the Normal, Truncated Normal and Beta approximations. The D statistic is very simple in that it does not provide any

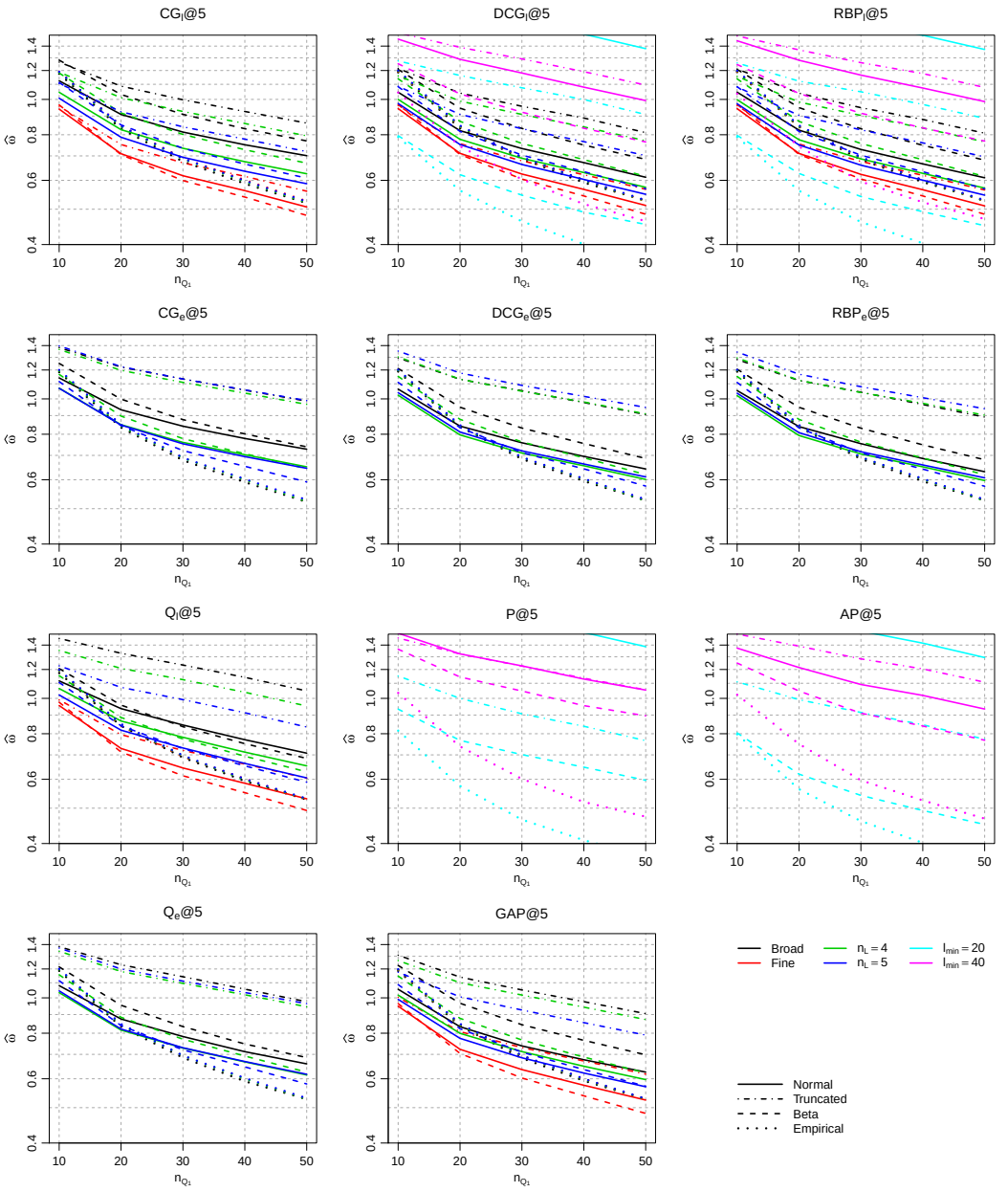

Figure 4.5: Average $\hat{\omega}$ statistics of the fits provided by the Normal, Truncated Normal, Beta and Empirical distributions for different query set sizes.

information about where this maximum distance occurs or about total distance between the functions for that matter. A more informative measure is the Cramér-von Mises ω^2 statistic [Cramér 1928, von Mises 1931]:

$$\omega^2 = \int_{-\infty}^{+\infty} \left(F(\lambda) - \hat{F}(\lambda) \right)^2 \, \mathrm{d}F(\lambda)$$

which measures the total squared distance between F and $\hat{F}$. Because our distributions are discrete, the following definition is used instead[2]:

$$\hat{\omega}^2 = \sum_{q \in \mathcal{Q}_2} \left(F(\lambda_q) - \hat{F}(\lambda_q) \right)^2 \tag{4.7}$$

4.2.3 Results

For all 53 systems in the MIREX 2007, 2009, 2010 and 2011 AMS editions, 200 random trials of the experiment were run. This was repeated for each of the forty $(\Lambda, \mathcal{L})$ combinations of interest, and repeated for different query subset sizes $n_{\mathcal{Q}_1} \in \{10, 20, 30, 40, 50\}$. Therefore we have a grand total of 2,212,000 trials, each providing the $\hat{\omega}^2$ statistic for the four $\hat{F}$ estimates.

Figure 4.5 plots the (log-scaled) $\hat{\omega}$ statistics for all combinations[3]. In general, it can be seen that the empirical distribution converges more rapidly towards $\hat{\omega} = 0$, and it provides the best approximation across combinations. The Normal and Beta distributions perform similarly, and the Truncated Normal is clearly outperformed in all cases. Fits are very similar across effectiveness measures, but clear differences can be observed across relevance scales. The Fine scale is better modeled, followed by the $n_{\mathcal{L}} = 5$, $n_{\mathcal{L}} = 4$ and Broad scales. That is, the more fine-grained the relevance scale the better its distribution is modeled. For the binary scales, the best approximation is clearly provided by the Empirical distribution.

For $n_{\mathcal{Q}_1} = 50$, Table 4.4 shows that the Empirical distribution is clearly the best approximation except for the Fine relevance scale, where the Beta distribution takes over and even the Normal distribution performs generally better. However, for small query sets such as $n_{\mathcal{Q}_1} = 20$ things change considerably. Table 4.5 shows that the Empirical distribution still provides the best approximation when there are few relevance levels, such as in the $\ell_{min} = 20$, $\ell_{min} = 40$ and Broad scales, but with more levels the continuous distributions perform better: the Normal distribution provides the best approximation with $n_{\mathcal{L}} = 4$ and $n_{\mathcal{L}} = 5$, while the Beta distribution again shows the best results with the Fine scale.

4.3 Probability of Success

Being able to accurately describe the distribution of effectiveness and map it onto a distribution of probability of satisfaction, allows us to go one step further in the analysis of evaluation results. So far we can estimate the probability $\hat{P}(Sat)$ that a user will find a system satisfactory for some query. But researchers may make the decision of considering the system successful if this probability is larger than some threshold. For instance, one may consider that if the majority of users is satisfied by the system, then we can consider it successful. In this case, we may set the threshold $\hat{P}(Sat) > 0.5$. The probability of success can be estimated as:

$$\hat{P}(Succ) = 1 - \hat{F}_{\hat{P}(Sat)}(0.5) \tag{4.8}$$

that is, the fraction of queries for which the estimated satisfaction is larger than 0.5.

[2] We iterate through query set $\mathcal{Q}_2$ because it is always larger than or equal to the reduced sample $\mathcal{Q}_1$ used to compute the $\hat{F}$ approximations.

[3] I show $\hat{\omega}$ rather than $\hat{\omega}^2$ to maintain units. Relative comparisons are the same.

Measure	Broad				Fine			
	Normal	Trunc.	Beta	Empir.	Normal	Trunc.	Beta	Empir.
CG_l@5	0.701	0.8611	0.7694	**0.5194**	0.5071	0.5602	**0.4808**	0.526
CG_e@5	0.7279	0.986	0.7388	**0.5215**				
DCG_l@5	0.6118	0.8108	0.685	**0.5278**	0.512	0.5671	**0.4844**	0.5262
DCG_e@5	0.6425	0.9061	0.6866	**0.5262**				
Q_l@5	0.7076	1.0482	0.6847	**0.5279**	0.5304	0.6044	**0.4928**	0.5277
Q_e@5	0.6592	0.9757	0.686	**0.5273**				
RBP_l@5	0.6097	0.8078	0.6814	**0.5283**	0.5106	0.5646	**0.4843**	0.5264
RBP_e@5	0.6315	0.8916	0.6811	**0.5266**				
GAP@5	0.6259	0.9052	0.6982	**0.5269**	0.5246	0.6182	**0.4822**	0.5273

Measure	$n_{\mathcal{L}} = 4$				$n_{\mathcal{L}} = 5$			
	Normal	Trunc.	Beta	Empir.	Normal	Trunc.	Beta	Empir.
CG_l@5	0.6254	0.7945	0.6686	**0.5213**	0.5873	0.7195	0.6083	**0.5246**
CG_e@5	0.6512	0.9658	0.6435	**0.5234**	0.6453	0.9896	0.5921	**0.5286**
DCG_l@5	0.5743	0.7703	0.6153	**0.5267**	0.5494	0.7014	0.569	**0.5291**
DCG_e@5	0.6015	0.9098	0.62	**0.5259**	0.6111	0.9476	0.5769	**0.53**
Q_l@5	0.653	0.953	0.629	**0.5295**	0.6052	0.8346	0.5891	**0.5311**
Q_e@5	0.6146	0.9455	0.6255	**0.5279**	0.6173	0.9641	0.5808	**0.5307**
RBP_l@5	0.5712	0.7648	0.6147	**0.5267**	0.5464	0.6957	0.5685	**0.5291**
RBP_e@5	0.5973	0.9021	0.6192	**0.5259**	0.6075	0.9396	0.5762	**0.5297**
GAP@5	0.5969	0.8719	0.6225	**0.5276**	0.5706	0.7905	0.5724	**0.5291**

Measure	$\ell_{min} = 20$				$\ell_{min} = 40$			
	Normal	Trunc.	Beta	Empir.	Normal	Trunc.	Beta	Empir.
P@5	1.386	0.7657	0.5964	**0.3683**	1.0539	1.0551	0.8957	**0.4739**
AP@5	1.2939	0.7727	0.4512	**0.3596**	0.9349	1.1075	0.7677	**0.4681**
DCG_l@5	1.3777	0.9091	0.4543	**0.3605**	0.9926	1.0948	0.7644	**0.4638**
RBP_l@5	1.3704	0.8873	0.4507	**0.3607**	0.9863	1.0784	0.7686	**0.4701**

Table 4.4: Average $\hat{\omega}$ statistics of the fits provided by the Normal, Truncated Normal, Beta and Empirical distributions for $n_{\mathcal{Q}_1} = 50$. Best per measure in bold.

Let us consider an example with systems ANO and GT from MIREX 2009. Figure 4.6 (top) shows their distributions of DCG_e@5 scores with $n_{\mathcal{L}} = 5$. The average effectiveness is nearly the same in both systems: $\mathrm{E}[\lambda_{\mathsf{ANO}}] = 0.3854$ and $\mathrm{E}[\lambda_{\mathsf{GT}}] = 0.3875$. According to the mapping in Table 4.3, the expected probabilities of user satisfaction as per (4.4) are also very similar: $\mathrm{E}\left[\hat{P}(Sat)_{\mathsf{ANO}}\right] = 0.5343$ and $\mathrm{E}\left[\hat{P}(Sat)_{\mathsf{GT}}\right] = 0.5332$. However, the bottom histograms show that the distributions of $\hat{P}(Sat)$ are quite different, and the GT system does indeed have a fatter left tail. As per (4.8), the probabilities of success are $\hat{P}(Succ)_{\mathsf{ANO}} = 0.62$ and $\hat{P}(Succ)_{\mathsf{GT}} = 0.55$. This is again an example of two systems whose probability of satisfaction is larger than it was in principle guessed (about 15% more). But in this case, under the criterion of success, it is shown that one of the systems is about 7% more successful than the other. That is, comparing the systems from a success criterion directly contradicts our conclusions based solely on mean effectiveness.

Measure	Broad				Fine			
	Normal	Trunc.	Beta	Empir.	Normal	Trunc.	Beta	Empir.
CG_l@5	0.9096	1.0872	1.0316	**0.8302**	0.7098	0.7523	**0.7057**	0.8457
CG_e@5	0.9335	1.2225	0.9983	**0.8344**				
DCG_l@5	**0.8208**	1.0406	0.9462	0.833	0.7114	0.7559	**0.7062**	0.8418
DCG_e@5	0.8421	1.1315	0.9488	**0.8379**				
Q_l@5	0.9373	1.3292	0.9584	**0.8403**	0.729	0.7952	**0.7127**	0.8398
Q_e@5	0.875	1.2323	0.9544	**0.833**				
RBP_l@5	**0.8237**	1.0399	0.9494	0.8407	0.711	0.7543	**0.707**	0.8422
RBP_e@5	0.8395	1.123	0.9483	**0.8364**				
GAP@5	**0.8333**	1.14	0.9665	0.8369	0.7227	0.8078	**0.704**	0.8423

Measure	$n_{\mathcal{L}} = 4$				$n_{\mathcal{L}} = 5$			
	Normal	Trunc.	Beta	Empir.	Normal	Trunc.	Beta	Empir.
CG_l@5	**0.8261**	1.0107	0.9179	0.8409	**0.7875**	0.925	0.8513	0.8463
CG_e@5	0.8496	1.1974	0.8973	**0.8428**	0.8476	1.226	**0.8447**	0.8496
DCG_l@5	**0.7773**	0.9892	0.8704	0.8416	**0.7522**	0.9095	0.8171	0.8474
DCG_e@5	**0.7966**	1.1325	0.879	0.8406	**0.8114**	1.1776	0.8338	0.8471
Q_l@5	0.8669	1.2081	0.8863	**0.8438**	**0.8181**	1.0721	0.8409	0.8509
Q_e@5	**0.8159**	1.181	0.8845	0.8429	**0.8213**	1.1999	0.8395	0.8478
RBP_l@5	**0.7751**	0.9836	0.8704	0.8426	**0.75**	0.9039	0.8171	0.8484
RBP_e@5	**0.7936**	1.1247	0.8786	0.8411	**0.8081**	1.1688	0.8329	0.8476
GAP@5	**0.7993**	1.1001	0.8786	0.8427	**0.7743**	1.0067	0.8228	0.8489

Measure	$\ell_{min} = 20$				$\ell_{min} = 40$			
	Normal	Trunc.	Beta	Empir.	Normal	Trunc.	Beta	Empir.
P@5	1.7547	1.0001	0.7669	**0.5754**	1.3241	1.3235	1.1445	**0.738**
AP@5	1.6545	0.9918	0.6199	**0.5638**	1.215	1.3903	1.0474	**0.7496**
DCG_l@5	1.7616	1.1631	0.6217	**0.5621**	1.2878	1.3889	1.0352	**0.7441**
RBP_l@5	1.7574	1.1223	0.6266	**0.5616**	1.2803	1.3669	1.0382	**0.7423**

Table 4.5: Average $\hat{\omega}$ statistics of the fits provided by the Normal, Truncated Normal, Beta and Empirical distributions for $n_{\mathcal{Q}_1} = 20$. Best per measure in bold.

4.4 Discussion

In Chapter 3 I anticipated that a $(\Lambda, \mathcal{L})$ combination for which the *sat* function does not track the $P(Sat) = \lambda$ function does not necessarily mean it is a bad predictor of user satisfaction; it just means that it is not as intuitive and immediate as it might seem. Simple polynomial models were fitted to estimate these *sat* functions for several combinations of effectiveness measure and relevance scale. In terms of measures, the Λ_e formulations with exponential gains were again outperformed by the Λ_l versions with linear gain; estimation errors were slightly lower. Although fits were generally good (mean squared residuals of about 0.02), rank-based measures behaved better than the set-based CG_l@5; especially DCG_l@5, Q_l@5 and RBP_l@5. Within the binary measures, P@5 was clearly better fitted than AP@5. In terms of scales, the Fine scale was again the best one, followed by the $n_{\mathcal{L}} = 5$ and $n_{\mathcal{L}} = 4$ artificial graded scales and the Broad scale; that is, the more relevance levels the better the fit. The artificial binary scales performed worse in general. This suggests the use of the Fine scale alone to gather relevance judgments because it is the one that performs the best

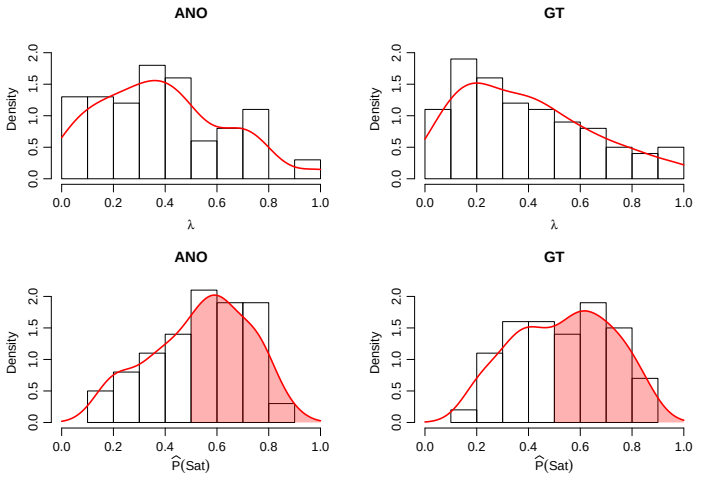

Figure 4.6: Distributions of $DCG_e@5$ scores with $n_{\mathcal{L}} = 5$ (top) and corresponding distributions of $\hat{P}(Sat)$ (bottom) for systems ANO and GT in MIREX 2009.

and, in any case, all other scales may be reproduced from the Fine relevance scores. If we decided to just use the Fine scale, the best measure would be $RBP_l@5$.

In order to better estimate the distribution of effectiveness of a system based on the sample of queries in a test collection, several probability distributions were compared. Results suggested that the Empirical distribution produces the best fit in virtually all $(\Lambda, \mathcal{L})$ cases, provided that the query sample is sufficiently large. For small collections with $n_{\mathcal{Q}} < 30$ queries, the Beta and Normal distributions provide better fits than the Empirical distribution, probably because the resolution of the latter is just too low with that few data points. In terms of measures, the Λ_l variants outperformed the Λ_e variants again. The overall best measures were again $DCG_l@5$ and $RBP_l@5$, and the binary $P@5$ and $AP@5$ behaved alike. In terms of relevance scales, the ones with more relevance levels had better results. This is probably due to the fact that using more relevance levels results in more variability in effectiveness scores; the Normal and Beta distributions are continuous distributions, so they are expected to work better when the underlying distribution is closer to a continuous distribution rather than discrete. In the case of the Empirical distribution, this similarly results in better resolution and fewer ties. The opposite case is that of the binary scales, which have only two relevance levels and thus result in fewer possible outcomes. The continuous distributions are consistently outperformed by a large margin here, because the Empirical distribution does not make continuity assumptions and is more faithful to the discrete data.

4.5 Summary

After running an evaluation experiment, researchers usually report the mean effectiveness of a system over some sample of queries as the indicator of system performance. In this chapter I discussed how to easily report the mean probability of user satisfaction too. To that end, simple polynomial models were fitted to represent the effectiveness-satisfaction mapping that can be used to estimate the distribution of satisfaction scores. Unlike intuition tells us,

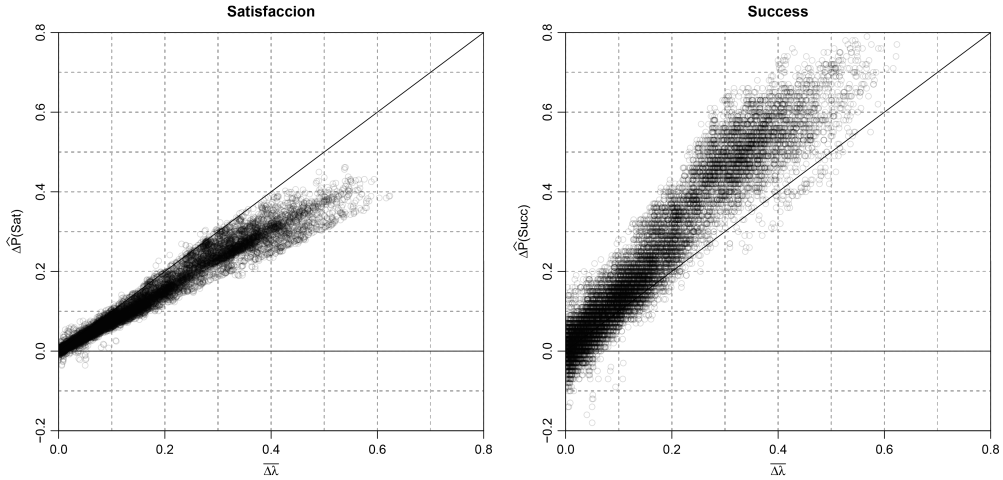

Figure 4.7: Estimated difference in $P(Sat)$ (left) and in $P(Succ)$ (right) as a function of $\overline{\Delta\lambda}$ for all 40 $(\Lambda, \mathcal{L})$ combinations and all pairs of systems from MIREX 2007, 2009, 2010 and 2011.

the relation between effectiveness and satisfaction is not an equality. This means that when researchers conclude that some system is better than another one based on the difference in effectiveness, the opposite conclusion may be reached when considering user satisfaction[4]. At the very least, the difference in user satisfaction will probably be smaller than the difference in effectiveness. The left plot in Figure 4.7 shows the estimated differences between all systems in MIREX 2007, 2009, 2010 and 2011, for all forty $(\Lambda, \mathcal{L})$ combinations studied in this chapter, for a total of 14,080 data points. As the plot shows, differences in user satisfaction are indeed overestimated in general.

Considering the distribution of scores, and not just the mean as usual, has two immediate advantages. First, it allows us to report confidence intervals around the mean estimates of effectiveness and satisfaction. These intervals provide a better report of average system behavior because they account for sampling error and are therefore more informative than reporting just the means as usual. Second, it allows us to report prediction intervals to analyze how extreme systems can be for arbitrary new queries. Consider two systems that yield very similar average satisfaction but differ on variability. A system consistently returning good results should be preferred over a system that sometimes does extremely well but sometimes does extremely bad, even if on average they are similar. As before, we may easily reach contradictory conclusions as well. Analyzing systems from this perspective, allows researchers to focus on queries for which systems perform particularly bad.

Drawing that line between a successful and an unsuccessful system output for some query allows us to disregard differences that are larger than in principle necessary. If we consider a system output successful when it is likely to satisfy the majority of users, we find that comparisons between systems can be quite different from what effectiveness tells us. Similarly, the right plot in Figure 4.7 shows that differences in system success are actually underestimated by differences in effectiveness. This means that when systems seem similar

[4] This can never happen for a single query because satisfaction is proportional to effectiveness, but when averaging over a sample it may happen because relative differences vary for specific queries.

to each other based on the usual reports on effectiveness, it probably is the case that they behave quite differently in reality.

In order to better describe the distribution of scores, several probability distributions were studied in terms of predictive power. Results suggest that the Empirical distribution is generally the best alternative when the query set size is relatively large. However, when there are just a few dozen queries it is better to use the Normal and Beta distributions. This is particularly interesting because one of the objectives of the next chapters is actually to reduce the query set in order to reduce annotation costs. Finally, the Fine scale has proved so far to be the best choice, and has the additional advantage that it allows us to easily reproduce the other scales. Surprisingly, in this case the Beta distribution consistently provided the best fits for all measures and query set sizes.

Chapter 5

Optimality of Statistical Significance Tests

Chapter 2 identified the reliability of an Information Retrieval evaluation experiment as the extent to which conclusions about system performance can be repeated in another experiment. A number of factors are involved, such as query sets and disagreement among human assessors. The usual way to assess reliability is to use a statistical significance test such as the t-test. Unfortunately, these tests are based on assumptions that are knowingly violated in IR evaluation, so it is necessary to study how they are affected. This chapter presents a study to analyze five statistical significance tests from three optimality criteria: power, safety and exactness. Different effectiveness measures and relevance scales are compared too. Finally, I discuss the usual misunderstanding with statistical significance, how it is mistaken with practical significance, and how it can easily mislead researchers.

5.1 Reliable System Comparisons

An Information Retrieval researcher is often faced with the question of which of two IR systems, A and B, performs better. She conducts an experiment with a test collection containing query set $\mathcal{Q}$, and chooses an effectiveness measure such as $RBP@5$ or $nDCG@10$. Based on the average effectiveness difference $\overline{\Delta\lambda}_{\mathcal{Q},\mathsf{AB}}$ she concludes that, for instance, system A is better. But we know there is inherent noise in the evaluation for a wealth of factors such as document collections, query sets, relevance assessors, etc. Therefore the researcher needs the conclusion to be reliable, that is, the observed difference unlikely to have happened just by random chance.

The usual way to proceed is to employ a statistical significance test, which provides a measure of confidence that the observed difference is indeed real and there is an actual difference between the two systems. In such a case, the difference is considered statistically significant (here notated as $\mathsf{A} \overset{*}{\succ} \mathsf{B}$); in practice this means that she can be confident that the difference measured with a similar test collection will be (at least) as large as $\overline{\Delta\lambda}_{\mathcal{Q},\mathsf{AB}}$. If, on the other hand, the difference is not found to be statistically significant (notated simply

as $A \succ B$), she can not be confident that the observed difference is indeed real and should therefore expect any result with a different test collection.

5.1.1 Statistical Hypothesis Testing

In a statistical hypothesis testing scenario a researcher obtains some experimental data from which she wishes to conclude something with certain confidence. She states two mutually exclusive hypotheses concerning the data: the null hypothesis H_0 and the alternative or research hypothesis H_1. For example, the data may be the time taken to solve two different problems P_1 and P_2 by a sample of 100 subjects. The researcher may be interested in knowing if the two problems take the same time to solve; in this case the null hypothesis is $H_0 : \mu_{P_1} = \mu_{P_2}$ and the alternative is $H_1 : \mu_{P_1} \neq \mu_{P_2}$.

In order to decide for one or another hypothesis, the researcher chooses a statistical hypothesis test to use. Different tests deal with different types of hypotheses and make different assumptions regarding the distribution underlying the data, sampling process, etc. Following the test of choice, a test statistic is computed from the experimental data. This test statistic is known to follow some predefined distribution, defined by the test, if the null hypothesis H_0 were actually true. From that distribution the researcher can compute the probability of having observed the experimental data if H_0 were true; this is the p-value. If the p-value is very small, the researcher has strong evidence that the null hypothesis is not true and therefore rejects it in favor of the alternative H_1. In our example, let us assume that the test results in $p = 0.00014$; this is the probability of observing the difference found in our experimental data if the two problems actually took the same time to be solved. Because it is extremely small, we reject this hypothesis and conclude that they do not take the same time to be solved.

The maximum value a p-value can take to reject the null hypothesis is called the significance level, denoted by α. If $p \leq \alpha$ then H_0 is rejected, and if $p > \alpha$ then it is not. Choosing a value for α is a rather arbitrary matter (see Section 5.6), but usually it is set to $\alpha = 0.05$ or $\alpha = 0.01$. That is, if the probability of observing the experimental data is less than 5% or 1% under the null hypothesis, we consider that probability to be too low and consequently reject the null hypothesis in favor of the alternative.

Type I and Type II Errors

The fact that some experimental data is extremely unlikely under the null hypothesis does not necessarily mean that the null hypothesis is false; although unlikely, it is still possible. Let us assume that the two problems in the example above are actually the same, and therefore the null hypothesis is true by definition. If we ran the experiment a number of times, it is expected that $100\alpha\%$ of the times we reject the null hypothesis because, just by random chance, we obtained differences large enough to be considered unlikely. In these cases we would be incorrectly rejecting the null hypothesis, that is, committing a Type I error. If the two problems were indeed different and still we obtained experimental data where they are so similar that the p-value is larger than α, we would be incorrectly accepting the null hypothesis, that is, committing a Type II error. The probability of committing such

		Truth	
		H_0 true	H_0 false
Test	Accept H_0	correct	Type II error
	Reject H_0	Type I error	correct

Table 5.1: Statistical hypothesis testing as a binary decision problem.

an error is usually denoted by β. The probability of correctly rejecting the null hypothesis is therefore $1 - \beta$, which is known as the power of the test.

We may therefore consider statistical hypothesis testing as a binary decision problem in which the null hypothesis is either true or false and, based on a statistical test, we accept it or reject it (see Table 5.1).

5.1.2 Statistical Significance Tests

For the particular case of IR evaluation, we are interested in the comparison of two systems A and B according to their distribution of effectiveness scores with a set of queries $\mathcal{Q}$. Five common statistical significance tests are considered: the paired t-test, the Wilcoxon test, the sign test, the bootstrap test and the permutation test [Smucker et al. 2007]. For simplicity, the notation in this section omits subindices A and B.

Paired t-test

The paired t-test is a parametric test to compare the means of two paired distributions [Student 1908]. The hypotheses are $H_0 : \mu_{\Delta\lambda} = 0$ (both systems have the same mean) and $H_1 : \mu_{\Delta\lambda} \neq 0$ (they do not). The test statistic is based on the distribution of differences observed with the test collection:

$$t = \frac{\overline{\Delta\lambda}_{\mathcal{Q}}}{sd_{\mathcal{Q}}/\sqrt{n_{\mathcal{Q}}}} \tag{5.1}$$

where $sd_{\mathcal{Q}}$ is the sample standard deviation of $\Delta\lambda_q$ scores. The statistic t follows a Student's t distribution with $n_{\mathcal{Q}} - 1$ degrees of freedom. Using the cumulative distribution function of this distribution, the p-value is computed as twice the area to the right of t[1]. The main assumptions underlying the test are that the true distribution of $\Delta\lambda$ scores is normally distributed, and that queries are sampled randomly.

Wilcoxon Signed-Rank Test

The Wilcoxon Signed-Rank test is a non-parametric test to compare the medians of the distributions of effectiveness [Wilcoxon 1945]. The hypotheses are $H_0 : m_{\Delta\lambda} = 0$ and $H_1 : m_{\Delta\lambda} \neq 0$, where $m_{\Delta\lambda}$ is the median effectiveness difference. The test ranks all non-zero $\Delta\lambda_q$ scores by their absolute value. Let W^+ be the sum of ranks corresponding to positive $\Delta\lambda_q$ scores, and similarly W^- be the sum of the negatives. The test statistic is:

$$W = \min\left(W^+, W^-\right) \tag{5.2}$$

[1] Twice because the null hypothesis $H_0 : \mu_{\Delta\lambda} = 0$ is two-tailed. For the one-tailed $H_0 : \mu_{\Delta\lambda} \leq 0$ we keep the area to the right of t.

and it follows a Wilcoxon Signed-Rank distribution with $\sum \mathbb{1}(\Delta\lambda_q \neq 0)$ observations. Using its cumulative distribution function, the p-value is computed as twice the area to the right of W. The assumptions underlying the test are that the $\Delta\lambda$ distribution is symmetrical around 0 and that queries are sampled randomly.

Sign Test

The sign test is a non-parametric test to compare the medians of the distributions of effectiveness [Conover 1999]. The hypotheses are again $H_0 : m_{\Delta\lambda} = 0$ and $H_1 : m_{\Delta\lambda} \neq 0$. The test statistic is based on the number of queries for which the difference is positive:

$$S = \sum_{q \in \mathcal{Q}} \mathbb{1}(\Delta\lambda_q > 0) \tag{5.3}$$

Under the null hypothesis, the probability of observing a positive difference is 0.5, so S follows a Binomial distribution with $\sum \mathbb{1}(\Delta\lambda_q \neq 0)$ trials and probability of success 0.5. The p-value can then be computed as the fraction of cases in which one can observe S successes. The assumption underlying the test is that queries are sampled randomly.

Bootstrap Test–Shift Method

The bootstrap test is a resampling test to compare the means of the distributions of effectiveness [Efron and Tibshirani 1998]. The hypotheses are $H_0 : \mu_{\Delta\lambda} = 0$ and $H_1 : \mu_{\Delta\lambda} \neq 0$. The test attempts to recreate the true $\Delta\Lambda$ distribution by sampling from all $\Delta\lambda_q$ scores. Let B_i be one of T samples of size $n_{\mathcal{Q}}$ built by randomly selecting $\Delta\lambda_q$ effectiveness differences *with* replacement. Let $\overline{B_i}$ be the mean of each sample, and let $\overline{B}$ be the mean of these means. The p-value is computed as:

$$p = \frac{\sum_{i=1}^{T} \mathbb{1}\left(\left|\overline{B_i} - \overline{B}\right| \geq \left|\overline{\Delta\lambda}_{\mathcal{Q}}\right|\right)}{T} \tag{5.4}$$

that is, the fraction of samples for which the shifted mean $\left|\overline{B_i} - \overline{B}\right|$ is at least as large as the observed $\left|\overline{\Delta\lambda}_{\mathcal{Q}}\right|$. The assumption underlying the test is that queries are sampled randomly.

Permutation Test

The permutation or randomization test is a resampling test to compare the means of the distributions of effectiveness [Good 2005]. The hypotheses are again $H_0 : \mu_{\Delta\lambda} = 0$ and $H_1 : \mu_{\Delta\lambda} \neq 0$. The test assumes that under the null hypothesis it is equally likely for $\lambda_{q,\mathsf{A}}$ and $\lambda_{q,\mathsf{B}}$ to be generated by A or B, that is, systems are interchangeable. Let P_i be one of T samples of size $n_{\mathcal{Q}}$ built by randomly and independently swapping the signs of all $\Delta\lambda_q$ scores, and let $\overline{P_i}$ be the mean of each sample. The p-value is computed as:

$$p = \frac{\sum_{i=1}^{T} \mathbb{1}\left(\left|\overline{P_i}\right| \geq \left|\overline{\Delta\lambda}_{\mathcal{Q}}\right|\right)}{T} \tag{5.5}$$

that is, the fraction of samples for which the mean $\left|\overline{P_i}\right|$ is at least as large as the observed $\left|\overline{\Delta\lambda}_{\mathcal{Q}}\right|$. The assumption underlying the test is that under the null hypothesis systems are interchangeable, that is, that both samples are generated from the same distribution and observations are arbitrarily assigned to one or another system.

5.1.3 Optimality Criteria

There has been a debate regarding statistical significance testing in IR evaluation. Classical tests such as the t-test, the Wilcoxon test and the sign test make different assumptions about the distributions, measurement levels and sampling methods, and distributions from IR evaluations are known to violate these assumptions. The bootstrap test is an alternative that makes fewer assumptions and has other advantages over classical tests, and the permutation test is an even less stringent test in terms of assumptions that theoretically provides exact p-values. Because IR evaluations violate most of the assumptions, it is very important to know how robust these tests are in practice and which one is optimal.

Previous work by Smucker et al. [2007] compared these five tests with TREC Ad Hoc data, reaching the following conclusions:

- The bootstrap, t-test and permutation test largely agree with each other, so there is hardly any practical difference in using one or another.
- The permutation test should be the test of choice, though the t-test seems suitable as well; the bootstrap test shows a bias towards small p-values.
- The Wilcoxon and sign tests are unreliable and should be discontinued.

However, all these conclusions were based on the assumption that the permutation test is optimal. For example, authors showed that the Wilcoxon and sign tests fail to detect significance when the permutation test does and vice versa. That is, they are unreliable *according to* the permutation test.

But based on the logics of hypothesis testing we may follow different criteria to chose an optimal test. We may want the test to be *powerful* and produce significant results as often as possible to avoid Type II errors. Additionally, we may want it to be *safe* and yield low Type I error rates so that it is unlikely that we draw wrong conclusions by incorrectly rejecting null hypotheses. But power and safety are inversely related, and different tests show different relations depending on the significance level α. The lower α the lower the power and the safer the test, because we need $p \leq \alpha$ for the result to be significant. Error rates are expected to be at the nominal α level, so the higher the significance level the higher the power, but the higher expected error rate too. The test is *exact* if we can trust that the actual error rate is as dictated by the significance level. If the error rate is below the nominal level it means we are being too conservative and we are missing significant results; if it is above it means we are deeming as significant results that probably are not. In general, we want to use the most powerful test that maintains the error rates at the expected level.

5.2 Effectiveness Measures and Relevance Scales

The effectiveness measures studied so far were formulated in a user-oriented way, so that they resulted in $\lambda = 1$ only when the system returned ideal results according to their user model (see Section 3.1). These user-oriented formulations were followed to establish the effectiveness-satisfaction mapping. However, when using a test collection there is a possibility that the relevance judgments do not have the necessary characteristics to produce $\lambda = 1$. As an example, the definition of $AP@k$ in (3.1) requires the system to retrieve k relevant documents to achieve $AP@k = 1$, but if the total number of relevant documents

in the collection is $\left|\mathcal{R}^1\right| < k$, it is impossible for any system to obtain $AP@k = 1$. That is, effectiveness scores depend on the characteristics of the test collection, and thus user satisfaction depends on the actual set of documents that are available for systems to retrieve.

Consider a query q_1 for which there are 5 relevant documents in the collection. System A retrieves documents with relevance $\langle 1, 1, 1, 1, 1\rangle$, while system B retrieves documents with relevance $\langle 1, 1, 1, 1, 0\rangle$. In this scenario we have $\Delta AP@5_{q_1,\mathsf{AB}} = 1 - 0.8 = 0.2$ with the user-oriented formulation in (3.1). Let us now consider query q_2 for which there is only one relevant document; system A retrieves that document at the top of the list, but system B does not retrieve it. In this case, we would have $\Delta AP@5_{q_2,\mathsf{AB}} = 0.2 - 0 = 0.2$. In both situations the difference in effectiveness is 0.2, but the qualitative difference is substantially larger in q_2. With query q_1 both systems performed indeed very similarly, but for query q_2 system A did the best that could possibly be done with that query *and the set of documents in the collection*; system B failed at retrieving whatever few relevant documents there are. Under these circumstances, the system-oriented definition of $AP@k$ in (2.3) would have yielded $\Delta AP@5_{q_1,\mathsf{AB}} = 1 - 0.8 = 0.2$ and $\Delta AP@5_{q_2,\mathsf{AB}} = 1 - 0 = 1$, showing that the difference is actually larger with the second query. Therefore, when comparing systems with a test collection we must account for the characteristics of the document set, in particular the known relevance judgments.

5.2.1 System-Oriented Effectiveness Measures

The effectiveness measures considered in this chapter follow their original formulations as in Section 2.5, although I still modify them slightly so that all λ scores are normalized between 0 and 1 for the sake of comparison. Additionally, and based on the results from Chapter 3 and Chapter 4, only the Λ_l variants with linear gain functions are considered. This is also supported by Kanoulas and Aslam [2009], who found both the linear and exponential gain functions to be similarly reliable with TREC data when using $nDCG@k$.

Binary Relevance Scale

Precision. No modification is needed in this case because the measure is already normalized. The formulation used is (2.2).

Average Precision. The original formulation in (2.3) will always yield $AP@k < 1$ if the number of relevant documents in the collection is $\left|\mathcal{R}^1\right| > k$, even if the system only retrieves relevant documents. This can have a negative effect if there are too many relevant documents. In the example above, if there were 1,000 relevant documents for q_1, we would only have $\Delta AP@5_{q_1,\mathsf{AB}} = 0.005 - 0.004 = 0.001$. The formulation used here is:

$$AP@k = \frac{1}{\min(k, |\mathcal{R}^1|)} \sum_{i=1}^{k} r_{\mathsf{A}_i} \cdot P@i \tag{5.6}$$

so that a system retrieving only relevant documents would obtain $AP@k = 1$, regardless of how many relevant documents there are in the collection. This is the formulation implemented for example in the `ntcireval` evaluation package used in NTCIR.

Graded Relevance Scale

Cumulative Gain. The formulation used in this chapter is the same as in (3.3), that normalizes scores between 0 and 1.
Discounted Cumulative Gain. Similarly, the bounded formulation in (3.4) is used.
Normalized Discounted Cumulative Gain. In Chapter 3 I ignored $nDCG@k$ because it is not expected to correlate to user satisfaction (see Section 3.1), but in this chapter I do consider it. The formulation followed is (2.10) with the $d(i) = \log_2(i+1)$ discount function.
Q-Measure. Similar to the modification for $AP@k$, the score normalization accounts for the number of relevant documents in the collection. The formulation used here is:

$$Q@k = \frac{1}{\min(k, |\mathcal{R}^{>0}|)} \sum_{i=1}^{k} \mathbb{1}(r_{\mathsf{A}_i} > 0) \frac{\sum_{j=1}^{i} \mathbb{1}\left(r_{\mathsf{A}_j} > 0\right) + \beta \cdot \sum_{j=1}^{i} g\left(r_{\mathsf{A}_j}\right)}{i + \beta \cdot \sum_{j=1}^{i} g\left(r_{\mathsf{I}_j}\right)} \tag{5.7}$$

so that a system retrieving only relevant documents obtains $Q@k = 1$ regardless of how many relevant documents there are in the collection. This is again the formulation in `ntcireval`.
Rank-Biased Precision. Formulation (3.6) used in Chapter 3 already modified the original one in (2.12) to consider a cutoff k, but normalized by considering the relevance judgments in the ideal ranking to be $\langle n_{\mathcal{L}}-1, n_{\mathcal{L}}-1, \ldots, n_{\mathcal{L}}-1\rangle$. Here we normalize by considering the ideal ranking to be the best that can possibly be done with the known judgments:

$$RBP@k = \frac{\sum_{i=1}^{k} g(r_{\mathsf{A}_i}) \cdot p^{i-1}}{\sum_{i=1}^{k} g(r_{\mathsf{I}_i}) \cdot p^{i-1}} \tag{5.8}$$

Graded Average Precision. The formulation used in this chapter is based on the original one in (2.16), considering the known judgments in the collection to normalize scores. However, and similarly to $AP@k$ and $Q@k$, it is normalized by considering the best a system can do with k documents rather than all known judgments:

$$GAP@k = \frac{\sum_{i=1}^{k} \mathrm{E}[P@i]}{\sum_{\ell=1}^{n_{\mathcal{L}}-1} \sum_{i=1}^{k} \mathbb{1}(r_{\mathsf{I}_i} = \ell) \sum_{s=1}^{\ell} p_\ell} \tag{5.9}$$

That is, instead of counting how many documents are judged with level ℓ (i.e. $\left|\mathcal{R}^{\ell}\right|$), we count how many of the top k documents in the ideal ranking I are judged with level ℓ (i.e. $\sum_{i=1}^{k} \mathbb{1}(r_{\mathsf{I}_i} = \ell)$).

5.2.2 Relevance Scales

Based on the results from Chapter 3 and Chapter 4, I consider here only the original Broad and Fine scales, and also include the artificial $\ell_{min} = 40$ binary scale for completeness. Table 5.2 lists all fourteen $(\Lambda, \mathcal{L})$ combinations studied.

5.3 Data and Methods

To compare the five statistical significance tests at hand, I employed data from the MIREX 2007, 2009, 2010 and 2011 AMS task because they all used a different set of 100 queries. The document collection was the same in all four editions, and all queries were randomly

Measure	Broad	Fine	$\ell_{min}=40$
P@5			x
AP@5			x
CG_l@5	x	x	P@5
DCG_l@5	x	x	
$nDCG_l$@5	x	x	
Q_l@5	x	x	AP@5
RBP_l@5	x	x	
GAP@5	x	x	AP@5

Table 5.2: All 14 combinations of effectiveness measures and relevance scales studied (marked with x), and equivalent combinations (e.g. Q_l@5 is the same as AP@5 with the $\ell_{min} = 40$ scale).

selected audio clips from the document collection itself. We can therefore consider all these queries as coming from the same universe of queries.

For each test collection, the 100 queries were split into two disjoint subsets of 50 queries each: $\mathcal{Q}_1$ and $\mathcal{Q}_2$. For each of these two subsets all systems were evaluated as per each $(\Lambda, \mathcal{L})$ combination. This provided us with a number of system pairwise comparisons as per $\mathcal{Q}_1$ and similarly as per $\mathcal{Q}_2$. In particular, there are 66 system pairwise comparisons in 2007, 105 in 2009, 28 in 2010 and 153 in 2011 for a total of 352 comparisons. All five statistical significance tests were run between each of these system pairs[2]. This gives us a total of 352 pairs of p-values per test, which can be regarded as the two p-values observed with two different test collections of size 50 for any two systems. We performed 300 random trials of this experiment, so there are a total of 105,600 system pairwise comparisons and the corresponding 105,600 with another query subset. The same was repeated for each of the fourteen $(\Lambda, \mathcal{L})$ combinations of interest, leading to a total of 1,478,400 pairs of p-values per test, and therefore a grand total of nearly 15 million p-values for all test collections, measures, scales and statistical significance tests.

Given an arbitrary query set split, the 352 pairs of p-values provided by a test can be used to study its optimality. Consider a researcher that used query subset $\mathcal{Q}_1$ and ran a test to compute a p-value; under the significance level α she draws a conclusion. What can she expect with a different query set $\mathcal{Q}_2$? One of these situations can occur:

- **Non-significance.** The result with $\mathcal{Q}_1$ is $\mathsf{A} \succ \mathsf{B}$. We can really expect any result with $\mathcal{Q}_2$; there is a lack of statistical power in the experiment.
- **Success.** The result with both $\mathcal{Q}_1$ and $\mathcal{Q}_2$ is $\mathsf{A} \overset{*}{\succ} \mathsf{B}$. Both experiments show evidence of one system outperforming the other.
- **Lack of power.** The difference is $\mathsf{A} \overset{*}{\succ} \mathsf{B}$ with $\mathcal{Q}_1$ but it is $\mathsf{A} \succ \mathsf{B}$ with $\mathcal{Q}_2$. There is evidence of a lack of power in the second experiment.
- **Minor conflict.** The result with $\mathcal{Q}_1$ is $\mathsf{A} \overset{*}{\succ} \mathsf{B}$, but with $\mathcal{Q}_2$ it is $\mathsf{A} \prec \mathsf{B}$. The second experiment shows some weak evidence of a wrong conclusion in the first one.
- **Major conflict.** The result with $\mathcal{Q}_1$ is $\mathsf{A} \overset{*}{\succ} \mathsf{B}$, but with $\mathcal{Q}_2$ it is $\mathsf{A} \overset{*}{\prec} \mathsf{B}$. The two experiments conflict with each other.

[2] As in [Smucker et al. 2007, 2009], I calculated 100,000 samples in the permutation and bootstrap tests for a resolution in p of 0.00001.

A powerful test minimizes the non-significance rate, a safe test minimizes the minor and major conflict rates, and an exact test keeps the global conflict rate at the nominal α level.

5.4 Results

The full set of nearly 15 million paired p-values was analyzed from two perspectives. First, all measures, scales and collections were joined together for the purpose of analyzing the optimality of each significance test separately. Second, all tests and collections were joined together for the purpose of analyzing the optimality of effectiveness measures and relevance scales separately. In all cases, as many as 32 significance levels are considered $\alpha \in \{0.0001, \ldots, 0.0009, 0.001, \ldots, 0.009, \ldots, 0.1, \ldots, 0.5\}$.

5.4.1 Optimal Statistical Significance Test

For every statistical significance test I computed the non-significance, success, lack of power and conflict rates joining together all measures, scales and collections. Table 5.3 reports the results for a selection of significance levels, and Figure 5.1 plots detailed views in the full range. Please note that all plots are log-scaled.

Non-significance rate. The bootstrap test consistently produces smaller p-values and is therefore the most powerful of all tests across significance levels. Next are the permutation test, t-test and Wilcoxon test, though differences are just less than 1% fewer significant results at the usual $\alpha = 0.05$ level; all tests yield significance in about 67% of the cases. The sign test is by far the least powerful of all five.

Success rate. The Wilcoxon test is the most successful of all for $\alpha \leq 0.001$, followed alternatively by the t-test and the bootstrap and permutation tests. The bootstrap test then performs best for $0.001 < \alpha < 0.03$, and from that point on the permutation and t-test outperform the others. In general, all these tests perform considerably well; almost 90% of all significant results are replicated with the second query subset at the usual α levels. The sign test is clearly the worst of all again.

Lack of power rate. Most of the unsuccessful comparisons are due to a lack of power with the second query subset; the sign of the difference is the same, but it is not statistically significant. Relative results are comparable to results on success: the Wilcoxon test dominates at small significance levels and the bootstrap test dominates at the usual levels, again followed by the permutation test and the t-test.

Minor conflict rate. Surprisingly, the sign test produces the smallest conflict rates almost consistently across α levels, and it is therefore the safest of all tests. The Wilcoxon test dominates next for virtually all levels, followed by the t-test. The bootstrap test consistently produces more conflicts than the others, but at the usual $\alpha = 0.05$ all tests produce conflicts in slightly over 1% of significant cases. The permutation test generally produces the second largest conflict rate.

Major conflict rate. It is noticeable that for small significance levels no test shows any major conflict at all. For instance, at $\alpha = 0.003$ the t-test provides as many as 790,636 (54%) significant comparisons, and yet none of them results in a major conflict with the second query subset. In general, the Wilcoxon test has the best rates at $\alpha < 0.03$. The sign

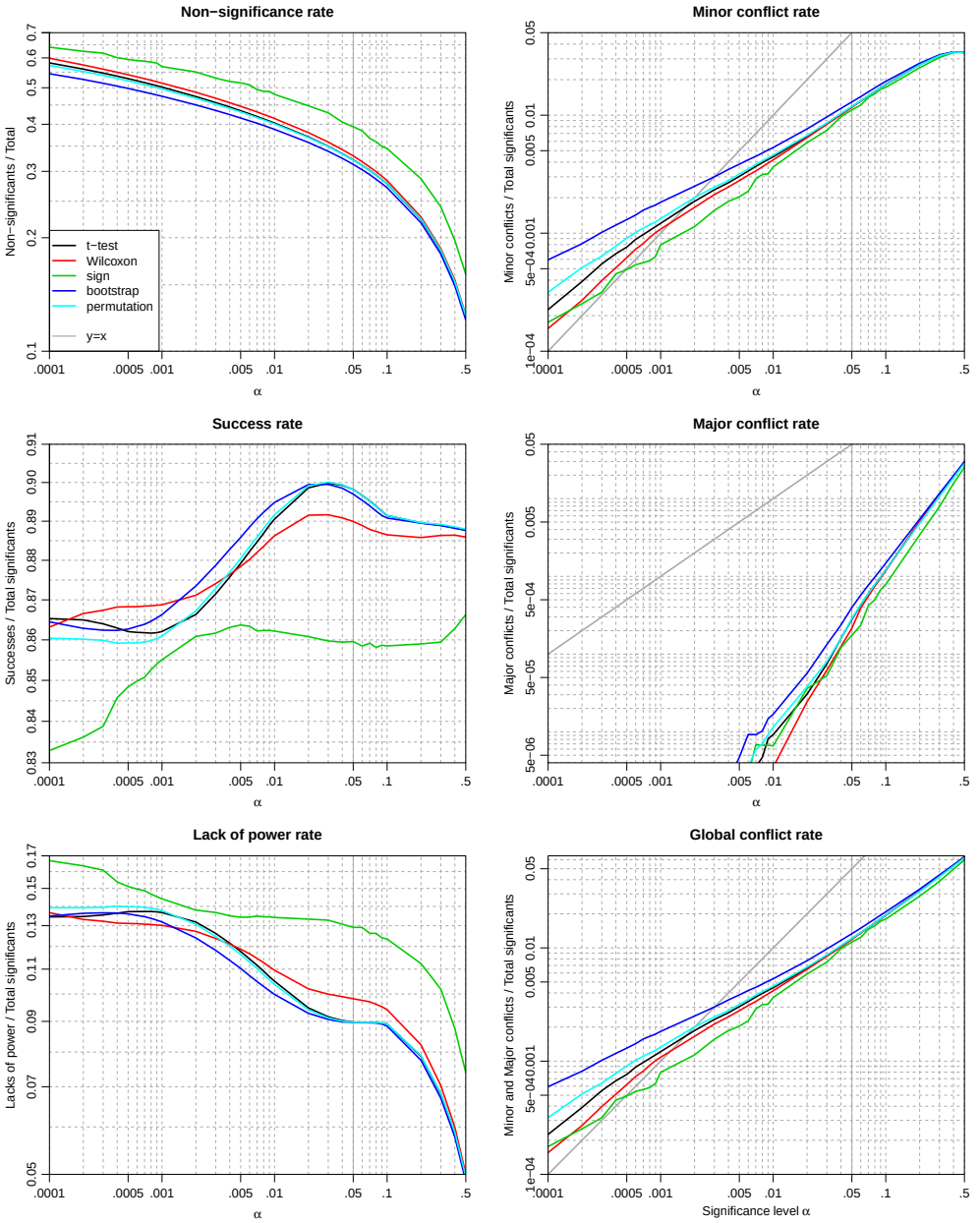

Figure 5.1: Left: non-significance rates (lower is better), success rates (higher is better), and lack of power rates (lower is better). Right: minor conflict rates (lower is better), major conflict rates (lower is better), and global conflict rates ($rate = \alpha$ is better). All rates per significance test.

test then takes over, followed alternatively by the Wilcoxon and permutation tests. The bootstrap test performs worse overall. It is important to bear in mind the magnitudes of the major conflict rates. For instance, at $\alpha = 0.05$ the t-test produced 275 major conflicts and the bootstrap test produced 397. While the difference may seem small compared to the total of significants (0.02787% vs 0.03972%), this is actually a large +44% increase.

α	Non-significance rate					Minor Conflict rate				
	t-test	Wilcox.	sign	boot.	perm.	t-test	Wilcox.	sign	boot.	perm.
0.0001	0.5821	0.5988	0.6406	**0.5447**	0.5732	0.0002	**0.0002**	0.0002	0.0006	0.0003
0.0005	0.5286	0.5413	0.5945	**0.4978**	0.5216	0.0008	0.0006	**0.0005**	0.0013	0.0009
0.001	0.502	0.5144	0.569	**0.4747**	0.4961	0.0012	0.0011	**0.0008**	0.0018	0.0013
0.005	0.4345	0.4464	0.5151	**0.4156**	0.4312	0.003	0.0028	**0.002**	0.0038	0.0032
0.01	0.4032	0.4143	0.4802	**0.3878**	0.401	0.0044	0.0042	**0.0036**	0.0053	0.0046
0.05	0.3219	0.3298	0.394	**0.3131**	0.3215	0.0118	0.0117	**0.0111**	0.013	0.0118
0.1	0.2782	0.2836	0.3449	**0.2715**	0.2784	0.0183	0.0182	**0.0172**	0.0193	0.0182
0.5	0.1231	0.1249	0.1601	**0.121**	0.1242	0.034	0.034	0.0345	**0.0339**	0.034

α	Success rate					Major Conflict rate				
	t-test	Wilcox.	sign	boot.	perm.	t-test	Wilcox.	sign	boot.	perm.
0.0001	**0.8653**	0.8632	0.8329	0.8645	0.8604	**0**	**0**	**0**	**0**	1.6e-6
0.0005	0.8621	**0.8683**	0.8484	0.8627	0.8593	**0**	**0**	**0**	**0**	1.4e-6
0.001	0.8621	**0.8688**	0.8551	0.8663	0.8609	**0**	**0**	**0**	**0**	1.4e-6
0.005	0.8792	0.8784	0.8637	**0.8858**	0.8805	**1.2e-6**	1.2e-6	1.4e-6	4.7e-6	3.6e-6
0.01	0.8905	0.8863	0.8622	**0.8949**	0.8917	9.2e-6	**3.5e-6**	6.6e-6	1.7e-5	1.1e-5
0.05	**0.8982**	0.8899	0.8596	0.897	0.8982	0.0003	0.0002	**0.0002**	0.0004	0.0003
0.1	0.8915	0.8865	0.8585	0.8908	**0.8915**	0.0012	0.0012	**0.0008**	0.0015	0.0012
0.5	0.8877	0.8859	0.8663	0.8876	**0.888**	0.0289	0.0299	**0.0253**	0.0298	0.0284

α	Lack of Power rate					Global Conflict rate				
	t-test	Wilcox.	sign	boot.	perm.	t-test	Wilcox.	sign	boot.	perm.
0.0001	**0.1345**	0.1367	0.1669	0.1349	0.1393	0.0002	**0.0002**	0.0002	0.0006	0.0003
0.0005	0.1372	**0.1311**	0.1511	0.136	0.1398	0.0008	0.0006	**0.0005**	0.0013	0.0009
0.001	0.1367	**0.1302**	0.1441	0.1319	0.1377	0.0012	**0.0011**	0.0008	0.0018	0.0013
0.005	0.1178	0.1188	0.1343	**0.1103**	0.1164	0.003	0.0028	0.002	**0.0038**	0.0032
0.01	0.105	0.1096	0.1342	**0.0998**	0.1037	0.0044	0.0042	0.0036	**0.0053**	0.0046
0.05	0.0897	0.0982	0.1292	**0.0897**	0.0897	0.0121	0.0119	0.0113	**0.0133**	0.0121
0.1	0.0891	0.0941	0.1235	**0.0884**	0.0891	0.0195	0.0194	0.018	**0.0208**	0.0195
0.5	0.0494	0.0501	0.0739	**0.0487**	0.0496	0.0629	**0.064**	0.0597	0.0637	0.0625

Table 5.3: Left: non-significance rates (lower is better), success rates (higher is better), and lack of power rates (lower is better). Right: minor conflict rates (lower is better), major conflict rates (lower is better), and global conflict rates ($rate = \alpha$ is better). All rates per significance test. Best per α in bold.

Global conflict rate. Aggregating minor and major conflicts we have a global conflict rate that can be used as an overall indicator of test safety and exactness. Given the relative size of minor and major conflict rates, the trends are here nearly the same as with minor conflicts. The sign and Wilcoxon tests approximate better the nominal error rate for low significance levels, but the bootstrap test does better for the usual levels.

5.4.2 Optimal Effectiveness Measure and Relevance Scale

For every $(\Lambda, \mathcal{L})$ combination I computed the non-significance, success, lack of power and conflict rates joining together all collections and statistical significance tests. Table 5.4

Measure	Broad					
	Non-sig.	Success	Lack Power	Minor	Major	Global
CG_l@5	**0.32463**	**0.89026**	**0.09815**	**0.01137**	**0.00022**	0.01159
DCG_l@5	0.33638	0.8855	0.10287	0.01139	0.00024	0.01162
$nDCG_l$@5	0.33863	0.88331	0.10464	0.01175	0.0003	0.01205
Q_l@5	0.34395	0.88646	0.1014	0.01182	0.00032	**0.01214**
RBP_l@5	0.33675	0.88603	0.10223	0.01145	0.00029	0.01174
GAP@5	0.33667	0.88615	0.10213	0.01141	0.00032	0.01173

Measure	Fine					
	Non-sig.	Success	Lack Power	Minor	Major	Global
CG_l@5	**0.31743**	**0.89929**	**0.08932**	0.01115	0.00024	0.01139
DCG_l@5	0.32768	0.89527	0.09276	0.01175	0.00022	0.01197
$nDCG_l$@5	0.32992	0.89205	0.0954	0.01236	**0.0002**	0.01255
Q_l@5	0.34365	0.89036	0.0973	0.01202	0.00032	0.01234
RBP_l@5	0.32869	0.89285	0.09457	0.01233	0.00025	**0.01258**
GAP@5	0.32614	0.89546	0.09338	**0.01086**	0.0003	0.01116

Measure	$\ell_{min} = 40$					
	Non-sig.	Success	Lack Power	Minor	Major	Global
P@5	**0.35502**	**0.88**	**0.1061**	0.01362	**0.00027**	**0.0139**
AP@5	0.3588	0.87586	0.11097	**0.01284**	0.00032	0.01317

Table 5.4: Non-significance rates (lower is better), success rates (higher is better), lack of power rates (lower is better), minor conflict rates (lower is better), major conflict rates (lower is better), and global conflict rates ($rate = \alpha$ is better) for all measures and scales at $\alpha = 0.05$. All rates per measure and scale. Best per rate in bold face.

reports the results for the usual $\alpha = 0.05$, and Figure 5.2 and Figure 5.3 plot detailed views in the $\alpha \in [0.001, 0.1]$ range. Please note that all plots are again log-scaled.

Non-significance rate. Consistently across scales and α levels, CG_l@5 is the most powerful measure of all, followed by GAP@5, DCG_l@5 and RBP_l@5. All measures perform better with the Fine scale than with the Broad scale, though Q_l@5 behaves very similarly. The binary scale performs worse, and P@5 consistently outperforms AP@5.

Success rate. CG_l@5 shows the best success rates, followed by GAP@5 and DCG_l@5. Likewise, the Fine scale outperforms the Broad and binary scales, except in the case of Q_l@5. AP@5 shows once more worse rates than P@5.

Minor conflict rate. Q_l@5 shows the best performance within the Broad scale for small α levels, with DCG_l@5 as second best measure. For $0.002 \leq \alpha < 0.05$ DCG_l@5 performs best, alternatively followed by $nDCG_l$@5 and RBP_l@5. Finally, GAP@5 takes over for large levels, followed by CG_l@5. Within the Fine scale, Q_l@5 outperforms all other measures for $\alpha \leq 0.01$, followed by DCG_l@5 and RBP_l@5. For $\alpha > 0.01$ CG_l@5 and GAP@5 clearly perform better, followed by DCG_l@5. Within the binary scale P@5 performs better for low α levels, and AP@5 does better for the usual levels. The Fine and Broad scales behave similarly overall, and they both outperform the binary scale.

Major conflict rate. Within the Broad scale CG_l@5 has the lowest rates, followed by DCG_l@5. RBP_l@5 and $nDCG_l$@5 alternate next. Within the Fine scale, DCG_l@5,

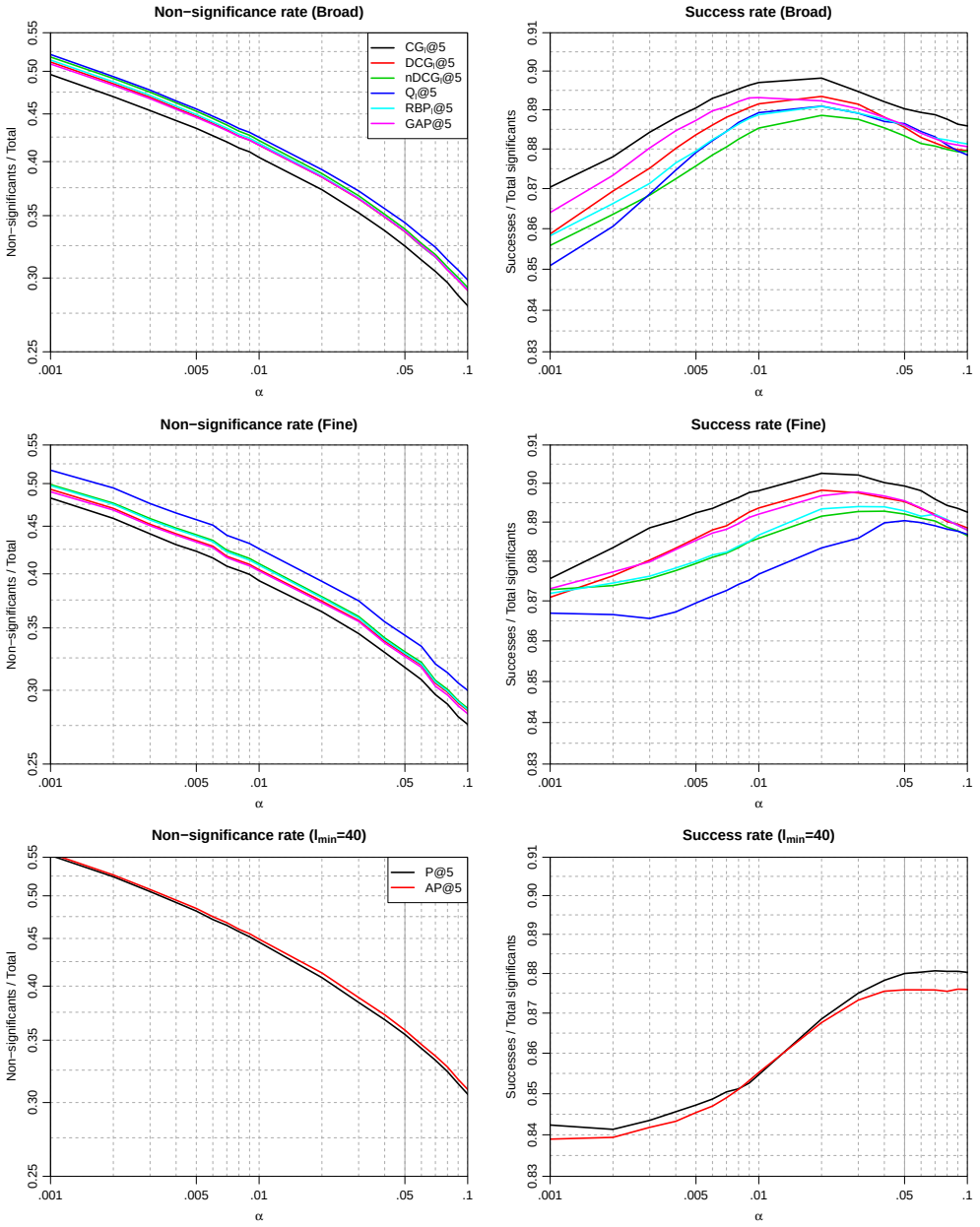

Figure 5.2: Non-significance rates (left, lower is better) and success rates (right, higher is better) for all measures and scales. All rates per measure and scale.

$nDCG_l$@5 and CG_l@5 alternate as best measure across α levels. Within the binary scale, P@5 consistently outperforms AP@5. The binary scale is again the worst of all three scales, and there is not a clear difference between the Broad and Fine graded scales. It is particularly remarkable that neither DCG_l@5 nor $nDCG_l$@5 produce any major conflict at all with the Fine scale until $\alpha = 0.02$.

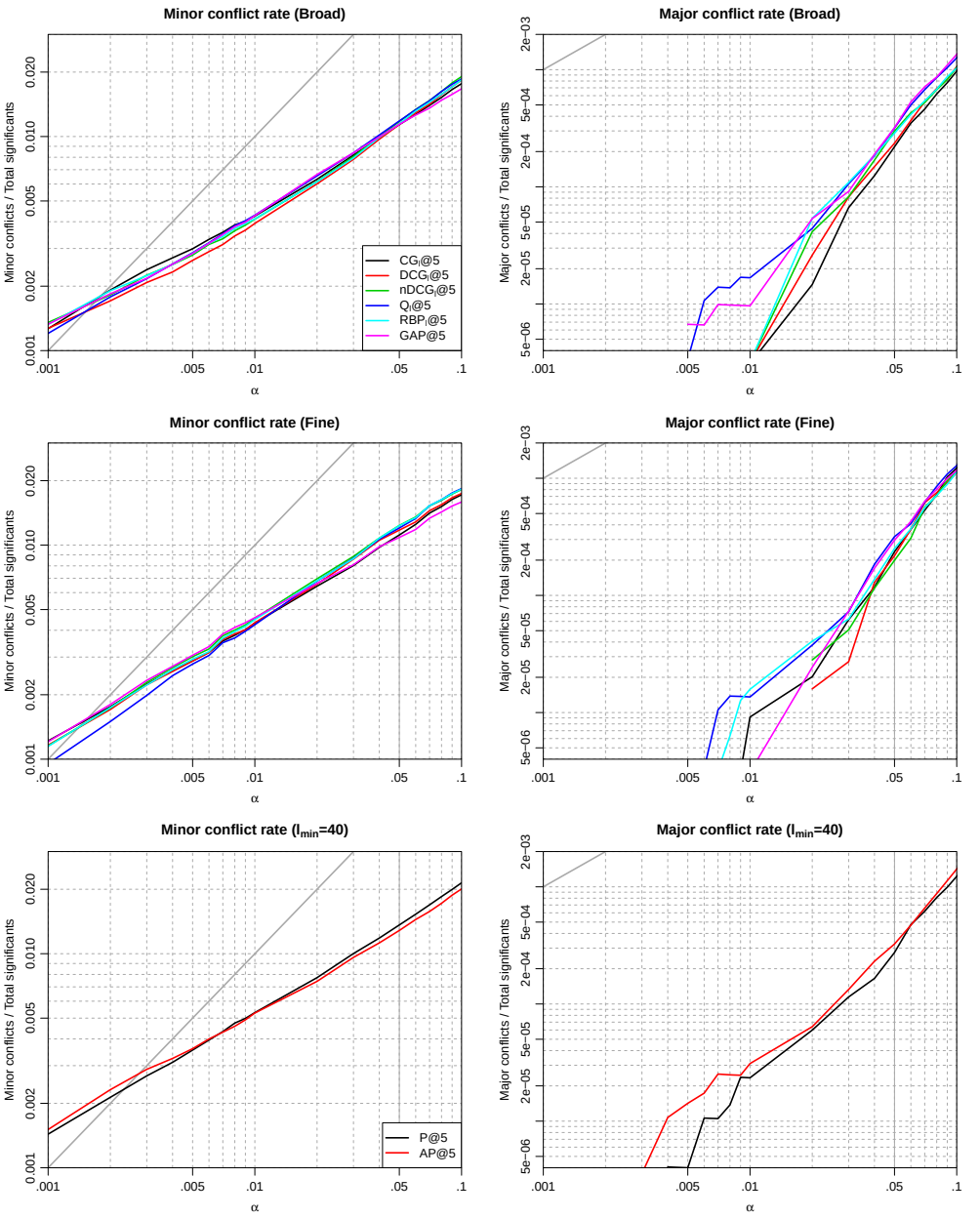

Figure 5.3: Minor conflict rates (left, lower is better) and major conflict rates (right, lower is better) for all measures and scales. All rates per measure and scale.

5.5 Discussion

The issue of statistical significance in IR evaluation has been previously tackled in Text IR, primarily with TREC data. Hull [1993] and Savoy [1997] provide early arguments supporting the use of statistical significance tests to draw reliable conclusions. Similar works can be found in the Music IR literature [Flexer 2006].

Zobel [1998] compared the t-test, Wilcoxon test and ANOVA at $\alpha = 0.05$, though with only one random split in 25-25 queries. He found lower conflict rates with the t-test than with the Wilcoxon test, and generally lower than the nominal 0.05 level. Given that the latter showed higher power and has more relaxed assumptions, he recommended it over the t-test. Sanderson and Zobel [2005] ran a larger study also with splits of up to 25-25 queries. They found that the sign test has higher conflict rates than the Wilcoxon test, which has itself higher conflict rates than the t-test. They also suggested that the actual conflict rate is below the nominal 0.05 level when using 50 query sets. Voorhees [2009] also observed conflict rates below the nominal 0.05 level for the t-test, but more unstable effectiveness measures resulted in higher rates. Cormack and Lynam [2007] used 124-124 query splits and various significance levels. They found the Wilcoxon test more powerful than the t-test and sign test; and the t-test safer than the Wilcoxon and sign test. Sakai [2006] proposed bootstrapping to compare effectiveness measures, but did not compare it with other tests.

Smucker et al. [2007] compared the same tests I study in this chapter, arguing that the t-test, permutation and bootstrap tests largely agree with each other. Nonetheless, they report rooted mean squared errors among their p-values of roughly 0.01, which is a large 20% for p-values of 0.05. Based on the argument that the permutation test is theoretically exact, they conclude that the Wilcoxon and sign tests are unreliable and suggest that they should be discontinued for IR evaluation. They find the bootstrap test to be overly powerful, and given the appealing theoretical exactness of the permutation test they propose its use over the others, though the t-test admittedly performed very similarly. In a later paper, Smucker et al. [2009] found that the t-test, bootstrap and permutation tests tended to disagree with smaller query sets, though the t-test still showed acceptable agreement with the permutation test, again assumed to be optimal. The bootstrap test tended again to produce smaller p-values, so authors recommend caution if using it.

Urbano et al. [2013a] conducted a large-scale study similar to this chapter but with TREC data and 50-50 query splits. They found the bootstrap test to be the most powerful of all tests, the t-test to be the safest and the Wilcoxon test to be the most exact; the permutation test was suboptimal according to all criteria. They also confirmed that all tests produce conflict rates below the nominal α level, therefore suggesting the use of the bootstrap test because it is the most powerful and yet the conflict rate is below expected. For large meta-evaluation studies, they suggested the t-test and Wilcoxon test because they are not computationally intensive as the bootstrap test.

This chapter presents a similar study to revisit these issues under different optimality criteria and for the particular case of AMS. Unlike previous works, I compared not only statistical significance tests, but also effectiveness measures and relevance scales. The results confirmed again the bootstrap test as the most powerful of all five. In terms of safety, the sign test is surprisingly the best of all tests, generally followed by the Wilcoxon test and the t-test. In general, all of them yielded conflict rates higher than expected for low significance levels, but much lower for the usual levels. This suggests that we are being too conservative when assessing statistical significance at $\alpha = 0.05$; we expect 5% of our conclusions to be significant but wrong, yet in practice only about 1.2% seem to be so. I therefore suggest to use the bootstrap test because it is the most powerful of all tests and yet the conflict rates are lower than expected. I must note though that this global conflict rate, as the

Bin	t-test	Wilcox.	sign	boot.	perm.
$p \leq .0001$	0.02604	**0.02505**	0.04073	0.0355	0.02878
$.0001 < p \leq .0005$	0.09423	**0.09103**	0.12708	0.12278	0.09904
$.0005 < p \leq .001$	0.12957	**0.12876**	0.15981	0.15484	0.13443
$.001 < p \leq .005$	0.17727	**0.17562**	0.21543	0.20027	0.18277
$.005 < p \leq .01$	**0.23258**	0.23412	0.27123	0.24833	0.23613
$.01 < p \leq .05$	0.29333	**0.29305**	0.33612	0.3039	0.2963
$.05 < p \leq .1$	0.32991	**0.328**	0.37381	0.333	0.3316
$.1 < p \leq .5$	0.32183	**0.32052**	0.35652	0.32348	0.32261

Table 5.5: RMS error of all five tests with themselves (lower is better). Best per bin in bold.

sum of minor and major conflicts, is just an *approximation* of the true Type I error rates[3] [Cormack and Lynam 2007]. This also begs the reconsideration of procedures to correct p-values for multiple comparisons [Carterette 2012, Boytsov et al. 2013]. For instance, Urbano et al. [2011c] show that, in the particular case of AMS, using 50 queries with a Wilcoxon test is almost as reliable as using 100 queries and Tukey's HSD test (as used in MIREX), evidencing the latter is too conservative [Seaman et al. 1991].

In terms of exactness, the sign and Wilcoxon tests were shown to be the best choice at low α levels, while the bootstrap test was best for the usual levels. However, the test that performs best overall and across α levels is the Wilcoxon test. One way to support this choice is by measuring the relative rooted mean squared error across α levels between the global conflict rates and the nominal level; the Error in the Wilcoxon test is 0.5836, followed by the sign test with 0.6221 and the t-test with 0.6668. Another way is to measure the stability of the test as its agreement with itself. Table 5.5 shows the agreement of the five tests with themselves: p-values with query subset $\mathcal{Q}_1$ compared to those with subset $\mathcal{Q}_2$. The Wilcoxon test turns out to be the most stable of all almost consistently across α levels, followed by the t-test and the permutation test.

In general, these results with AMS data are comparable with those of the Text IR literature in relative terms. The sign test can be discarded for being too conservative, and the bootstrap test is the most powerful of all. The t-test and Wilcoxon test are the safest, and the Wilcoxon test is generally the most exact. The permutation test is not found to be optimal under any of these criteria. There are differences though in absolute terms. The tests are more powerful and successful with the AMS data, and conflict rates are generally lower too. The difference in power should be explained by system differences being larger in AMS than in Text IR. The differences in success and conflicts are probably due to one key detail that makes the AMS evaluation more stable: queries *are* sampled randomly in the MIREX AMS test collections, while this is generally *not* the case in Text IR.

5.6 Statistical Significance and Practical Significance

A researcher is usually interested in the comparison between systems: is system A better or worse than system B? After running an experiment with a test collection, she has a

[3] We do not know whether system comparisons are really right or not because we do not know the true $\mu_{\Delta\lambda}$ scores. Conflict rates are approximations based on comparisons with two samples from the same distribution.

numeric answer to that question that measures the effectiveness difference between systems. Statistical methods are then used to check whether that difference is statistically significant or not. Statistical significance is usually thought of as a sort of bulletproof evidence that one system really is better than another, that the difference is somehow important. Researchers usually follow one or another research line based solely on statistical significance, and it has also become an essential requirement for publication in peer-reviewed venues.

However, there are several misconceptions regarding statistical significance [Ioannidis 2005, Ziliak and McCloskey 2008, Anderson et al. 2000]. In the case of IR evaluation experiments, null hypotheses about differences in performance are false almost by definition, so observing a small p-value to conclude significance is just a matter of meeting certain conditions in the experiment. On the other hand, very little attention is paid to effect-sizes and their implications in practical terms. In fact, even if statistical significance is present, the difference between two systems may very well be so subtle that users do not note the difference.

5.6.1 Understanding Evaluation Results

The effectiveness of IR systems is assessed with different system measures such as $GAP@k$ and $RR@k$. These measures are used to assign systems a score that represents how well they would satisfy users. For some system, an effectiveness measure defines a distribution of effectiveness scores Λ, describing the effectiveness of the system for an arbitrary query and user. The goal of evaluation experiments is usually finding the mean μ_λ of that distribution.

Computing the parameter μ_λ allows researchers to assess how well the system performs to get an idea of the expected user satisfaction according to the user model underlying the effectiveness measure. However, computing this distribution is not possible (see Section 4.1). IR evaluation experiments are run with a sample of queries $\mathcal{Q}$ and a sample of human assessors $\mathcal{H}$, so they are used as estimators of the true μ_λ. The average effectiveness across queries, $\overline{\lambda}_\mathcal{Q}$, is used as the estimate of μ_λ. Like any other estimate, $\hat{\mu}_\lambda$ bears some uncertainty, so statistical techniques such as confidence intervals should be employed to report the confidence on the estimation.

When comparing two systems, say A and B, one is usually interested in the distribution of the difference $\Delta\Lambda$, representing the *paired* difference in effectiveness between A and B for an arbitrary query and user. Again, a comparative IR evaluation experiment only provides an estimate $\hat{\mu}_{\Delta\lambda}$, whose sign indicates which system is expected to perform better.

Statistical Significance: p-values

Given that $\hat{\mu}_{\Delta\lambda}$ is an estimate, the immediate question is: how confident can we be of this difference? The observed $\overline{\Delta\lambda}_\mathcal{Q}$ could be just a random and rare observation due to the particular sample of queries and assessors used. Again, statistical techniques are needed to compute some sort of confidence on the difference. The most popular is hypothesis testing.

As mentioned above, we set our null hypothesis to $H_0 : \mu_{\Delta\lambda} = 0$ and the alternative to $H_1 : \mu_{\Delta\lambda} \neq 0$. With probability α researchers may conclude H_0 is not true when it actually is (a Type I error), and with probability β they may conclude H_0 is true when it is not (a Type II error). The result of a statistical significance test is a p-value. This is usually

mistaken with the probability of H_0 being true, but it is actually the probability of observing the difference $\overline{\Delta\lambda}_{\mathcal{Q}}$ (or one larger) under the assumption that H_0 is true [Cohen 1994]. That is, p-values are the probability of the data given the hypothesis, not the probability of the hypothesis given the data. If the reported p-value is smaller than the significance level α, we then reject the null hypothesis in favor of the alternative, and say that the difference is *statistically significant*.

But it is important to note that the test does *not* tell anything about H_0 being true or false, it only estimates the probability of observing the data if we assume it is true. The dichotomous significant versus not-significant interpretation is made by *us* based on the p-value and α, not by the test. This is often the ultimate goal of an IR evaluation: reaching significance. However, observing a statistically significant difference between two systems is usually misinterpreted as having high confidence that one system is much better than the other one because H_0 was rejected. In fact, all these null hypotheses are false almost by definition: any two different systems produce a distribution of differences with $\mu_{\Delta\lambda} \neq 0$ because they are different to begin with [Johnson 1999]. What is important is the magnitude of $\mu_{\Delta\lambda}$; differences of 0.0001, for instance, are probably not relevant, but differences of 0.8 definitely are. However, a difference of just 0.0001 will *always* be statistically significant under certain experimental conditions, so focusing on statistical significance alone becomes, at some point, meaningless [Gelman and Stern 2006].

Practical Significance: effect-sizes

The most popular procedure to test such hypotheses about population means is the paired t-test. The test statistic t is computed as in (5.1), from which we can compute the p-value. If $p \leq \alpha$, *we* (not the test) reject the null hypothesis and plainly conclude $\mu_{\Delta\lambda} > 0$; system A is superior (or system B if $\overline{\Delta\lambda}_{\mathcal{Q}} < 0$).

Examining (5.1) we can see three ways to increase t and therefore make the difference more likely to come up statistically significant. The first way is to actually further improve system A so that the observed difference $\overline{\Delta\lambda}_{\mathcal{Q}}$ is larger. The second way is to reduce variance so that $sd_{\mathcal{Q}}$ is smaller; bluntly put, make system A better than B for as many queries as possible. The third and most troublesome way is simply to use more queries. Equation (5.1) shows that the power of the test is directly proportional to the sample size $n_{\mathcal{Q}}$: the more queries we use to evaluate systems, the more likely to observe a significant difference. This shows that focusing on significance alone is eventually meaningless: all a researcher needs to do in order to obtain significance is simply to evaluate with more queries.

Increasing the sample size (number of queries) increases the power of the test to detect ever smaller differences because the standard error on the mean, $sd_{\mathcal{Q}}/\sqrt{n_{\mathcal{Q}}}$, decreases (the blue distribution in Figure 4.3 gets narrower). Thus, observing a statistically significant difference does not mean that the systems are very different, in fact *they always are*. It just means that the observed difference and the sample size used were large enough to conclude *with confidence* that the true difference is larger than zero.

But the test only provides evidence that there is *a* difference, it does *not* say how large that difference is. As it turns out, this is what matters as we saw in Chapter 3. This is the effect-size, which measures the *practical* significance of the difference. As shown in Section 3.3, large $\Delta\lambda$ scores (large effect-sizes) do translate into more user satisfaction, but

small differences do not. However, with a sufficiently large number of queries we may be able to detect a statistically significant difference whose effect-size is extremely small, having no value for real users. In such a case we would have statistical significance, but no practical significance at all. Of course this does not mean that researchers should disregard tiny incremental improvements in effectiveness; in the end all those improvements should add up. It means that we should not pay attention solely to statistical significance because it does not tell us, as usually believed, how important those improvements are.

5.6.2 Reporting and Interpreting Results

We showed above that obtaining small p-values (statistical significance) should not be the sole focus of researchers when running evaluation experiments. The focus should really be on obtaining large effect-sizes (practical significance). The easiest way to report effect-sizes is just to report the observed effectiveness difference $\overline{\Delta\lambda}_{\mathcal{Q}}$ between systems or the absolute score $\overline{\lambda}_{\mathcal{Q}}$ of a single system. But these figures are just estimates of population means, and therefore subject to error. A better way to report effect-sizes is with confidence intervals, computed as in (4.6). For instance, the CL1 system in MIREX AMS 2009 obtained an average $CG_l@5$ score of 0.2525 as per the Fine judgments. A usual way to report this result in the literature is:

$$CG_l@5 = 0.2525$$

omitting the readily available information on experimental error. A more appropriate report would include a 95% confidence interval:

$$CG_l@5 = 0.2525 \pm 0.0507$$

Along with the results in Chapter 3 and Chapter 4, this report can easily be extended to include information regarding the probability of user satisfaction and of system success:

$$\begin{aligned} CG_l@5 &= 0.2525 \pm 0.0507 \\ P(Sat) &= 0.3526 \pm 0.0434 \\ P(Succ) &= 0.24 \end{aligned}$$

In addition, using the Empirical distribution of $\hat{P}(Sat)$ scores, we may report that 95% of the future observations are expected to yield a level of satisfaction $P(Sat) > 0.19$.

On the other hand, the difference between systems BSWH2 and BSWH1 was found to be statistically significant, with a difference $\Delta CG_l@5 = 0.0597$. A typical report in the literature would indicate statistical significance as follows:

$$\Delta CG_l@5 = 0.0597^* \quad \text{or} \quad \Delta CG_l@5 = 0.0597 \ (p < 0.05)$$

The results in Chapter 3 and Chapter 4 can easily be included as well:

$$\begin{aligned} \Delta CG_l@5 &= 0.0597 \pm 0.0294^* \\ \Delta P(Sat) &= 0.0543 \pm 0.0298^* \\ \Delta P(Succ) &= 0.07 \end{aligned}$$

Such a report only tells us that the p-value is smaller than 0.05, but it does not tell us how large it is. Therefore, we do not know how likely it is for our conclusion to be wrong, we just know that the probability is less than 0.05. To illustrate why this is crucial, let us consider system LR as another alternative that improves BSWH1 as well, with a difference $\Delta CG_l@5 = 0.0332 \pm 0.0363$ but *not* statistically significant. Because BSWH2 improves the baseline with statistical significance an LR does not, in principle we would choose BSWH2. Now consider a full report of both alternative systems, including the actual p-values and some fictional cost. For BSWH2 the p-value is 0.00011, and the investment needed to implement it is \$1,000,000. For LR the p-value is 0.0721, and the required investment is \$100. Even though the first system improves the baseline more than the second one, the possibility of being wrong and committing a Type I error is just too risky.

The above example may be considered too extreme, but it illustrates the need to report actual p-values. The important matter is not only the potential risk of committing a Type I error though. In the example all systems were evaluated with the same sample of $n_{\mathcal{Q}} = 100$ queries, but if BSWH2 were evaluated with a larger sample of, say, 800 queries, then the p-value would be that small probably just because of the extremely large query sample. In addition, the distinction between $p = 0.051$ and $p = 0.049$ is clearly unreasonable, but without full reports the former work could be rejected based solely on significance. In fact, it is striking how authors argue the importance of their work when p-values are slightly above 0.05, but take it for granted when they are slightly below.

In summary, I suggest to report not only the observed scores but also their confidence intervals, and the actual p-values rather than binary indicators of significance. For instance, a proper report for a single system would read as $CG_l@5 = 0.5842 \pm 0.023$. For the difference between two systems, I suggest $\Delta CG_l@k = 0.0371 \pm 0.0314$ ($p = 0.024$). By reporting the p-value we leave the interpretation of significance to the reader and his operational context: a large effect-size (e.g. $\Delta\lambda = 0.43$), even if not statistically significant (e.g. $p = 0.06$), is definitely worth implementing. After all, the levels $\alpha = 0.05$ and $\alpha = 0.01$, despite widely accepted, are completely arbitrary[4]. People generally consider $p = 0.054$ as significant, while others might request $p < 0.001$. It depends on the context of the reader and factors such as the cost of committing a Type I error or the cost of implementing one or another technique. In any case, attention should be paid to effect-sizes and how they relate to user satisfaction, not only to p-values.

5.7 Summary

Using a test collection to estimate the average system performance of a system, or the difference between two systems, is subject to random error due to sampling. To account for this error, researchers usually employ a statistical significance test to assess the reliability of the conclusions drawn from the evaluation experiment with the test collection. There are several tests that can be used for this purpose, but each of them has a different set of assumptions. Unfortunately, these assumptions are generally violated in IR evaluation

[4] The general use of $\alpha = 0.05$ can be traced back to Fisher [1925], who personally considered this threshold convenient because it nicely corresponds to the probability of observing values beyond 2 standard deviations beyond the mean of a standard normal distribution.

experiments, so it is necessary to analyze them and figure out which one is optimal and under what conditions.

The results show that a researcher who wants to maximize the number of significant results may use the more powerful bootstrap test and still be safe in the usual scenario. Researchers that want to maximize safety may use the Wilcoxon test and the t-test, and researchers that want to be able to trust the significance level may generally proceed with the Wilcoxon test (though for large α levels the bootstrap test is the best choice). For large meta-analysis studies I encourage the use of the t-test and Wilcoxon test because they are far less computationally expensive and show near-optimal behavior. Unlike previous work concluded, our results suggest that in practice the permutation test is not optimal under any criterion. The argument of discontinuing the sign test is further supported by these results.

Regarding effectiveness measures, CG_l@5 was consistently the best one in terms of power and success rates, followed by GAP@5, DCG_l@5 and RBP_l@5. In terms of conflicts, the best measures were Q_l@5, DCG_l@5 and RBP_l@5. However, all measures yield conflict rates lower than expected, so the focus should be on power. Regarding relevance scales, the Fine scale was clearly superior to the Broad and $\ell_{min} = 40$ scales.

Reaching statistical significance in IR evaluation experiments is usually the most important goal for researchers. A difference between systems is usually regarded as important if statistical significance is involved, when in reality it just means that we can be confident that there is *a* difference, and we already know that. With the development of ever larger test collections, statistical significance can easily be misunderstood, suggesting large differences between systems when they are actually very similar. To predict the real-world implications of these differences, researchers need to focus on effect-sizes as indicators of practical significance. That is, it does not matter whether there is a difference or not (in fact, there always is), what matters is how large it is. Final user satisfaction, as seen in Chapter 3, is only predicted with effect-sizes; statistical significance serves just as a measure of confidence.

Chapter 6
Test Collection Size

The reliability of a test collection is proportional to its size. Test collections with large query sets and redundant relevance judgments made by different assessors provide better estimates of the distributions of system effectiveness. But building a collection with many queries and assessors is expensive, so researchers have to find a balance between reliability and cost. In this chapter Generalizability Theory is employed to analyze the optimal test collection characteristics. In particular, I analyze the effect of the number of queries, of the number of assessors, and of the evaluation cutoff. From the perspectives of obtaining reliable differences between pairs of systems and obtaining reliable estimates of absolute scores, several effectiveness measures and relevance scales are analyzed as well. The results presented in this chapter can be used by researchers as a guide in the creation of new test collections, or the expansion of old ones, to ensure they are reliable for their purposes.

6.1 Generalizability Theory

Generalizability Theory (GT) is a statistical framework for addressing issues related to the reliability of measurements [Brennan 2001, Shavelson and Webb 1991]. It originated in the Social Sciences as a means to determine the reliability of opinion surveys, student tests, etc. Consider a test with several questions, a set of students to take it, and a set of professors to grade them. Under GT, the average score of a student over the set of questions is an estimate of the true average score of the student for the universe of all admissible questions. GT may be used to measure how much variability comes from different noise sources, so that we can determine how well our observed scores generalize to the true scores.

Using GT we can measure how much our estimates depend on the particular set of questions (i.e. differences in difficulty) or on the particular set of professors (i.e. differences in permissibility). If many of the questions are too easy and all students give correct answers, then those questions are not useful to determine which students perform better. Similarly, if there are large differences among the scores given by different professors, we have an indication that our measurements are too noisy and that more professors are needed to obtain more reliable scores. GT can be used to identify the questions that are too easy or too hard, or the professors that are too permissive or too restrictive. But most importantly,

it can be used to identify where our resources should be put on (e.g. more questions or more professors) or different experimental designs that will work better.

6.1.1 GT to Measure Test Collection Reliability

Bodoff and Li [2007] proposed Generalizability Theory as a tool to measure test collection reliability that directly addresses variability of scores rather than just the mean as was common before (e.g. [Voorhees 1998, Zobel 1998, Buckley and Voorhees 2000, Voorhees and Buckley 2002, Sanderson and Zobel 2005, Sakai 2007, Voorhees 2009]). In our case, retrieval systems are the students, tested for different queries rather than test questions, and graded according to relevance assessors, measures and scales rather than professors. For our purposes, we consider an IR evaluation experiment as fitting the following fully-crossed $s \times q$ model (systems crossed with queries):

$$\lambda_{q,\mathsf{A}} = \lambda + \lambda_{\mathsf{A}} + \lambda_q + \varepsilon_{q\mathsf{A}} \tag{6.1}$$

where $\lambda_{q,\mathsf{A}}$ is the effectiveness score of system A for query q, λ is the grand average effectiveness of the population of systems for the universe of all queries, λ_{A} is the average effectiveness of system A for the universe of all queries (our goal), λ_q is the average effectiveness of the population of all systems for query q, and $\varepsilon_{q\mathsf{A}}$ is the residual modeling the particular deviation for system A and query q.

In the model in (6.1), the grand mean λ is a constant, and the other effects can be modeled as random variables with their own expectation and variance. As such, the variance of the observed scores is modeled as the sum of these variance components:

$$\sigma^2 = \sigma_s^2 + \sigma_q^2 + \sigma_{sq}^2 \tag{6.2}$$

where σ_s^2 is the variance due to actual differences among systems, σ_q^2 is the variance due to differences in difficulty among queries, and σ_{sq}^2 is the variance due to the system-query interaction effect whereby some systems are particularly good (or bad) for some queries. The variance due to other effects, such as assessors, is in this case confounded with the interaction effect.

GT has two stages: a Generalizability study (G-study) to estimate the variance components in (6.2) based on previous data, and a Decision study (D-study) that subsequently computes reliability indicators for a different experimental design.

6.1.2 G-Study

Using Analysis of Variance (ANOVA), the variance components in (6.2) can be estimated from previous data, usually an existing test collection[1]:

$$\begin{aligned} \hat{\sigma}_{sq}^2 = \hat{\sigma}_e^2 &= \mathrm{E}[MS_{residual}] \\ \hat{\sigma}_s^2 &= \frac{\mathrm{E}[MS_s] - \hat{\sigma}_e^2}{n_q} \\ \hat{\sigma}_q^2 &= \frac{\mathrm{E}[MS_q] - \hat{\sigma}_e^2}{n_s} \end{aligned} \tag{6.3}$$

[1] On counted occasions, the estimate for a variance component can be negative. In these situations it is substituted with zero [Brennan 2001].

where $\mathrm{E}[MS_\nu]$ is the expected Mean Square of component ν, and n_s and n_q are the number of systems and queries in the available previous data [Brennan 2001, Shavelson and Webb 1991]. These estimates can be used to compute the proportion of total variance that is due to each of the effects, such as how much of it is due to differences in query difficulty. Intuitively, if there are wide differences among queries it means our universe of admissible observations is too diverse, so we need many queries in our experiment to compute accurate estimates. In principle, we want the variance due to system differences to be as large as possible compared to the other facets. That would mean that systems obtain very different effectiveness scores, and it is therefore easy to distinguish the good ones from the bad ones.

6.1.3 D-Study

In the D-study, we can use the variance estimates from the G-study to compute the reliability of a different query set size n'_q. To this end, two reliability indicators are usually employed: the generalizability coefficient and the index of dependability.

Generalizability Coefficient ($\mathbf{E\rho^2}$) is the ratio of system variance to itself plus relative error variance:

$$\mathrm{E}\rho^2\left(n'_q\right) = \frac{\sigma_s^2}{\sigma_s^2 + \frac{\sigma_e^2}{n'_q}} \tag{6.4}$$

and it provides a measure of the stability of relative differences between systems $\overline{\Delta\lambda}$. By extension, it measures the reliability of the ranking of systems. For a collection to be reliable, $\mathrm{E}\rho^2$ must therefore tend to 1.

Index of Dependability ($\mathbf{\Phi}$) is the ratio of system variance to itself plus absolute error variance:

$$\Phi\left(n'_q\right) = \frac{\sigma_s^2}{\sigma_s^2 + \frac{\sigma_q^2 + \sigma_e^2}{n'_q}} \tag{6.5}$$

and it provides a measure of the stability of absolute effectiveness scores $\overline{\lambda}$. For a collection to be reliable, Φ must therefore tend to 1 as well.

The main advantage of these indicators is that they allow us to estimate the reliability of an arbitrary query set size n'_q without following the traditional methodologies based on random *what if* scenarios and extrapolation [Bodoff 2008, Urbano et al. 2013b]. From equations (6.4) and (6.5) it can be seen that the reliability of the collection increases as the number of queries increases, because estimates of query difficulty and system-query interactions are more precise. Additionally, we may see that query difficulty variance σ_q^2 does not affect relative stability because it does not matter how well or bad systems do for queries (i.e. $\overline{\lambda}_q$), just how well or bad they do with respect to the other systems.

With simple algebraic manipulation, we can calculate the minimum number of queries needed to reach some level of relative or absolute stability π:

$$n'_{\mathrm{E}\rho^2}(\pi) = \left\lceil \frac{\pi \cdot \sigma_e^2}{\sigma_s^2\,(1-\pi)} \right\rceil \tag{6.6}$$

$$n'_{\Phi}(\pi) = \left\lceil \frac{\pi\left(\sigma_q^2 + \sigma_e^2\right)}{\sigma_s^2\,(1-\pi)} \right\rceil \tag{6.7}$$

Measure	Broad											
	2007			2009			2010			2011		
	$\hat{\sigma}_s^2$	$\hat{\sigma}_q^2$	$\hat{\sigma}_e^2$	$\hat{\sigma}_s^2$	$\hat{\sigma}_q^2$	$\hat{\sigma}_e^2$	$\hat{\sigma}_s^2$	$\hat{\sigma}_q^2$	$\hat{\sigma}_e^2$	$\hat{\sigma}_s^2$	$\hat{\sigma}_q^2$	$\hat{\sigma}_e^2$
CG_l@5	0.162	0.4	**0.438**	0.35	0.291	**0.359**	0.266	0.448	**0.286**	0.179	0.478	**0.343**
DCG_l@5	0.155	0.405	0.44	0.348	0.28	0.372	0.259	0.44	0.301	0.18	0.465	0.355
$nDCG_l$@5	0.17	0.347	0.483	0.374	**0.23**	0.397	0.285	0.366	0.349	0.198	0.407	0.395
Q_l@5	0.157	0.351	0.492	0.374	0.236	0.39	0.28	0.382	0.338	**0.206**	**0.403**	0.391
RBP_l@5	**0.173**	**0.346**	0.481	**0.375**	0.232	0.394	**0.29**	**0.363**	0.347	0.197	0.41	0.393
GAP@5	0.151	0.363	0.486	0.363	0.236	0.401	0.268	0.394	0.338	0.204	0.409	0.387

Measure	Fine											
	2007			2009			2010			2011		
	$\hat{\sigma}_s^2$	$\hat{\sigma}_q^2$	$\hat{\sigma}_e^2$	$\hat{\sigma}_s^2$	$\hat{\sigma}_q^2$	$\hat{\sigma}_e^2$	$\hat{\sigma}_s^2$	$\hat{\sigma}_q^2$	$\hat{\sigma}_e^2$	$\hat{\sigma}_s^2$	$\hat{\sigma}_q^2$	$\hat{\sigma}_e^2$
CG_l@5	0.173	0.404	0.423	0.369	0.292	**0.339**	0.262	0.502	**0.236**	0.184	0.511	**0.304**
DCG_l@5	0.17	0.412	**0.418**	0.36	0.286	0.354	0.259	0.489	0.252	0.186	0.501	0.314
$nDCG_l$@5	0.206	0.3	0.494	0.422	0.182	0.396	0.353	0.297	0.349	0.219	0.412	0.369
Q_l@5	0.19	0.318	0.492	0.398	0.187	0.415	0.334	0.32	0.346	0.214	**0.405**	0.381
RBP_l@5	**0.208**	**0.296**	0.495	**0.425**	**0.182**	0.393	**0.36**	**0.296**	0.344	0.217	0.413	0.37
GAP@5	0.188	0.328	0.483	0.416	0.202	0.383	0.342	0.33	0.328	**0.226**	0.419	0.355

Measure	$\ell_{min} = 40$											
	2007			2009			2010			2011		
	$\hat{\sigma}_s^2$	$\hat{\sigma}_q^2$	$\hat{\sigma}_e^2$	$\hat{\sigma}_s^2$	$\hat{\sigma}_q^2$	$\hat{\sigma}_e^2$	$\hat{\sigma}_s^2$	$\hat{\sigma}_q^2$	$\hat{\sigma}_e^2$	$\hat{\sigma}_s^2$	$\hat{\sigma}_q^2$	$\hat{\sigma}_e^2$
P@5	**0.158**	**0.375**	0.467	**0.34**	0.271	**0.388**	**0.238**	0.466	**0.296**	0.167	0.464	**0.369**
AP@5	0.137	0.398	**0.465**	0.313	**0.27**	0.417	0.208	**0.454**	0.338	**0.174**	**0.452**	0.375

Table 6.1: Estimated variance components (over total variance) for all measures and scales of interest and for the MIREX 2007, 2009, 2010 and 2011 AMS test collections. Best measure per component, year and scale in bold.

which can be used to estimate how many more queries we need to add to our collection for it to reach some degree of reliability. The main use of this approach can be found in the TREC Million Query Track [Allan et al. 2007, 2008], which set out to study whether many queries with a few judgments yield more reliable results than a few queries with many judgments. The conclusion was that $n_{\mathcal{Q}} \approx 80$ queries are sufficient for a reliable ranking, while $n_{\mathcal{Q}} \approx 130$ are needed for reliable absolute scores. However, Urbano et al. [2013b] later showed that the recommended number of queries varies too much across tasks, and even across different collections within the same task. As a result, the optimum collection characteristics need to be analyzed on a case by case basis, with the actual systems to evaluate.

6.2 The Effect of Query Set Size

I analyzed the reliability of the MIREX 2007, 2009, 2010 and 2011 AMS test collections by running a G-study to compute variance components and then a D-study to analyze the effect of the query set size. All these four collections had a query set of size $n_{\mathcal{Q}} = 100$, and all queries were randomly sampled from the document collection itself. A G-study and the corresponding D-study are run separately for each MIREX edition and each combination of effectiveness measure and relevance scale as studied in Chapter 5.

Measure	Broad							
	2007		2009		2010		2011	
	$\hat{\mathrm{E}\rho}^2$	$\hat{n}'_{\mathrm{E}\rho^2}(.95)$	$\hat{\mathrm{E}\rho}^2$	$\hat{n}'_{\mathrm{E}\rho^2}(.95)$	$\hat{\mathrm{E}\rho}^2$	$\hat{n}'_{\mathrm{E}\rho^2}(.95)$	$\hat{\mathrm{E}\rho}^2$	$\hat{n}'_{\mathrm{E}\rho^2}(.95)$
CG_l@5	**0.9738**	**52**	**0.9898**	**20**	**0.9893**	**21**	0.9812	37
DCG_l@5	0.9725	54	0.9894	21	0.9885	23	0.9807	38
$nDCG_l$@5	0.9724	54	0.9895	21	0.9879	24	0.9804	38
Q_l@5	0.9696	60	0.9897	20	0.9881	23	0.9814	37
RBP_l@5	0.9729	53	0.9896	20	0.9882	23	0.9804	38
GAP@5	0.9687	62	0.9891	21	0.9875	24	**0.9814**	**36**

Measure	Fine							
	2007		2009		2010		2011	
	$\hat{\mathrm{E}\rho}^2$	$\hat{n}'_{\mathrm{E}\rho^2}(.95)$	$\hat{\mathrm{E}\rho}^2$	$\hat{n}'_{\mathrm{E}\rho^2}(.95)$	$\hat{\mathrm{E}\rho}^2$	$\hat{n}'_{\mathrm{E}\rho^2}(.95)$	$\hat{\mathrm{E}\rho}^2$	$\hat{n}'_{\mathrm{E}\rho^2}(.95)$
CG_l@5	0.9761	47	**0.9909**	**18**	**0.9911**	**18**	0.9838	32
DCG_l@5	0.976	47	0.9903	19	0.9904	19	0.9834	33
$nDCG_l$@5	0.9765	46	0.9907	18	0.9902	19	0.9834	33
Q_l@5	0.9748	50	0.9897	20	0.9898	20	0.9825	34
RBP_l@5	**0.9768**	**46**	0.9908	18	0.9905	19	0.9833	33
GAP@5	0.975	49	0.9909	18	0.9905	19	**0.9845**	**30**

Measure	$\ell_{min} = 40$							
	2007		2009		2010		2011	
	$\hat{\mathrm{E}\rho}^2$	$\hat{n}'_{\mathrm{E}\rho^2}(.95)$	$\hat{\mathrm{E}\rho}^2$	$\hat{n}'_{\mathrm{E}\rho^2}(.95)$	$\hat{\mathrm{E}\rho}^2$	$\hat{n}'_{\mathrm{E}\rho^2}(.95)$	$\hat{\mathrm{E}\rho}^2$	$\hat{n}'_{\mathrm{E}\rho^2}(.95)$
P@5	**0.9713**	**57**	**0.9887**	**22**	**0.9877**	**24**	0.9784	42
AP@5	0.9673	65	0.9868	26	0.984	31	**0.9789**	**41**

Table 6.2: Estimated $\mathrm{E}\rho^2$ scores (higher is better) for all measures and scales of interest and for the MIREX 2007, 2009, 2010 and 2011 AMS test collections, along with required number of queries to reach $\hat{\mathrm{E}\rho}^2 = 0.95$ (lower is better). Best per year and scale in bold face.

6.2.1 G-Study

Table 6.1 shows the results of the G-study for all measures and scales of interest. In particular, the table lists the estimated fraction of total variance due to each of the effects as per equations (6.3); for instance, in the 2009 collection 29.2% of the variance in CG_l@5 scores is due to the query difficulty effect when using the Fine scale. It can be seen that differences among systems have the smallest effect on the effectiveness scores. On the other hand, the query and system-query interaction effects are larger, meaning that queries are very diverse and therefore there is much variability due to query difficulty and some systems being particularly good or bad for some queries (e.g. systems good for rock and roll queries, but not for jazz).

A clear difference can be seen among test collections: the 2009 and 2010 collections have a large proportion of variance due to the system effect, while in 2007 and 2011 it is about half as much. This is not a fault of the collection per se, because the same methodology and data was indeed followed to build them. The difference comes from the particular systems that participated those years. During the first and second editions in 2006 (see Section 6.4) and 2007 the variance due to systems was quite small, probably because the actual retrieval techniques employed were all very similar across teams. The task did not run

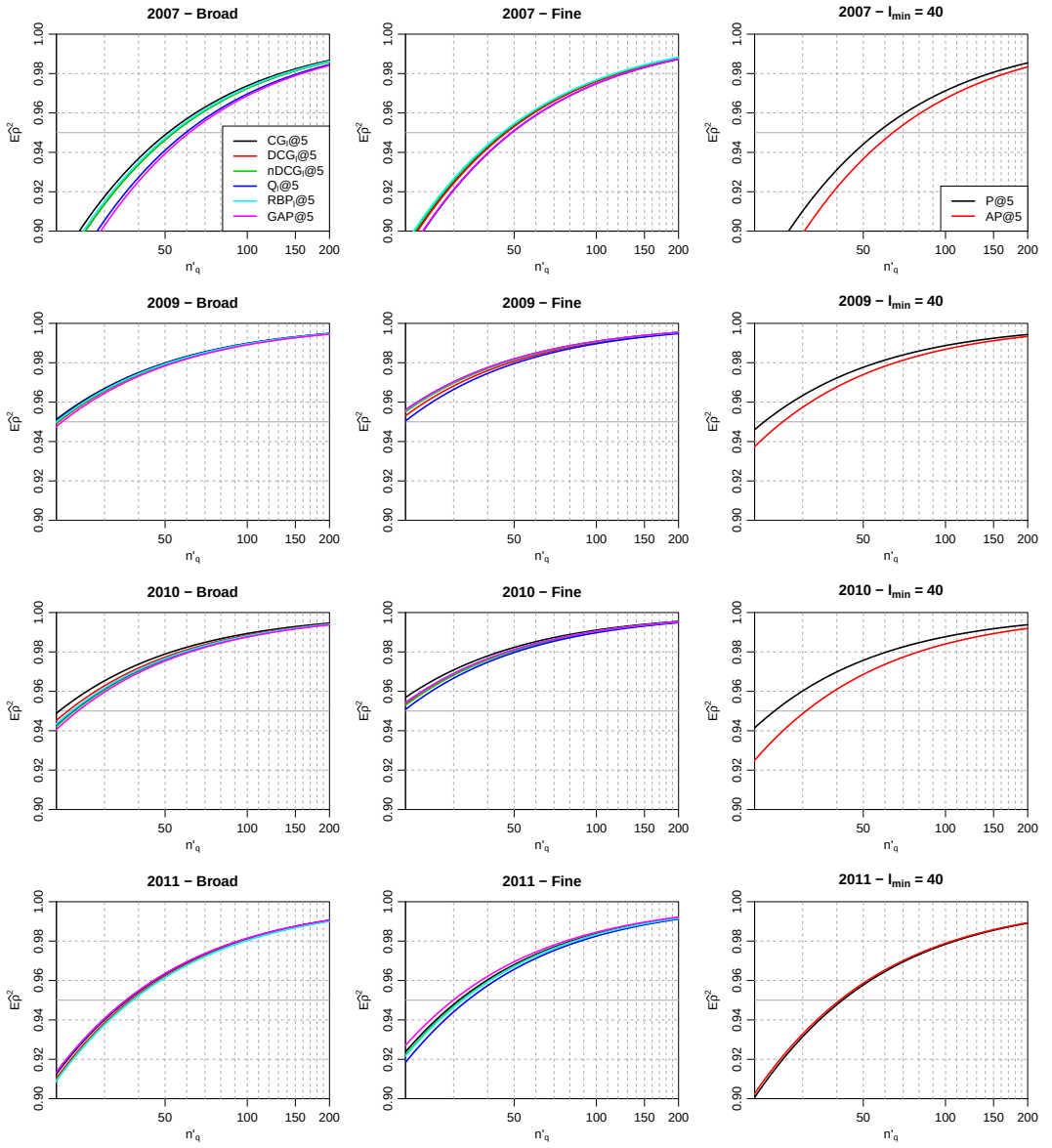

Figure 6.1: Estimated Eρ^2 scores as a function of query set size n'_q, for all measures and scales of interest and for the MIREX 2007, 2009, 2010 and 2011 AMS test collections (higher is better).

in 2008 because there "was a general consensus that developers needed more time to make non-trivial improvements to their systems" [Downie et al. 2010], evidencing that retrieval techniques were in fact similar. The task then ran again in 2009 and 2010, when a larger system effect was observed probably due to systems being this time more different from each other because some teams were more successful than others in improving retrieval techniques. With time, differences among systems were reduced due to systems catching up with each other, as evidenced by the low variance due to systems in 2011 and 2012 (see Section 6.3).

Measure	Broad							
	2007		2009		2010		2011	
	$\hat{\Phi}$	$\hat{n}'_\Phi(.95)$	$\hat{\Phi}$	$\hat{n}'_\Phi(.95)$	$\hat{\Phi}$	$\hat{n}'_\Phi(.95)$	$\hat{\Phi}$	$\hat{n}'_\Phi(.95)$
CG_l@5	0.951	98	0.9818	36	0.9731	53	0.9561	88
DCG_l@5	0.9484	104	0.9816	36	0.9722	55	0.9565	87
$nDCG_l$@5	0.9535	93	0.9835	32	0.9755	48	0.961	78
Q_l@5	0.9491	102	0.9836	32	0.9749	49	**0.9629**	**74**
RBP_l@5	**0.9544**	**91**	**0.9836**	**32**	**0.9761**	**47**	0.9608	78
GAP@5	0.9466	108	0.9828	34	0.9734	52	0.9625	75

Measure	Fine							
	2007		2009		2010		2011	
	$\hat{\Phi}$	$\hat{n}'_\Phi(.95)$	$\hat{\Phi}$	$\hat{n}'_\Phi(.95)$	$\hat{\Phi}$	$\hat{n}'_\Phi(.95)$	$\hat{\Phi}$	$\hat{n}'_\Phi(.95)$
CG_l@5	0.9543	91	0.9832	33	0.9726	54	0.9576	85
DCG_l@5	0.9535	93	0.9825	34	0.9722	55	0.958	84
$nDCG_l$@5	0.9628	74	**0.9865**	**27**	0.982	35	0.9655	68
Q_l@5	0.9592	81	0.9851	29	0.9805	38	0.9646	70
RBP_l@5	**0.9634**	**73**	0.9866	26	**0.9825**	**34**	0.9653	69
GAP@5	0.9586	82	0.9861	27	0.9812	37	**0.9669**	**66**

Measure	$\ell_{min} = 40$							
	2007		2009		2010		2011	
	$\hat{\Phi}$	$\hat{n}'_\Phi(.95)$	$\hat{\Phi}$	$\hat{n}'_\Phi(.95)$	$\hat{\Phi}$	$\hat{n}'_\Phi(.95)$	$\hat{\Phi}$	$\hat{n}'_\Phi(.95)$
P@5	**0.9494**	**102**	**0.981**	**37**	**0.969**	**61**	0.9526	95
AP@5	0.941	120	0.9785	42	0.9634	73	**0.9546**	**91**

Table 6.3: Estimated Φ scores (higher is better) for all measures and scales of interest and for the MIREX 2007, 2009, 2010 and 2011 AMS test collections, along with required number of queries to reach $\hat{\Phi} = 0.95$ (lower is better). Best per year and scale in bold face.

In terms of relevance scales, it is clear that the Fine scale results in larger variance due to systems than the Broad and $\ell_{min} = 40$ scales, so it is expected to be more stable. In terms of effectiveness measures, CG_l@5 and DCG_l@5 generally have the lowest σ_e^2 system-query interaction effect, so they are expected to be the most stable in terms of relative scores. However, in terms of σ_q^2 they are clearly the worst measures, so they should perform significantly worse than the others when considering absolute score stability. Both RBP_l@5 and $nDCG_l$@5 yield the smaller query difficulty effect, so they are expected to be the most stable in this case. In the binary relevance scale, P@5 is clearly superior to AP@5.

6.2.2 D-Study

Using the variance components estimated in the G-study above, a D-study was run to find out the optimal query set size. For each collection, effectiveness measure and relevance scale, the $E\rho^2$ and Φ scores are estimated as per equations (6.4) and (6.5).

Table 6.2 shows the $E\hat{\rho}^2$ estimates (relative stability). In all cases we see very high scores, with an average $E\hat{\rho}^2 = 0.98$. As anticipated in the G-study, the 2009 and 2010 collections are more stable and the Fine scale performs better than the Broad and $\ell_{min} = 40$ scales. In terms of measures, CG_l@5 is clearly the most stable measure within the Broad scale,

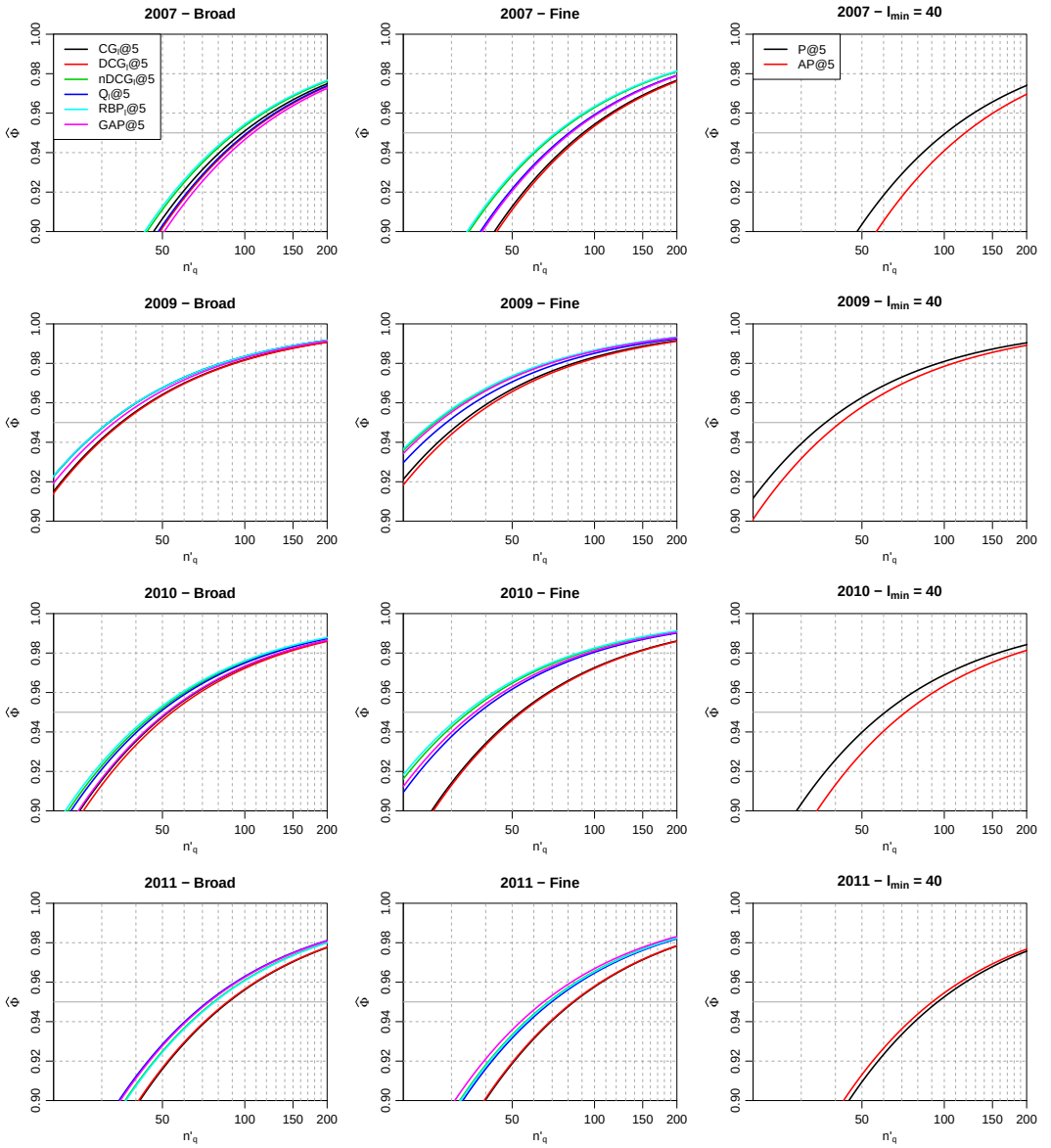

Figure 6.2: Estimated Φ scores as a function of query set size n'_q, for all measures and scales of interest and for the MIREX 2007, 2009, 2010 and 2011 AMS test collections (higher is better).

followed by RBP_l@5. Within the Fine scale we may draw the same conclusion, and within the binary scale P@5 is clearly the measure of choice. Regarding the required number of queries to reach $\mathrm{E}\hat{\rho}^2 = 0.95$ as to (6.6), we see that in all cases the test collections ($n_{\mathcal{Q}} = 100$) are much larger than necessary. On average, 35 queries seem enough, requiring about 60 queries in the worst cases.

Figure 6.1 shows how $\mathrm{E}\hat{\rho}^2$ increases with the size of the query set; note that the x axis is log-scaled. It can be seen that in all cases the trend converges quite early, with the arguable exception of the 2007 collection. In fact, the average improvement in $\mathrm{E}\hat{\rho}^2$ is about $+0.025$

when going from 50 to 100 queries. This is an improvement rather small considering that the judging effort is doubled, especially given that all scores are already larger than $\mathrm{E}\rho^2 = 0.95$.

Table 6.3 shows the $\hat{\Phi}$ estimates (absolute stability). We can observe again very high stability scores, and in virtually all cases the $\hat{\Phi} = 0.95$ threshold is passed. This time it can be seen that RBP_l@5 and $nDCG_l$@5 performed better, as anticipated in the G-study. The difference between the Broad and Fine scales is shortened, though the Fine scale still performs better. Within the binary $\ell_{min} = 40$ scale, P@5 does better than AP@5 in general. Regarding the required number of queries to reach $\hat{\Phi} = 0.95$ as per (6.7), we see that collections need to be larger if our goal is absolute score stability than we needed for relative stability: 64 queries are required on average, with about 100 in the worst cases.

Figure 6.2 shows how $\hat{\Phi}$ increases as the query set size increases; note again that the x axis is log-scaled. Compared to Figure 6.1 it can be seen that convergence takes longer because in this case we do account for variance due to queries. Nevertheless, quite high stability is achieved with rather small query sets. However, the improvement of using 50 queries instead of 100 is about +0.045, that is, nearly twice as large than for $\mathrm{E}\hat{\rho}^2$. Differences among measures are clearer here, especially within the Fine scale: CG_l@5 and DCG_l@5 are the worst measures for stable absolute scores. In terms of scales, differences are very small, although the Fine scale is again slightly superior to the Broad and $\ell_{min} = 40$ scales.

6.3 The Effect of Evaluation Cutoff

Given the reliability results for the 2007, 2009, 2010 and 2011 collections, I proposed to reduce the query set size in MIREX 2012 from $n_Q = 100$ to $n_Q = 50$ and increase the evaluation cutoff from $k = 5$ to $k = 10$, that is, judge the top 10 documents retrieved rather than just the top 5 as usual. It was expected for $\overline{\Delta\lambda}$ scores to be stable with half the queries, and for the rank-aware measures to become more discriminative than the traditional CG_l@5 when considering the top 10 documents retrieved instead of just 5. Although the cutoff is twice as much, the query set size is half the usual, so the total number of relevance judgments was not expected to increase.

A total of 10 systems were submitted by 7 teams, and 2,622 relevance judgments were needed to evaluate all systems for $k = 10$. It was therefore necessary to judge 262 documents per system, while in the previous editions 363 documents were needed on average (see Table 1.1). In terms of judging effort, the cost was thus reduced to about 72%. In terms of test collection reliability, Figure 6.3-top shows how $\mathrm{E}\hat{\rho}^2$ is affected by the evaluation cutoff k. For all measures and scales the relative stability of effectiveness scores is improved between +0.01 and +0.03. In general, improvements are slightly larger within the Broad and $\ell_{min} = 40$ scales than within the Fine scale, though the latter still outperforms the others in absolute terms. In terms of reliability, the stability of relative scores was thus improved as well. The bottom plots show how $\hat{\Phi}$ is affected by the evaluation cutoff. Relative improvements are generally smaller than with $\mathrm{E}\hat{\rho}^2$, and in some cases the absolute stability of scores is ever so slightly reduced (RBP_l@5 with Fine judgments, P@5 and AP@5). Doubling the evaluation cutoff k thus resulted in larger stability of effectiveness scores, but we must consider the judging effort it requires. Figure 6.3 shows that $k = 10$ outperforms $k = 5$ for the same number of queries, but using $k = 10$ requires more judgments to begin

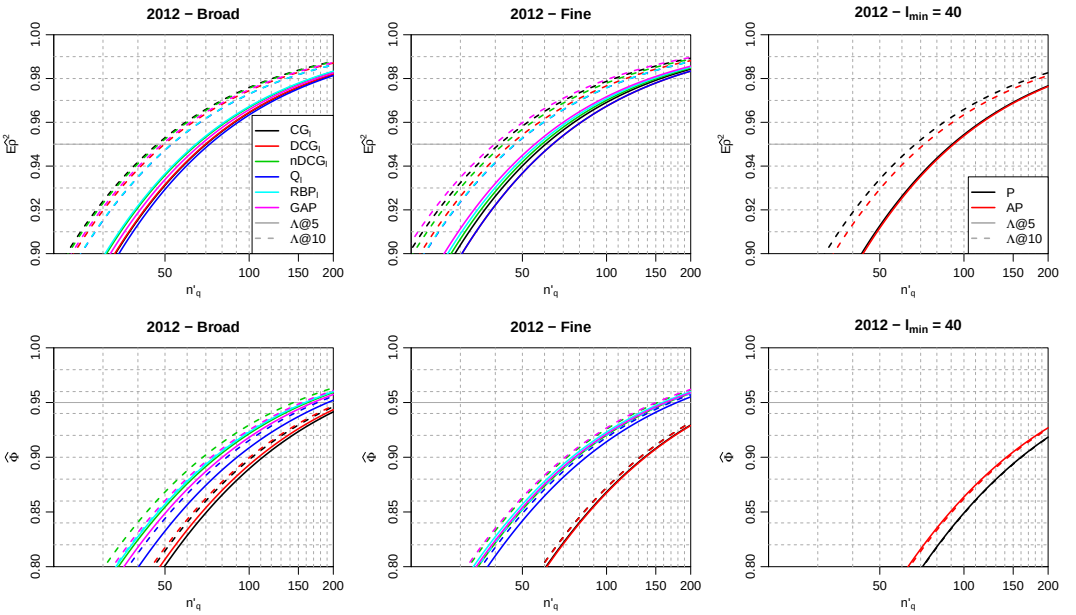

Figure 6.3: Estimated $E\rho^2$ (top) and Φ (bottom) scores in MIREX 2012 as a function of the query set size n'_q and evaluation cutoff k, for all measures and scales of interest (higher is better).

with. This means that for some fixed budged we can use fewer queries with $k = 10$, so the stability of scores will probably be smaller.

Assuming the MIREX 2012 queries and systems but with $k = 5$, it would have been necessary to judge 1,542 documents, that is, 31 documents per query. In the actual $k = 10$ scenario 52 documents were judged per query (+70%). Figure 6.4 plots the stability scores as a function of the number of relevance judgments, correcting the number of queries accordingly. It can be seen that using more queries is always more reliable than using a larger evaluation cutoff in terms of $E\hat{\rho}^2$. However, assuming an average of 363 judgments per system (vertical solid line), as historically observed in MIREX, the difference between $k = 5$ and $k = 10$ is just about 0.01, and between stability scores that are already larger than $E\hat{\rho}^2 = 0.95$. As the number of judgments increases, the difference gets smaller. In terms of absolute stability, the bottom plots also show that using more queries is always more reliable than using a deeper evaluation cutoff, but relative differences among measures show that choosing an evaluation cutoff can be tricky. For instance, in the Fine scale we can see that for the same judging effort CG_l@5 and DCG_l@5 are more reliable than their Λ@10 counterparts, but both GAP@10 and $nDCG_l$@10 are more reliable with the deeper $k = 10$ cutoff, and even RBP_l@10 and Q_l@10 perform very similarly.

6.4 The Effect of Assessor Set Size

In the first edition of the MIREX AMS task in 2006, three different assessors made judgments for every query-document pair. The test collection built for that occasion can therefore be used to measure the effect of assessor set size $n_{\mathcal{H}}$. Bodoff [2008] describes how to estimate

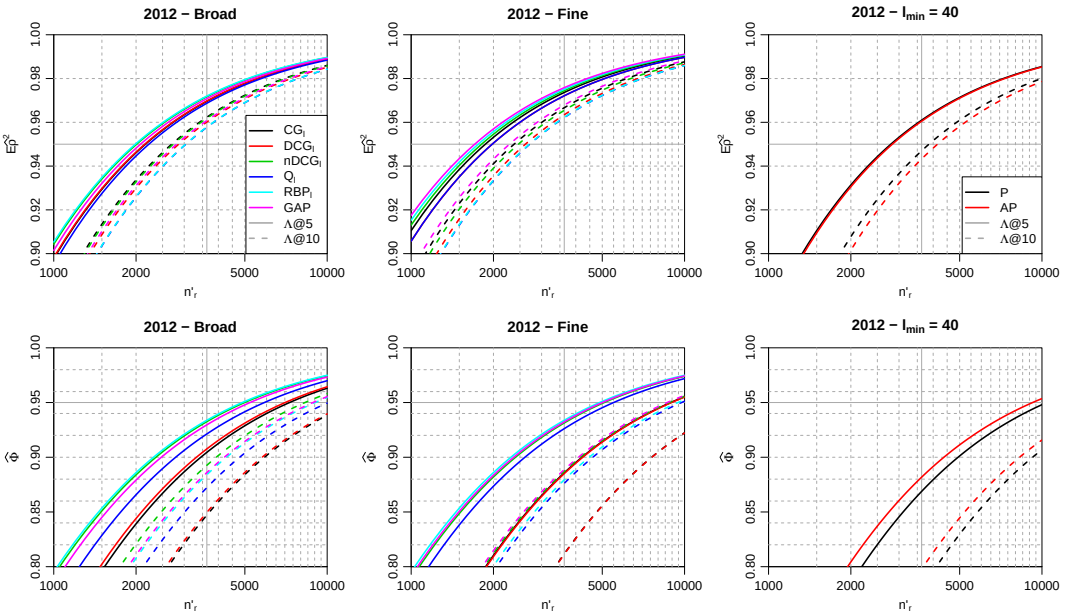

Figure 6.4: Estimated $E\rho^2$ (top) and Φ (bottom) scores in MIREX 2012 as a function of the number of relevance judgments n'_r and evaluation cutoff k, for all measures and scales of interest (higher is better). The vertical solid line marks the usual number of judgments that would have been made in a traditional MIREX setting.

variance components in a G-study with a fully crossed experimental design $s \times q \times h$ where all assessors provide judgments for all queries and all systems. However, this experimental design is different from that in MIREX 2006, because the set of three assessors was different across queries. That is, assessor h_i in query q_j is not necessarily the same as assessor h_i in query q_k. In our experimental design we have assessors nested within queries: $s \times h : q$. The variance components can still be estimated with an Analysis of Variance with a nested model as follows [Brennan 2001]:

$$\begin{aligned}
\hat{\sigma}^2_{sh:q} = \hat{\sigma}_e &= \mathrm{E}[MS_{residual}] \\
\hat{\sigma}^2_{sq} &= \frac{\mathrm{E}[MS_{sq}] - \hat{\sigma}^2_{sh:q}}{n_h} \\
\hat{\sigma}^2_{h:q} &= \frac{\mathrm{E}[MS_{h:q}] - \hat{\sigma}^2_{sh:q}}{n_s} \\
\hat{\sigma}^2_{q} &= \frac{\mathrm{E}[MS_{q}] - n_h\hat{\sigma}^2_{sq} - n_s\hat{\sigma}^2_{h:q} - \hat{\sigma}^2_{sh:q}}{n_s n_h} \\
\hat{\sigma}^2_{s} &= \frac{\mathrm{E}[MS_{s}] - n_h\hat{\sigma}_{sq} - \hat{\sigma}^2_{sh:q}}{n_h n_q}
\end{aligned} \tag{6.8}$$

The main difference with the crossed experimental design in Section 6.1 is that we are now able to estimate the variability due to assessors within queries by computing $\hat{\sigma}^2_{h:q}$. In the crossed model this variance was confounded with the residual variance, so our estimates for the other variance components are more accurate here too. Relative error variance includes again all factors crossed with the system main effect, and absolute error variance includes

Measure	Broad				
	$\hat{\sigma}_s^2$	$\hat{\sigma}_q^2$	$\hat{\sigma}_{sq}^2$	$\hat{\sigma}_{h:q}^2$	$\hat{\sigma}_e^2$
CG_l@5	0.028	0.374	**0.287**	0.029	**0.282**
DCG_l@5	0.028	0.35	0.289	0.027	0.306
$nDCG_l$@5	0.027	0.2	0.314	0.05	0.408
Q_l@5	**0.041**	0.266	0.296	**0.027**	0.37
RBP_l@5	0.029	0.195	0.322	0.045	0.409
GAP@5	0.028	**0.184**	0.305	0.049	0.434

Measure	Fine				
	$\hat{\sigma}_s^2$	$\hat{\sigma}_q^2$	$\hat{\sigma}_{sq}^2$	$\hat{\sigma}_{h:q}^2$	$\hat{\sigma}_e^2$
CG_l@5	0.032	0.404	0.261	0.016	**0.288**
DCG_l@5	0.031	0.383	**0.26**	0.014	0.312
$nDCG_l$@5	0.034	0.227	0.303	0.002	0.434
Q_l@5	**0.05**	0.234	0.306	0.002	0.408
RBP_l@5	0.036	**0.223**	0.312	**0.002**	0.428
GAP@5	0.034	0.238	0.285	0.021	0.422

Measure	$\ell_{min} = 40$				
	$\hat{\sigma}_s^2$	$\hat{\sigma}_q^2$	$\hat{\sigma}_{sq}^2$	$\hat{\sigma}_{h:q}^2$	$\hat{\sigma}_e^2$
P@5	**0.037**	0.37	**0.223**	**0.014**	**0.356**
AP@5	0.034	**0.319**	0.237	0.015	0.395

Table 6.4: Estimated variance components (over total variance) for all measures and scales of interest in the MIREX 2006 AMS test collection. Best measure per component and scale in bold.

all factors in the model. The generalizability coefficient and the index of dependability are therefore defined as:

$$\mathrm{E}\rho^2\left(n'_q, n'_h\right) = \frac{\sigma_s^2}{\sigma_s^2 + \frac{\sigma_{sq}^2}{n'_q} + \frac{\sigma_{sh:q}^2}{n'_q n'_h}} \tag{6.9}$$

$$\Phi\left(n'_q, n'_h\right) = \frac{\sigma_s^2}{\sigma_s^2 + \frac{\sigma_q^2 + \sigma_{sq}^2}{n'_q} + \frac{\sigma_{h:q}^2 + \sigma_{sh:q}^2}{n'_q n'_h}} \tag{6.10}$$

Table 6.4 shows the G-study results of the nested experimental design with the MIREX 2006 data. In the Broad scale we can see that the assessor within query effect $\sigma_{h:q}^2$ is comparable to the system effect σ_s^2, meaning that the differences observed among systems are as large as those observed among assessors [Schedl et al. 2013a]. Within the Fine scale the system effect is quite larger as desirable, meaning that effectiveness differences are less noisy due to assessor effects. This noise reduction is significantly larger with $nDCG$@5, Q_l@5 and RBP_l@5. Even within the binary $\ell_{min} = 40$ scale the system to assessor effect ratio is larger than in the Broad scale, so effectiveness scores are less sensitive to disagreements among assessors.

To illustrate the effect of assessor set size, D-studies were run for $n'_h \in \{1, 2, 3, 4, 5\}$ according to RBP_l@5 with Fine judgments (other measures and scales have very similar relative results). Figure 6.5-top shows how the D-study results are affected by the assessor set size. As the plots show, using more assessors does indeed improve effectiveness score

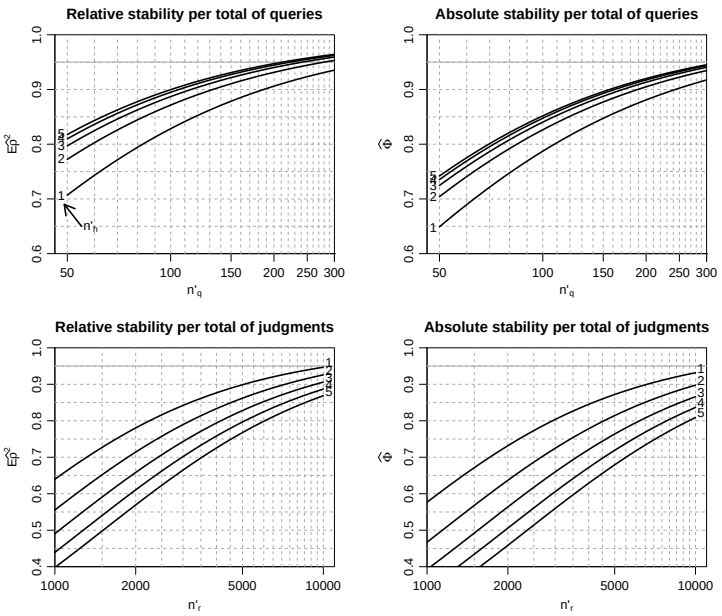

Figure 6.5: Estimated $E\rho^2$ (left) and Φ (right) scores in MIREX 2006 as a function of the number of assessors, and as a function of the number of queries n'_q (top) and the number of relevance judgments n'_r (bottom) for RBP_l@5 with Fine judgments (higher is better).

stability. For instance, at the usual 100 queries using two assessors instead of just one improves relative stability by about +0.05. However, improvements get smaller with larger query sets, and in any case using more than 2 assessors per query does not seem to be worth it. But we must consider the judging effort again. Having two assessors improves score stability over using just one, but the required number of judgments is twofold. For a fixed budget, this means that we can evaluate on half as many queries if using two assessors per query. Figure 6.5-bottom similarly shows the stability of effectiveness scores, but as a function of the number of relevance judgments rather than the number of queries. As expected, for the same judging effort it is always more reliable to use more queries than assessors, though accuracy should be slightly lower. For a large number of judgments though, improvements are again very small.

6.5 Discussion

Setting our stability goal to $E\rho^2 = 0.95$ and $\Phi = 0.95$, D-study results with MIREX 2007, 2009, 2010 and 2011 data evidence that fewer queries than the traditional 100 used in MIREX are actually necessary to obtain stable effectiveness scores. On average, as few as 35 queries are usually enough for $\overline{\Delta\lambda}$ scores to be stable, and about 65 are required for $\overline{\lambda}$ scores to be stable. This suggests that the traditional MIREX setting with 100 queries is actually misspending human resources on relevance judging. Using data from MIREX 2006 I showed that using more than one assessor per query results in more stability, though going beyond two merely does. However, when accounting for the extra amount of judging effort, it

does not pay off; it is better to spend resources on more queries with just one assessor. Using data from MIREX 2012 it was shown that spending judging effort on a deeper evaluation cutoff leads to comparable stability in scores, but it provides valuable information beyond the typical top $k = 5$ documents retrieved. Assuming the music recommendation scenario discussed in Chapter 3, I personally believe that looking at 10 documents rather than just 5 provides a more realistic setting for playlist consumption. Further considering that as of now AMS systems do not consider diversity of artists or genres, it definitely seems that only 5 documents are not representative of an actual use case scenario [Bollen et al. 2010].

For other fundamentally different scenarios, such as known item search or plagiarism detection, other test collection characteristics may be preferred. In any case, this chapter provides the tools for researchers to measure the reliability of their current test collections, and it allows them to design an optimal test collection based on previous data. An important benefit of using Generalizability Theory for this purpose is that it allows researchers to easily measure the reliability of a test collection *while* it is being built. Using the MIREX AMS task as an example, queries could be iteratively selected for judging and subsequently included in the query set. At each iteration, the stability of the current collection can be measured, and the required number of queries be estimated. If an even larger set is needed, a new query can be randomly selected and evaluated as well, repeating the process. If the estimated effort is too large for the current budget, researchers may spend resources somewhere else or change plans accordingly. The usefulness of GT in this scenario is clear in the case of MIREX. As mentioned, the number of queries can generally be reduced, but for some editions where systems were very similar (e.g. 2006 and 2007), many queries were needed to reliably differentiate the good systems from the bad ones. However, there is no way of knowing how similar systems are going to be *before* actually proceeding with the evaluation[2]. With GT though, we can estimate how large the collection needs to be *while* we build it. The question left is: how reliable do we want the test collection to be? The reader is referred to [Urbano et al. 2013b] for a detailed discussion on this topic.

Regarding relevance scales, results evidence again that the Fine scale is superior to the Broad scale, and that the binary $\ell_{min} = 40$ scale is significantly less reliable. Regarding measures, CG_l@5 and RBP_l@5 performed best when seeking stability of relative scores, followed by DCG_l@5 and $nDCG_l$@5. When seeking stability of absolute scores, RBP_l@5 and $nDCG_l$@5 performed best overall, followed by GAP@5 and Q_l@5. For absolute stability, CG_l@5 and DCG_l@5 were therefore the worst measures, probably due to the fact that they are the only two measures that do not account for the full set of relevance judgments. They just consider the relevance of the top k documents retrieved by the system and ignore all other judgments, so they are unaware of query difficulty.

6.6 Summary

In order to draw reliable conclusions from an IR evaluation experiment, researchers need test collections to be large. The more queries, documents and human assessors, the more precise our estimates of system effectiveness and the more reliable our comparisons between systems. But building large collections is expensive, so researchers need tools to estimate when a

[2] At least at this point. Section 8.2 discusses methods to do this.

collection is sufficiently large, and how their resources should be spent to get the largest benefit. Using Generalizability Theory, it was shown that collecting redundant relevance judgments is more reliable, but it does not pay off compared to just including more queries in the test collection. It was also shown that evaluating systems with a deeper cutoff is also more reliable, and arguably more realistic. But considering the actual judging effort, it is still more expensive than just using more queries. However, provided that query sets are relatively large, the loss in reliability is negligible, while interesting insight can be gained from deeper judgments. In general, the norm of $n_{\mathcal{Q}} = 100$ queries in MIREX seems overly expensive. It was shown that with about one third of the queries we can get stable estimates of relative effectiveness differences, and with about two thirds we can reliably estimate absolute scores.

Chapter 7

Learning Relevance Distributions

Even if the query set size needed to reliably evaluate some systems can be reduced to minimize the cost of building a test collection, we still need to judge all the top k documents retrieved by every system for every query, which can be expensive. This chapter introduces the notion of probabilistic evaluation, where effectiveness scores are estimated based on some model that estimates the relevance of documents. Only a small fraction of documents are actually judged by human assessors, so the total cost of the evaluation experiment is further reduced by minimizing judging effort. Two models to estimate relevance are presented here, each relying on different sets of features and to be used in different scenarios.

7.1 Probabilistic Evaluation

The traditional evaluation methodology used in MIREX is expensive in the sense that a complete set of relevance judgments is needed: all the top k documents retrieved by every system have to be judged for every query. In cases with many queries or many systems, this can be expensive. However, we may investigate how to reliably evaluate systems with an incomplete set of judgments, that is, avoid having to judge all documents and still be confident about the result of the experiment. The idea is to use random variables to represent relevance judgments [Carterette et al. 2006, Aslam and Yilmaz 2007]. The upside is that their value can be estimated fairly well for most documents; the downside is that these estimates will have some degree of error and uncertainty. The goal in this chapter is to develop models that can estimate relevance judgments as accurately and precisely as possible, allowing us to compute good estimates of effectiveness scores even if we have only very few judgments available.

Let R_d be a random variable representing the relevance level assigned to document d. The distribution of R_d is multinomial and depends on the relevance scale $\mathcal{L}$ that is used by human assessors. Its expectation and variance can be defined as:

$$\mathrm{E}[R_d] = \sum_{\ell \in \mathcal{L}} P(R_d = \ell) \cdot \ell \tag{7.1}$$

$$\mathrm{Var}[R_d] = \sum_{\ell \in \mathcal{L}} P(R_d = \ell) \cdot \ell^2 - \mathrm{E}[R_d]^2 \tag{7.2}$$

Whenever a human assessor judges document d and provides the actual judgment r_d, we fix $\mathrm{E}[R_d] \leftarrow r_d$ and $\mathrm{Var}[R_d] \leftarrow 0$, that is, no uncertainty about R_d. Because relevance judgments are now represented by random variables, effectiveness measures need to be reformulated so that effectiveness scores are also treated as random variables (this issue is dealt with in Chapter 8). That way, an effectiveness score will be defined over a distribution of possible assignments of relevance on the documents that are not judged yet. As such, we will be able to estimate effectiveness with some degree of confidence. In the case of having no relevance judgments, confidence on the estimates is minimum. The more judgments we have at our disposal, the more accurate our estimates and the more confident we are about them. In the ultimate case where all documents are judged, confidence would be 100%. The goal is thus to judge as few documents as possible to reach some level of confidence, say 95%.

7.2 Estimation of Relevance Judgments

In order to estimate the relevance of a document d with (7.1) we need to know what $P(R_d = \ell)$ is for each relevance level $\ell \in \mathcal{L}$. There are two immediate choices: a fixed distribution for all documents, maybe estimated from judgments in previous MIREX editions; or a distribution for each document as predicted by a model fitted with various features.

7.2.1 Fixed Distribution

A simple choice is to assume that every relevance assignment is equally likely with probability $1/n_{\mathcal{L}}$ [Carterette et al. 2006, Urbano and Schedl 2012]. For the Broad scale, all three relevance levels would have probability 1/3, while for the Fine scale each assignment would have probability 1/101. According to equations (7.1) and (7.2), an arbitrary unjudged document would have expectation $\mathrm{E}[R_d] = 1$ and variance $\mathrm{Var}[R_d] = 2/3$ in the Broad scale, and in the Fine scale it would have expectation $\mathrm{E}[R_d] = 50$ and variance $\mathrm{Var}[R_d] = 850$. I will not further consider the artificial scales from previous chapters, as they are all basically computed from the Fine scale judgments.

A more informative approach is to use again past judgments from MIREX to compute the prior distribution of relevance assignments. Figure 7.1 shows the historical distributions of judgments made in MIREX. Accordingly, an arbitrary unjudged document d would have expectation $\mathrm{E}[R_d] = 0.8959$ and variance $\mathrm{Var}[R_d] = 0.6338$ in the Broad scale, while in the Fine scale it would have expectation $\mathrm{E}[R_d] = 43.41$ and variance $\mathrm{Var}[R_d] = 900.4$.

7.2.2 Learned Distribution

A better alternative is to estimate the relevance of each document individually [Carterette 2007, Aslam and Yilmaz 2007, Urbano and Schedl 2013]. The problem reduces then to fitting a model that, given certain features about a query-document, allows us to estimate its

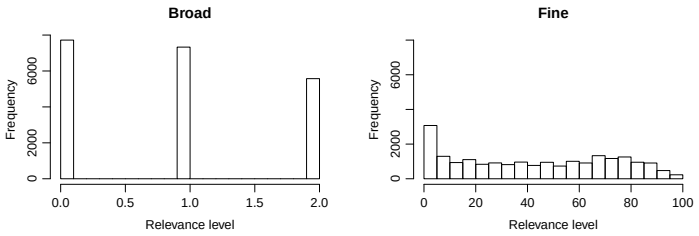

Figure 7.1: Distributions of relevance judgments made in MIREX 2007, 2009, 2010 and 2011.

relevance level. We may consider two frameworks for creating such a model: classification and regression. The classification approach is not appropriate because it ignores the order of the relevance levels. In the Broad scale, for instance, it means that if the true relevance of a document were $r_d = 0$, an estimation $\mathrm{E}[R_d] = 1$ would be as good as an estimation $\mathrm{E}[R_d] = 2$, while the latter is clearly worse. Linear regression is not appropriate either, because the predicted relevance could be well outside the $[0, n_{\mathcal{L}} - 1]$ limits. This could be solved with truncated regression [Long 1997], but we would still need to make assumptions about its underlying distribution. Multinomial regression has the same problem as classification, namely that it ignores the order of the levels in the outcome.

Ordinal logistic regression is the most appropriate framework [Liu and Agresti 2005, Carterette and Jones 2007]. The dependent variable R_d is modeled as an ordinal variable and, as opposed to classification and multinomial regression, the order of relevance levels is therefore taken into account. For an arbitrary relevance scale $\mathcal{L}$, the model for our ordinal variable is defined as:

$$\log \frac{P(R_d \geq \ell | \boldsymbol{\theta}_d)}{P(R_d < \ell | \boldsymbol{\theta}_d)} = \alpha_\ell + \sum_{k=1}^{|\boldsymbol{\theta}_d|} \beta_k \cdot \boldsymbol{\theta}_{d,k} \tag{7.3}$$

where β_k are the effect parameters to fit, α_ℓ is the fitted intercept for the particular relevance level ℓ, $\boldsymbol{\theta}_d$ is the feature vector for document d and $\boldsymbol{\theta}_{d,k}$ is the k-th feature value. Once the model is fitted, we can use the inverse logit function to compute $P(R_d \geq \ell | \boldsymbol{\theta}_d)$. Then, the probability of R_d being equal to some relevance level ℓ is computed as[1]:

$$P(R_d = \ell | \boldsymbol{\theta}_d) = P(R_d \geq \ell | \boldsymbol{\theta}_d) - P(R_d \geq \ell + 1 | \boldsymbol{\theta}_d) \tag{7.4}$$

This proportional odds model is generalized by the Vector Generalized Additive Model (VGAM) [Yee and Wild 1996], which is implemented in standard statistical packages such as R [Yee 2010] and facilitate the above calculations.

Therefore, the ordinal logistic framework allows us to estimate the distribution $P(R_d = \ell)$ in equation (7.1), which in turn enables the computation of expectation and variance as usual. As opposed to using the fixed distribution, this model is expected to produce estimates closer to the true relevance judgments and with reduced variance. As a result, the estimated effectiveness scores are expected to be more precise so that we require fewer judgments to pass a threshold on confidence such as 95%.

[1] Note that $P(R_d \geq 0 | \boldsymbol{\theta}_d)$ is always 1.

7.2.3 Features

Three types of features are considered for inclusion in the regression model to estimate relevance scores: output-based features, judgment-based features and audio-based features.

Output-based Features

This set of features only represent different aspects of the system outputs, so they can still be used when there are no judgments at all. For an arbitrary document d and query q:

- *fSYS*: fraction of systems that retrieved d for q within the top k documents. Intuitively, the more systems retrieve d, the more likely for it to be relevant to q.
- *fTEAM*: fraction of research teams participating in MIREX that retrieved d for q. Systems by the same team are likely to return similar documents, so the effect of *fSYS* could be biased if teams participate with a large number of systems. *fTEAM* can be used to reduce this bias.
- *OV*: degree of overlap between systems, to calibrate inherent similarities among systems when using the *fSYS* and *fTEAM* features. Overlap is defined as the amount of unique documents retrieved by all systems divided by the maximum possible. In the MIREX setting with no incompleteness, overlap is equal to $n_{\mathcal{R}}/(k \cdot n_{\mathcal{S}} \cdot n_{\mathcal{Q}})$.
- *aRANK*: average rank at which systems retrieved d for q. Documents retrieved closer to the top of the results lists are expected to be more similar to q.
- *sGEN*: whether the musical genre of d is the same as q (either 1 or 0), as documents of the same genre are usually considered similar to each other [Pohle 2010].
- *fGEN*: fraction of all documents retrieved for q by all systems that belong to the same musical genre as d does. Even if d does not have the same genre as q, if that genre is still commonly retrieved for q it is likely to be similar too; this feature is intended to model similarities between genres.
- *fART*: fraction of all documents retrieved for q that belong to the same artist as d does. Similarly, this feature is intended to model similarities between artists. Note that a feature like *sGEN* for artists does not make sense in the MIREX setting because all retrieved documents by q's artist are filtered out [Flexer and Schnitzer 2010, Downie et al. 2010].

Judgment-based Features

This set of features takes advantage of known judgments to produce better estimates:

- *aSYS*: average relevance of documents retrieved by the systems that retrieved d for q. Intuitively, a document retrieved by good systems is likely to be a good result.
- *aDOC*: average relevance of all the other documents retrieved for q. Likewise, this feature models query difficulty: if documents retrieved for q are not relevant, d is not likely to be relevant either.
- *aGEN*: average relevance of all the documents retrieved for q that belong to the same genre as d does. This is expected to improve *aDOC* on a per-genre basis.
- *aART*: average relevance of all the documents retrieved for q performed by the same artist as d. This is expected to improve *aDOC* on a per-artist basis.

Features *aSYS*, *aGEN* and *aART* are similar to *fSYS*, *fGEN* and *fART*. The former model relevance of systems, genres and artists based on known judgments, while the latter are based only on the system outputs.

Audio-based Features

Finally, I included a feature based on the actual audio content of the documents:

- *aSIM*: relevance level ℓ such that the similarity between d and all judged documents with relevance ℓ is maximum. According to the *Cluster Hypothesis* [Hearst and Pedersen 1996], the relevance of the document is likely to be the same as the relevance of the judged documents that are most similar to it.

The similarity between documents is computed based on the KL-divergence between Gaussian Mixture Models of Mel-Frequency Cepstral Coefficients [Mandel and Ellis 2005].

7.3 Results

The ordinal logistic regression model in (7.3) was initially fitted four times, using two different sets of features and using both the Broad and Fine scales. All relevance judgments made for the MIREX AMS task in 2007, 2009, 2010 and 2011 were used as examples to fit the models.

7.3.1 Goodness of Fit

For the first model, called M_{jud}, I started with a saturated model incorporating all features described above, iteratively simplifying it by removing non-significant effects. The final model includes features *fSYS*, *aSYS* and *aART*. All these features showed a very significant effect on the response ($p < 0.0001$). While other features did improve the model, they did so very marginally, so I decided to keep it as simple as possible. The predictions of M_{jud} are particularly good, with $R^2 = 0.9156$ in the Broad case and $R^2 = 0.9002$ in the Fine case[2]. When fitting the models for the Fine scale, I further simplified by breaking the scale down to $n_{\mathcal{L}} = 10$ levels rather than the original $n_{\mathcal{L}} = 101$ to reduce the number of α_ℓ parameters to fit in the model (7.3). The actual scale used was $\mathcal{L} := \{5, 15, 25, ..., 95\}$.

Even though M_{jud} produces very good estimates, we can only use it to estimate the relevance of documents for which we can compute both *aSYS* and *aART*. However, because our goal is to reduce the amount of judgments as much as possible, we will not be able to estimate the relevance of most documents until we have made a fair amount of judgments. A second model, called M_{out}, was therefore fitted using only output-based features. With this model, we can always estimate R_d, even when there are no judgments available at all. Proceeding as before, I simplified to a model using features *fSYS*, *OV*, *fART*, *sGEN*, *fGEN* and the *fSYS:OV* and *sGEN:fGEN* second order interactions. Despite all features showed again a statistically significant effect ($p < 0.0001$), the predictions are significantly worse than with M_{jud}, resulting in $R^2 = 0.3627$ and $R^2 = 0.3439$ respectively for the Broad and Fine judgments.

[2] The coefficient of determination R^2 is a goodness of fit indicator, measuring the proportion of variability in the response that is accounted for by the model; $R^2 = 1$ means that the model fits the data perfectly.

M_{out}										
Parameter	Broad					Fine				
	All	2007	2009	2010	1011	All	2007	2009	2010	1011
fSYS	123	91	53	121	93	140	107	70	137	97
OV	213	172	102	91	147	306	251	125	150	207
fSYS:OV	76	57	18	263	73	78	61	23	11	67
fART	295	191	257	319	133	283	174	276	290	125
sGEN	708	561	470	620	459	792	613	557	672	517
fGEN	2141	1428	1169	1888	2034	2313	1548	1250	2090	2148
sGEN:fGEN	279	174	92	263	328	478	321	183	447	496
R^2	0.3627	0.3459	0.3296	0.3780	0.4032	0.3439	0.3280	0.3175	0.3569	0.3786
RMSE	0.3254	0.3188	0.313	0.352	0.345	0.2412	0.2432	0.2341	0.2619	0.2501
Avg. *Var*	0.1054	0.1088	0.1121	0.0995	0.0989	0.0569	0.0577	0.0596	0.0538	0.0545

M_{jud}										
Parameter	Broad					Fine				
	All	2007	2009	2010	1011	All	2007	2009	2010	1011
fSYS	6	4	4	21	1	15	10	11	37	3
aSYS	144	103	104	115	106	109	74	88	89	75
aART	30810	23058	20753	26864	21705	41552	31337	28147	35913	29164
R^2	0.9156	0.9122	0.9089	0.9166	0.9245	0.9002	0.8987	0.8980	0.8991	0.9051
RMSE	0.1376	0.1301	0.1272	0.1427	0.1518	0.0922	0.091	0.0899	0.0936	0.0957
Avg. *Var*	0.0178	0.0167	0.0172	0.0175	0.0196	0.0069	0.0067	0.0069	0.0071	0.007

Table 7.1: Likelihood-ratio Chi-squared statistics of all effects fitted in each model, along with R^2 score, rooted mean squared error between predicted and actual scores, and average variance of estimates for M_{out} (top) and M_{jud} (bottom) models. Models for year Y are fitted excluding all judgments from Y, and tested against those.

Table 7.1 shows the Likelihood-ratio Chi-squared statistics of all effects fitted in these models (under column "All")[3]. Within model M_{out}, the best effects are related to the genre and artist metadata, confirming that these are indeed good features to estimate the similarity between two music excerpts [Pohle 2010, Flexer and Schnitzer 2010]. Within the M_{jud} model, the best effect is clearly *aART*, showing again that if two songs by two artists are similar, other songs performed by them are likely to be similar too[4]. This supports the decision in MIREX of filtering out documents by the same artist as the query's; they are very likely going to be similar to it. At the bottom of the table we can see how well the models predict relevance judgments. In particular, the table reports the rooted mean squared error (RMSE) between the actual and the estimated judgments as per (7.1), as well as the average variance as per (7.2) over all judgments. In order for comparisons across the Broad and Fine scale to be meaningful, all scores were normalized between 0 and 1; the actual scales used here for comparison are therefore $\{0, 0.5, 1\}$ and $\{0.05, 0.15, \ldots, 0.95\}$. As the table shows, errors in the Fine scale are about one third smaller than in the Broad scale, and the variance of the estimates is about half as much. This means that not only are the Fine estimates more accurate and therefore closer to the actual judgments, but also

[3] Effects with larger values account for a larger portion of the variability in the response.
[4] This result leads to the natural use of MIREX song similarity judgments to build a ground truth of artist similarity, that is, two artists are similar to the extent their songs are similar [Schedl et al. 2013b].

that our confidence in the estimates is larger, so we should need fewer judgments overall. Comparing models, we see that M_{out} produces more than twice the error M_{jud} does, and the variance of the estimates is about tenfold. Similarly then, using M_{jud} estimates will not only produce more accurate results but also increase confidence.

7.3.2 Model Cross-Validation

To cross-validate the M_{out} and M_{jud} models, I fitted them again but removing portions of the available judgments. In particular, for every year $Y \in \{2007, 2009, 2010, 2011\}$ the models were fitted again, but excluding all judgments made for the MIREX Y edition. Once fitted, these models were then run with the features computed for documents in year Y to show their predictive power. This way we can rule out any overfitting in the larger models fitted in the previous section.

Similarly, Table 7.1 shows the goodness of fit statistics of these models. As can be seen, all models are fitted similarly well compared to the models using all data. In terms of effects, it can be seen that the relative importance within the same model is the same: genre-based and artist-based features are the best ones in M_{out} models, while $a\mathit{ART}$ performs remarkably well in all M_{jud} models.

7.4 Discussion

In MIREX 2006 three different assessors provided judgments for each query-document pair. If we consider one assessor's judgments as the truth, and the other's as mere estimates, we find that the average rooted mean squared error among assessors was 0.3963 with the Broad scale and 0.3116 with the Fine scale, again normalizing scales between 0 and 1. These (disagreement) errors are extremely similar to the errors of the M_{out} models (0.3627 and 0.3439), and quite larger than the errors of the M_{jud} model (0.1376 and 0.0922). Therefore, the errors we make when using these estimates are comparable to the differences we should expect just by having a different human assessor in the first place. The MIREX evaluations assume arbitrary final users, so these errors can be ignored for all practical purposes under the current MIREX setting. If the evaluations moved towards user-centric experiments, these estimates would be erroneous to the degree reported here.

In an scenario where we want to run an evaluation experiment with no judgments yet, we may proceed as follows. Since M_{out} is based only on the output of systems and metadata of the documents, we can initially estimate all relevance judgments with it. Effectiveness scores can then be estimated (see Chapter 8), and if the confidence in our results is not high enough we can proceed to judge some documents. Based on these new judgments, M_{jud} can be used to calculate a better estimate on some documents, after which we estimate again effectiveness scores. If confidence is high enough, we can stop judging; if it is not, we just iterate again and select another document for judging.

It should be noted that these models can be used in the MIREX setting because the genre and artist metadata are known to organizers (not to participants). In case no metadata is known about documents, we can still use the fixed distribution from historical MIREX judgments. The rooted mean squared error using this fixed distribution is 0.3981 with

the Broad scale and 0.3953 with the Fine scale. Even though these errors are larger than with M_{out}, the errors with the Broad scale are comparable to the disagreement observed between human assessors. Therefore, this fixed distribution could be used initially when no metadata is known. Note that if we did have information about artists but not about genres, we could still use M_{jud} to predict relevance when there are known judgments, and use the fixed historical distribution when there are not. Having genre metadata is more complex than artist information because it is subjective and usually multivalued [Lippens et al. 2004, Seyerlehner et al. 2010a, Scaringella et al. 2006], so this scenario may come up quite frequently.

7.5 Summary

The probabilistic approach to IR evaluation is introduced in this chapter. The goal is to reliably evaluate systems with incomplete relevance judgments, that is, with many documents still unjudged. Effectiveness scores are represented with random variables over the space of possible relevance assignments, and they are estimated based on models that predict the relevance of documents.

Two models are presented for this purpose. Model M_{out} is based on features computed from the system outputs and metadata of the documents, so they can always be used to estimate relevance even when no judgments are available at all. Model M_{jud} uses information about available judgments to better estimate the relevance of documents, and it can be used in conjunction with the first model to compute more accurate and precise estimates. Appendix A reports the fitted model parameters and shows an example of application for new evaluation experiments.

Comparing the Broad and Fine scales, it is shown that relevance judgments are better estimated within the Fine scale. Considering the disagreement among human assessors when making relevance judgments, it is shown that these models produce even smaller differences in the predictions, so estimation errors can be ignored in practice under the current MIREX evaluation setting.

Chapter 8
Low-Cost Evaluation

Chapter 7 introduced the probabilistic framework for evaluation in Information Retrieval. Under that framework, the relevance of a document is represented by a random variable, so in the end the effectiveness scores that result from the evaluation experiment are random variables too. This chapter presents the probabilistic definition for several effectiveness measures. This framework is simulated with MIREX data, showing that we can reliably estimate differences between systems with as little as 2% of the judgments usually required. In addition, it is shown that quite good estimates of the ranking of systems can be computed even when there are no judgments at all.

8.1 Probabilistic Effectiveness Measures

This section presents a probabilistic definition of some effectiveness measures to reflect the representation of relevance judgments as random variables. The measures included are CG_l@5 and DCG_l@5 for being the most stable for relative effectiveness scores, and RBP_l@5 and $nDCG_l$@5 for being the most stable for absolute effectiveness scores (see Chapter 6). The gain function used is again linear. The probabilistic formulations for absolute scores are presented first, followed by the formulations for differences.

8.1.1 Absolute Effectiveness Scores

Cumulative Gain

The definition in (3.3) is followed here, fixing the gain function to the linear $g(\ell) = \ell$:

$$CG_l@k = \frac{1}{k}\sum_{i=1}^{k}\frac{r_{\mathsf{A}_i}}{n_{\mathcal{L}}-1} = \frac{1}{k(n_{\mathcal{L}}-1)}\sum_{i=1}^{k} r_{\mathsf{A}_i}$$

When considering relevance as a random variable, $CG_l@k$ becomes a random variable that equals the sum of independent random variables. For simplicity, let η_{CG_l} be the normalization factor $k(n_{\mathcal{L}}-1)$, which is constant. The expectation and variance for $CG_l@k$ are:

$$\mathrm{E}[CG_l@k] = \frac{1}{\eta_{CG_l}} \sum_{i=1}^{k} \mathrm{E}[R_{\mathsf{A}_i}] \tag{8.1}$$

$$\mathrm{Var}[CG_l@k] = \frac{1}{\eta^2_{CG_l}} \sum_{i=1}^{k} \mathrm{Var}[R_{\mathsf{A}_i}] \tag{8.2}$$

Discounted Cumulative Gain

The formulation in (3.4) is similarly followed, fixing the gains to the linear case:

$$DCG_l@k = \frac{\sum_{i=1}^{k} r_{\mathsf{A}_i} / \log_2(i+1)}{\sum_{i=1}^{k} (n_{\mathcal{L}} - 1) / \log_2(i+1)} = \frac{1}{\sum_{i=1}^{k} (n_{\mathcal{L}} - 1)/\log_2(i+1)} \sum_{i=1}^{k} \frac{r_{\mathsf{A}_i}}{\log_2(i+1)}$$

Similarly, let us define again a normalization factor $\eta_{DCG_l} = \sum_{i=1}^{k} (n_{\mathcal{L}} - 1)/\log_2(i+1)$, which is constant. The expectation and variance for $DCG_l@k$ are:

$$\mathrm{E}[DCG_l@k] = \frac{1}{\eta_{DCG_l}} \sum_{i=1}^{k} \frac{\mathrm{E}[R_{\mathsf{A}_i}]}{\log_2(i+1)} \tag{8.3}$$

$$\mathrm{Var}[DCG_l@k] = \frac{1}{\eta^2_{DCG_l}} \sum_{i=1}^{k} \frac{\mathrm{Var}[R_{\mathsf{A}_i}]}{\log_2(i+1)^2} \tag{8.4}$$

Normalized Discounted Cumulative Gain

The formulation in (2.10) is used, fixing gains to the linear case:

$$nDCG_l@k = \frac{\sum_{i=1}^{k} r_{\mathsf{A}_i} / \log_2(i+1)}{\sum_{i=1}^{k} r_{\mathsf{I}_i} / \log_2(i+1)}$$

Unlike with $CG_l@k$ and $DCG_l@k$, the normalization factor in $nDCG_l@k$ is not constant; it is also a random variable. The ideal list of results is computed by sorting all documents by $\mathrm{E}[R_d]$ in descending order. We can work out the numerator and the denominator separately; their expectation and variance are like (8.3) and (8.4) but without the η_{DCG_l} constant. Once they are calculated, expectation and variance for $nDCG_l@k$ can be approximated with the Delta Method using Taylor series expansion [Casella and Berger 2002]:

$$\mathrm{E}\left[\frac{X}{Y}\right] \approx \frac{\mathrm{E}[X]}{\mathrm{E}[Y]} \tag{8.5}$$

$$\mathrm{Var}\left[\frac{X}{Y}\right] \approx \left(\frac{\mathrm{E}[X]}{\mathrm{E}[Y]}\right)^2 \left(\frac{\mathrm{Var}[X]}{\mathrm{E}[X]^2} + \frac{\mathrm{Var}[Y]}{\mathrm{E}[Y]^2} - 2\frac{\mathrm{Cov}\,[X,Y]}{\mathrm{E}[X]\,\mathrm{E}[Y]}\right) \tag{8.6}$$

The numerator and the denominator are not independent of each other in $nDCG_l@k$ because some of the top k documents retrieved by A may be in the ideal ranking I. However, for simplicity I will assume independence so that covariance is zero in equation (8.6).

Rank-Biased Precision

The formulation in (5.8) is followed here, but fixing gains to the linear case:

$$RBP_l@k = \frac{\sum_{i=1}^{k} r_{\mathsf{A}_i} \cdot p^{i-1}}{\sum_{i=1}^{k} r_{\mathsf{I}_i} \cdot p^{i-1}}$$

As with $nDCG_l@k$, both the numerator and the denominator are random variables, so we work them out separately. In the case of the numerator, expectation and variance are:

$$\mathrm{E} = \sum_{i=1}^{k} \mathrm{E}[R_{\mathsf{A}_i}] \cdot p^{i-1} \tag{8.7}$$

$$\mathrm{Var} = \sum_{i=1}^{k} \mathrm{Var}[R_{\mathsf{A}_i}] \cdot p^{2i-2} \tag{8.8}$$

Expectation and variance for the denominator are calculated the same way, but substituting A_i with I_i. Finally, to compute expectation and variance for $RBP_l@k$, we use the Taylor series approximation in (8.5) and (8.6).

8.1.2 Differences in Effectiveness Scores

Let us consider the difference $\Delta CG_l k$ between two systems A and B:

$$\Delta CG_l@k = \frac{1}{k}\sum_{i=1}^{k} \frac{r_{\mathsf{A}_i}}{n_{\mathcal{L}} - 1} - \frac{1}{k}\sum_{i=1}^{k} \frac{r_{\mathsf{B}_i}}{n_{\mathcal{L}} - 1} = \frac{1}{k(n_{\mathcal{L}} - 1)} \sum_{i=1}^{k} r_{\mathsf{A}_i} - r_{\mathsf{B}_i}$$

Taking relevance probabilistically, the expectation and variance would be:

$$\mathrm{E}[\Delta CG_l@k] = \frac{1}{k(n_{\mathcal{L}} - 1)} \sum_{i=1}^{k} \mathrm{E}[R_{\mathsf{A}_i}] - \mathrm{E}[R_{\mathsf{B}_i}]$$

$$\mathrm{Var}[\Delta CG_l@k] = \frac{1}{k^2(n_{\mathcal{L}} - 1)^2} \sum_{i=1}^{k} \mathrm{Var}[R_{\mathsf{A}_i}] + \mathrm{Var}[R_{\mathsf{B}_i}]$$

Now let us assume that both A and B retrieved the same document d within the top k documents. In a measure like $CG_l@k$ such document would have no effect in the computation of $\Delta CG_l@k$. The document would contribute $r_d/k(n_{\mathcal{L}} - 1)$ to the effectiveness of both A and B, so in the end it would have absolutely no effect in $\Delta CG_l@k$, *regardless of what* r_d *really is*. This is reflected on the expectation above because the overall contribution of d is $\mathrm{E}[R_d] - \mathrm{E}[R_d] = 0$. However, it is not reflected on variance. Document d does not increase variance because it does not affect the $\Delta CG_l@k$ score, and yet, $\mathrm{Var}[R_d]$ is added twice to the total variance. This is due to the fact that we are considering R_{A_i} and R_{B_i} independent of each other, when in reality, they could actually be the same variable, referring to the same document. In the particular case of $\Delta CG@k$, this happens not only when $\mathsf{A}_i = \mathsf{B}_i$, but in the general case where $\exists i, j \leq k : \mathsf{A}_i = \mathsf{B}_j$. In this section, differences in effectiveness are formulated to account for this possibility.

Difference in Cumulative Gain

Let us define a $CG_l@k$ score by iterating all documents in the collection rather than just the top k retrieved [Carterette et al. 2006]:

$$CG_l@k = \frac{1}{\eta_{CG_l}} \sum_{i=1}^{k} r_{\mathsf{A}_i} = \frac{1}{\eta_{CG_l}} \sum_{d \in \mathcal{D}} r_d \cdot \mathbb{1}\left(\mathsf{A}_d^{-1} \leq k\right)$$

where A_d^{-1} is the rank at which document d is retrieved by system A (i.e. $\mathsf{A}_{\mathsf{A}_i}^{-1} = i$). The Iverson Bracket makes a document contribute to the overall summation only if it was retrieved within the top k documents. Expectation and variance are defined similarly to (8.1) and (8.2), but restricting what documents contribute to the summation:

$$\mathrm{E}[CG_l@k] = \frac{1}{\eta_{CG_l}} \sum_{d \in \mathcal{D}} \mathrm{E}[R_d]\, \mathbb{1}\left(\mathsf{A}_d^{-1} \leq k\right) \tag{8.9}$$

$$\mathrm{Var}[CG_l@k] = \frac{1}{\eta_{CG_l}^2} \sum_{d \in \mathcal{D}} \mathrm{Var}[R_d]\, \mathbb{1}\left(\mathsf{A}_d^{-1} \leq k\right) \tag{8.10}$$

The difference $\Delta CG_l@k$ between two systems A and B can now be defined as:

$$\Delta CG_l@k = \frac{1}{\eta_{CG_l}} \sum_{d \in \mathcal{D}} r_d \left(\mathbb{1}\left(\mathsf{A}_d^{-1} \leq k\right) - \mathbb{1}\left(\mathsf{B}_d^{-1} \leq k\right)\right)$$

Finally, the expectation and variance for a $\Delta CG_l@k$ score are[1]:

$$\mathrm{E}[\Delta CG_l@k] = \frac{1}{\eta_{CG_l}} \sum_{d \in \mathcal{D}} \mathrm{E}[R_d] \left(\mathbb{1}\left(\mathsf{A}_d^{-1} \leq k\right) - \mathbb{1}\left(\mathsf{B}_d^{-1} \leq k\right)\right) \tag{8.11}$$

$$\mathrm{Var}[\Delta CG_l@k] = \frac{1}{\eta_{CG_l}^2} \sum_{d \in \mathcal{D}} \mathrm{Var}[R_d] \left|\mathbb{1}\left(\mathsf{A}_d^{-1} \leq k\right) - \mathbb{1}\left(\mathsf{B}_d^{-1} \leq k\right)\right| \tag{8.12}$$

Formulating differences of effectiveness by iterating all documents in the collection we solve the issue of independence of random variables.

Difference in Discounted Cumulative Gain

Likewise, let us define $DCG_l@k$ by iterating all documents in the collection rather than just the top k:

$$DCG_l@k = \frac{1}{\eta_{DCG_l}} \sum_{d \in \mathcal{D}} \frac{r_d}{\log_2(\mathsf{A}_d^{-1} + 1)} \cdot \mathbb{1}\left(\mathsf{A}_d^{-1} \leq k\right)$$

The difference $\Delta DCG_l@k$ between two systems A and B is then:

$$\Delta DCG_l@k = \frac{1}{\eta_{DCG_l}} \sum_{d \in \mathcal{D}} r_d \left(\frac{\mathbb{1}\left(\mathsf{A}_d^{-1} \leq k\right)}{\log_2(\mathsf{A}_d^{-1} + 1)} - \frac{\mathbb{1}\left(\mathsf{B}_d^{-1} \leq k\right)}{\log_2(\mathsf{B}_d^{-1} + 1)}\right)$$

Finally, expectation and variance are:

$$\mathrm{E}[\Delta DCG_l@k] = \frac{1}{\eta_{DCG_l}} \sum_{d \in \mathcal{D}} \mathrm{E}[R_d] \left(\frac{\mathbb{1}\left(\mathsf{A}_d^{-1} \leq k\right)}{\log_2(\mathsf{A}_d^{-1} + 1)} - \frac{\mathbb{1}\left(\mathsf{B}_d^{-1} \leq k\right)}{\log_2(\mathsf{B}_d^{-1} + 1)}\right) \tag{8.13}$$

$$\mathrm{Var}[\Delta DCG_l@k] = \frac{1}{\eta_{DCG_l}^2} \sum_{d \in \mathcal{D}} \mathrm{Var}[R_d] \left|\frac{\mathbb{1}\left(\mathsf{A}_d^{-1} \leq k\right)}{\log_2(\mathsf{A}_d^{-1} + 1)} - \frac{\mathbb{1}\left(\mathsf{B}_d^{-1} \leq k\right)}{\log_2(\mathsf{B}_d^{-1} + 1)}\right|^2 \tag{8.14}$$

[1] The absolute value of the Iverson Brackets is used in the variance so all documents have a positive contribution to the total variance.

Note that any document retrieved at the same rank by both systems will not contribute to expectation or variance, but if the rank is different for each system it does contribute because the discount functions are different.

Difference in Normalzied Discounted Cumulative Gain

Formulated by iterating all documents, $\Delta nDCG_l@k$ is defined similar to $\Delta DCG_l@k$ but with the exception that the normalization factor is a random variable rather than a constant:

$$\Delta nDCG_l@k = \frac{\sum_{d\in\mathcal{D}} r_d \left(\frac{\mathbb{1}\left(\mathsf{A}_d^{-1} \le k\right)}{\log_2(\mathsf{A}_d^{-1}+1)} - \frac{\mathbb{1}\left(\mathsf{B}_d^{-1} \le k\right)}{\log_2(\mathsf{B}_d^{-1}+1)} \right)}{\sum_{i=1}^{k} r_{\mathsf{I}_i}/\log_2(i+1)}$$

Expectation and variance can again be worked out by calculating the numerator and the denominator independently. For the numerator, expectation and variance are just like (8.13) and (8.14) but ignoring the η_{DCG_l} factor. For the denominator, they are calculated as in (8.3) and (8.4), but ignoring the η_{DCG_l} factor again and substituting A_i for I_i. Once we have numerator and denominator, final expectation and variance for $\Delta nDCG@k$ are computed with the Taylor series approximation in (8.5) and (8.6).

Difference in Rank-Biased Precision

The formulation by iterating all documents is very similar to that of $\Delta nDCG_l@k$, except that the discount function is $1/p^{i-1}$ instead of $\log_2(i+1)$:

$$\Delta RBP_l@k = \frac{\sum_{d\in\mathcal{D}} r_d \left(p^{\mathsf{A}_d^{-1}-1} \cdot \mathbb{1}\left(\mathsf{A}_d^{-1} \le k\right) - p^{\mathsf{B}_d^{-1}-1} \cdot \mathbb{1}\left(\mathsf{B}_d^{-1} \le k\right) \right)}{\sum_{i=1}^{k} r_{\mathsf{I}_i} \cdot p^{i-1}}$$

Expectation and variance are again calculated for the numerator and the denominator separately. For the numerator we have:

$$\mathrm{E} = \sum_{d\in\mathcal{D}} \mathrm{E}[R_d] \cdot \left(p^{\mathsf{A}_d^{-1}-1} \cdot \mathbb{1}\left(\mathsf{A}_d^{-1} \le k\right) - p^{\mathsf{B}_d^{-1}-1} \cdot \mathbb{1}\left(\mathsf{B}_d^{-1} \le k\right) \right) \tag{8.15}$$

$$\mathrm{Var} = \sum_{d\in\mathcal{D}} \mathrm{Var}[R_d] \cdot \left| p^{\mathsf{A}_d^{-1}-1} \cdot \mathbb{1}\left(\mathsf{A}_d^{-1} \le k\right) - p^{\mathsf{B}_d^{-1}-1} \cdot \mathbb{1}\left(\mathsf{B}_d^{-1} \le k\right) \right|^2 \tag{8.16}$$

The denominator is computed as in (8.7) and (8.8), but substituting A_i for I_i. Finally, expectation and variance for $\Delta RBP_l@k$ is calculated using the Taylor series approximation.

8.1.3 Averages over a Sample of Queries

All probabilistic formulations above can be used to estimate the effectiveness score λ_q for a single query q. However, in the end we need to estimate the average $\overline{\lambda}_{\mathcal{Q}}$ over a sample of queries $\mathcal{Q}$ in a test collection. For some arbitrary measure Λ, the expectation and variance of the average are computed by iterating all individual estimates:

$$\mathrm{E}\left[\overline{\Lambda}_{\mathcal{Q}}\right] = \frac{1}{n_{\mathcal{Q}}} \sum_{q \in \mathcal{Q}} \mathrm{E}[\Lambda_q] \tag{8.17}$$

$$\mathrm{Var}\left[\overline{\Lambda}_{\mathcal{Q}}\right] = \frac{1}{n_{\mathcal{Q}}^2} \sum_{q \in \mathcal{Q}} \mathrm{Var}[\Lambda_q] \tag{8.18}$$

Following the Central Limit Theorem as in (4.6), we can compute a $100(1-2\alpha)\%$ confidence interval on $\overline{\Lambda}_{\mathcal{Q}}$ as follows:

$$\mathrm{E}\left[\overline{\Lambda}_{\mathcal{Q}}\right] \pm t_\alpha \sqrt{\mathrm{Var}\left[\overline{\Lambda}_{\mathcal{Q}}\right]} \tag{8.19}$$

where t_α is gain the quantile function of the t-distribution with $n_{\mathcal{Q}} - 1$ degrees of freedom.

If we are interested in the average difference $\overline{\Delta\lambda}_{\mathcal{Q},\mathsf{AB}}$ between two systems, expectation and variance are computed similar to (8.17) and (8.18). Based on the sign of $\mathrm{E}\left[\overline{\Delta\Lambda}_{\mathcal{Q},\mathsf{AB}}\right]$ we may conclude that one system is better than another as usual, but we should be able to do so with some degree of confidence, because in this probabilistic setting we are *estimating* $\overline{\Delta\lambda}_{\mathcal{Q},\mathsf{AB}}$. We can use the t-distribution again to approximate the probability that system A actually performs worse than system B and therefore the difference is negative:

$$P\left(\overline{\Delta\Lambda}_{\mathcal{Q},\mathsf{AB}} \leq 0\right) = F_t\left(\frac{\mathrm{E}\left[\overline{\Delta\Lambda}_{\mathcal{Q},\mathsf{AB}}\right]}{\sqrt{\mathrm{Var}\left[\overline{\Delta\Lambda}_{\mathcal{Q},\mathsf{AB}}\right]}}\right)$$

where F_t is the cumulative distribution function of the t-distribution with $n_{\mathcal{Q}} - 1$ degrees of freedom. If $P\left(\overline{\Delta\Lambda}_{\mathcal{Q},\mathsf{AB}} \leq 0\right)$ is low we can be confident that A outperforms B, and if it is large we can be confident that B outperforms A. Either way, we can therefore define the confidence in the estimated difference as the maximum between the probability of it being positive and it being negative:

$$C_{\mathcal{Q},\mathsf{AB}} = \max\left(P\left(\overline{\Delta\Lambda}_{\mathcal{Q},\mathsf{AB}} \leq 0\right), 1 - P\left(\overline{\Delta\Lambda}_{\mathcal{Q},\mathsf{AB}} \leq 0\right)\right) \tag{8.20}$$

Whenever we pass a threshold on confidence, say $C_{\mathcal{Q},\mathsf{AB}} \geq 95\%$, we can conclude with confidence which system performs better based on the sign of $\mathrm{E}\left[\overline{\Delta\Lambda}_{\mathcal{Q},\mathsf{AB}}\right]$. From this definition of confidence in an arbitrary pair of systems, we can define the confidence in the ranking as the average confidence over all pairs.

8.2 Evaluation Without Relevance Judgments

The first scenario to consider is that of estimating effectiveness when there are absolutely no judgments available. Soboroff et al. [2001] studied the problem of ranking systems submitted to TREC, showing that randomly considering documents as relevant correlated positively with the true TREC rankings, thus serving as a lower bound on what to expect just by random chance. Rather than using random judgments, we can use the estimates provided by the M_{out} regression model in Chapter 7. Note that the M_{jud} model cannot be used because it does require some known judgments. I simulated the evaluation in MIREX 2007, 2009, 2010 and 2011 as if there were no judgments. M_{out} was used to estimate the relevance of documents, based on which all effectiveness scores where estimated with the probabilistic measures in the previous section.

CG_l@5												
Year	Broad						Fine					
	Conf.	Acc.	τ	$\text{RMSE}_{\Delta\lambda}$	ρ	RMSE_{λ}	Conf.	Acc.	τ	$\text{RMSE}_{\Delta\lambda}$	ρ	RMSE_{λ}
2007	0.944	0.909	0.818	0.092	0.93	0.068	0.949	0.939	0.879	0.08	0.944	0.054
2009	0.937	0.933	0.867	0.11	0.963	0.082	0.942	0.924	0.848	0.086	0.95	0.067
2010	0.946	0.893	0.786	0.044	0.905	0.083	0.947	0.857	0.714	0.038	0.881	0.074
2011	0.933	0.941	0.882	0.055	0.975	0.048	0.934	0.935	0.869	0.041	0.967	0.03

DCG_l@5												
Year	Broad						Fine					
	Conf.	Acc.	τ	$\text{RMSE}_{\Delta\lambda}$	ρ	RMSE_{λ}	Conf.	Acc.	τ	$\text{RMSE}_{\Delta\lambda}$	ρ	RMSE_{λ}
2007	0.942	0.909	0.818	0.089	0.93	0.066	0.946	0.909	0.818	0.079	0.93	0.054
2009	0.929	0.962	0.924	0.111	0.975	0.083	0.934	0.914	0.829	0.086	0.95	0.066
2010	0.947	0.857	0.714	0.045	0.881	0.081	0.949	0.929	0.857	0.039	0.929	0.072
2011	0.925	0.948	0.895	0.057	0.975	0.05	0.926	0.928	0.856	0.042	0.965	0.032

$nDCG_l$@5												
Year	Broad						Fine					
	Conf.	Acc.	τ	$\text{RMSE}_{\Delta\lambda}$	ρ	RMSE_{λ}	Conf.	Acc.	τ	$\text{RMSE}_{\Delta\lambda}$	ρ	RMSE_{λ}
2007	0.936	0.909	0.818	0.068	0.93	0.168	0.942	0.909	0.818	0.069	0.93	0.192
2009	0.926	0.962	0.924	0.076	0.975	0.188	0.931	0.943	0.886	0.069	0.975	0.179
2010	0.944	0.893	0.786	0.03	0.905	0.172	0.946	0.929	0.857	0.037	0.929	0.166
2011	0.922	0.908	0.817	0.041	0.955	0.174	0.925	0.902	0.804	0.034	0.944	0.185

RBP_l@5												
Year	Broad						Fine					
	Conf.	Acc.	τ	$\text{RMSE}_{\Delta\lambda}$	ρ	RMSE_{λ}	Conf.	Acc.	τ	$\text{RMSE}_{\Delta\lambda}$	ρ	RMSE_{λ}
2007	0.935	0.909	0.818	0.068	0.93	0.169	0.941	0.909	0.818	0.07	0.93	0.193
2009	0.928	0.952	0.905	0.075	0.968	0.19	0.933	0.943	0.886	0.068	0.971	0.18
2010	0.942	0.893	0.786	0.032	0.905	0.172	0.944	0.857	0.714	0.039	0.881	0.167
2011	0.924	0.935	0.869	0.042	0.967	0.175	0.927	0.935	0.869	0.034	0.967	0.185

Table 8.1: Confidence and accuracy of the effectiveness estimates when evaluating systems in MIREX 2007, 2009, 2010 and 2011 without relevance judgments.

Table 8.1 shows the confidence in the rankings when making no judgments at all. Confidence is very high across collections and measures, with an average of 94% and always above 92%. The accuracy of the rankings, measured as the fraction of system pairs for which the sign of $\text{E}\left[\overline{\Delta\Lambda_{\mathcal{Q}}}\right]$ is correct, is always above 0.9 except for the 2010 collection. The average is 0.92, that is, confidence is slightly overestimated. The average Kendall τ correlation between the actual and the estimated ranking is 0.84. The overall performance in ranking systems is therefore quite good considering that *no judgments are needed*. Examining the actual effectiveness estimates we see clear differences across measures. The average rooted mean squared error of the estimates is 0.07 with the Broad scale and 0.056 with the Fine scale in the case of CG_l@5 and DCG_l@5, while they are 0.176 and 0.181 respectively in the case of $nDCG_l$@5 and RBP_l@5. The high Spearman ρ correlation coefficients show that there is no clear bias in these errors. When measuring relative error though, the error with CG_l@5 and DCG_l@5 is 0.068, while for $nDCG_l$@5 and RBP_l@5 it is 0.0533.

CG_l@5						
Conf.	Broad			Fine		
	In bin		Acc.	In bin		Acc.
[0.5, 0.6)	16	(4.5%)	0.625	14	(4%)	0.571
[0.6, 0.7)	19	(5.4%)	0.895	20	(5.7%)	0.85
[0.7, 0.8)	11	(3.1%)	0.636	9	(2.6%)	0.667
[0.8, 0.9)	25	(7.1%)	0.76	23	(6.5%)	0.739
[0.9, 0.95)	16	(4.5%)	0.812	14	(4%)	0.643
[0.95, 0.99)	31	(8.8%)	0.903	29	(8.2%)	1
[0.99, 1]	234	(66.5%)	0.996	243	(69%)	0.988
E[Accuracy]			0.929			0.926

DCG_l@5						
Conf.	Broad			Fine		
	In bin		Acc.	In bin		Acc.
[0.5, 0.6)	23	(6.5%)	0.826	22	(6.2%)	0.636
[0.6, 0.7)	14	(4%)	0.786	16	(4.5%)	0.812
[0.7, 0.8)	14	(4%)	0.571	11	(3.1%)	0.364
[0.8, 0.9)	22	(6.2%)	0.864	21	(6%)	0.762
[0.9, 0.95)	23	(6.5%)	0.87	19	(5.4%)	0.895
[0.95, 0.99)	24	(6.8%)	0.917	27	(7.7%)	0.926
[0.99, 1]	232	(65.9%)	0.996	236	(67%)	0.996
E[Accuracy]			0.938			0.921

$nDCG_l$@5						
Conf.	Broad			Fine		
	In bin		Acc.	In bin		Acc.
[0.5, 0.6)	25	(7.1%)	0.56	22	(6.2%)	0.545
[0.6, 0.7)	16	(4.5%)	0.812	16	(4.5%)	0.812
[0.7, 0.8)	13	(3.7%)	0.615	14	(4%)	0.429
[0.8, 0.9)	22	(6.2%)	0.864	22	(6.2%)	0.773
[0.9, 0.95)	20	(5.7%)	0.9	12	(3.4%)	1
[0.95, 0.99)	28	(8%)	0.929	31	(8.8%)	0.935
[0.99, 1]	228	(64.8%)	0.996	235	(66.8%)	0.996
E[Accuracy]			0.924			0.918

RBP_l@5						
Conf.	Broad			Fine		
	In bin		Acc.	In bin		Acc.
[0.5, 0.6)	25	(7.1%)	0.64	23	(6.5%)	0.652
[0.6, 0.7)	13	(3.7%)	0.846	13	(3.7%)	0.846
[0.7, 0.8)	16	(4.5%)	0.688	14	(4%)	0.571
[0.8, 0.9)	23	(6.5%)	0.87	24	(6.8%)	0.75
[0.9, 0.95)	11	(3.1%)	0.818	8	(2.3%)	0.875
[0.95, 0.99)	35	(9.9%)	0.943	34	(9.7%)	0.971
[0.99, 1]	229	(65.1%)	0.996	236	(67%)	0.992
E[Accuracy]			0.932			0.926

Table 8.2: Accuracy vs. confidence in the sign of estimates when evaluating systems in MIREX 2007, 2009, 2010 and 2011 without relevance judgments.

Despite the average confidence in the ranking generally corresponds to the average accuracy of the ranking estimates, it can be the case that this average confidence is biased by a few system comparisons for which we are extremely confident. The question thus is: how trustworthy are each of the individual estimates? The 352 system pairs from all four collections were divided by confidence in the sign of the individual $\mathrm{E}\left[\overline{\Delta\Lambda}_{\mathcal{Q}}\right]$ estimates. Ideally, we want accuracy to correspond to confidence (e.g. 0.80 accuracy in all pairs with 80% confidence). For each bin of confidence, Table 8.2 reports the number of estimates that fall inside and the accuracy of all estimates within the bin. We see that confidence is slightly overestimated in the $[0.9, 0.99)$ interval, but it is noticeable that the confidence in the majority of estimates is very high. On average, 67% of the times our confidence in the estimate is above 99%, and in those cases the sign is correct in 99.5% of the times. The expected accuracy of individual estimates is 0.927 in general.

8.3 Estimating Differences in Effectiveness

The second scenario to consider is also that of ranking systems, but making relevance judgments up to the point where we reach a certain level of confidence in the ranking. The idea is to use the estimates produced by M_{out} from the beginning, and iteratively judge documents and update estimates with M_{jud} when possible. The key here is choosing the best documents for judging. In principle, we want to judge those documents that are more informative to know the difference between two systems [Carterette et al. 2006]. As discussed in Section 8.1.2, in the case of $\Delta CG_l@k$ a document that has been retrieved by both systems does not change the difference, regardless of how relevant it is. Therefore, judging that document will not help us in determining which system performs better. For this case in particular, we want to judge documents retrieved by one system but not by the other.

When there are several systems to compare, some documents will be more informative than others if they can affect more than one system comparison. A weigh w_d can be computed for each document, as an indicator of how much it would contribute to judge it. These weights can be derived from the $\mathrm{E}[\Delta\Lambda]$ equations in Section 8.1.2. For example, in the case of $\Delta CG_l@k$ equation (8.11) shows that the contribution of a document d is $\mathrm{E}[R_d] \cdot \left(\mathbb{1}\left(\mathsf{A}_d^{-1} \leq k\right) - \mathbb{1}\left(\mathsf{B}_d^{-1} \leq k\right)\right)$. Iterating all system pairs, the weight can be defined as

$$w_d = \sum_{(\mathsf{A},\mathsf{B}) \in [\mathcal{S}]^2} \left|\mathbb{1}\left(\mathsf{A}_d^{-1} \leq k\right) - \mathbb{1}\left(\mathsf{B}_d^{-1} \leq k\right)\right|$$

which computes the number of system pairs affected by d. Intuitively, at all times we want to judge those documents that affect the most system pairs. Based on equation (8.13), the weight when computing $\Delta DCG_l@k$ is:

$$w_d = \sum_{(\mathsf{A},\mathsf{B}) \in [\mathcal{S}]^2} \left| \frac{\mathbb{1}\left(\mathsf{A}_d^{-1} \leq k\right)}{\log_2(\mathsf{A}_d^{-1} + 1)} - \frac{\mathbb{1}\left(\mathsf{B}_d^{-1} \leq k\right)}{\log_2(\mathsf{B}_d^{-1} + 1)} \right|$$

Similarly, the weight for $\Delta nDCG_l@k$ can be computed as:

$$w_d = \frac{\displaystyle\sum_{(\mathsf{A},\mathsf{B}) \in [\mathcal{S}]^2} \left| \frac{\mathbb{1}\left(\mathsf{A}_d^{-1} \leq k\right)}{\log_2(\mathsf{A}_d^{-1} + 1)} - \frac{\mathbb{1}\left(\mathsf{B}_d^{-1} \leq k\right)}{\log_2(\mathsf{B}_d^{-1} + 1)} \right|}{\sum_{i=1}^{k} \mathrm{E}[R_{\mathsf{l}_i}]/\log_2(i+1)}$$

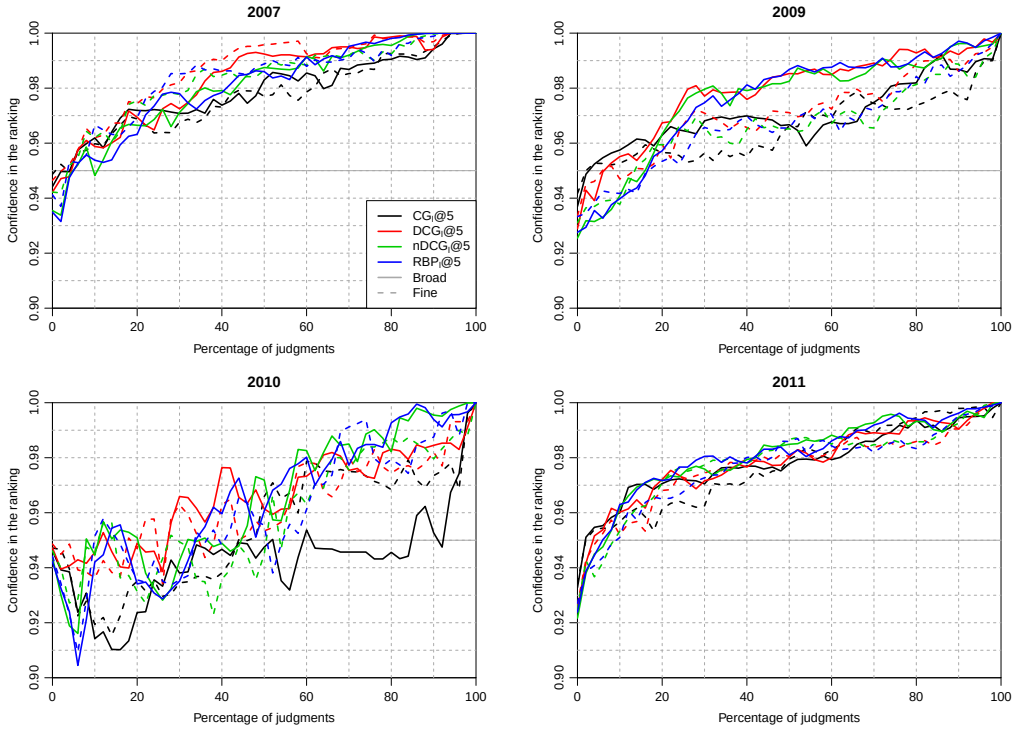

Figure 8.1: Confidence in the ranking of systems in MIREX 2007, 2009, 2010 and 2011 as the number of judgments increases.

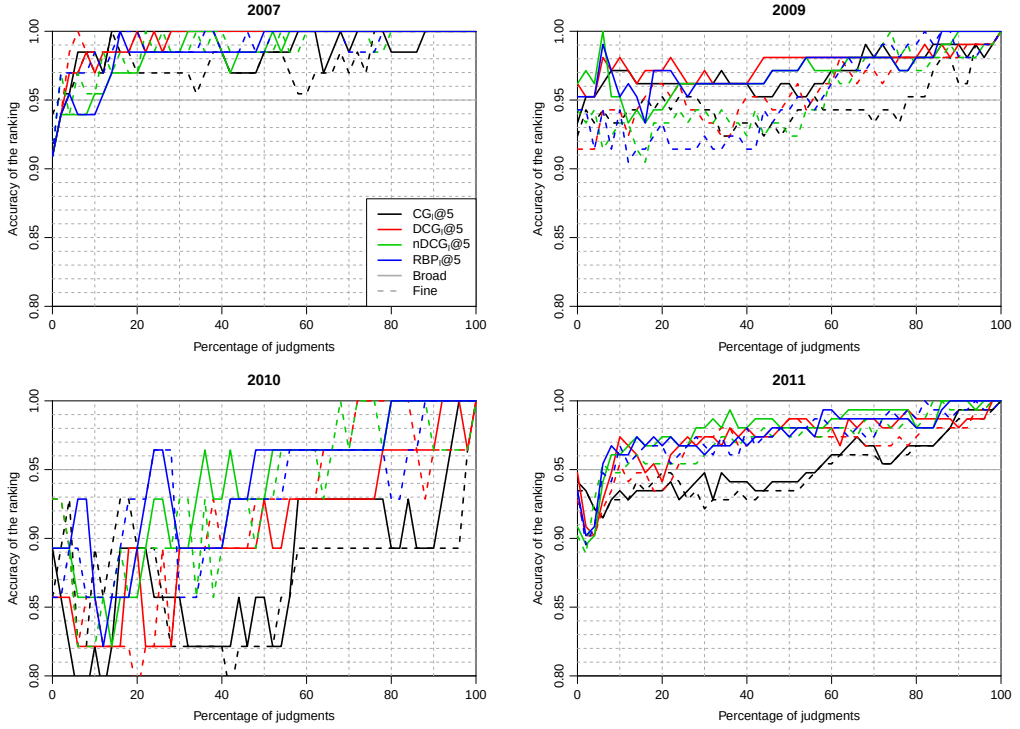

Figure 8.2: Accuracy of the estimated ranking of systems in MIREX 2007, 2009, 2010 and 2011 as the number of judgments increases.

CG_l@5										
Year	Broad					Fine				
	Judg.		Acc.	τ	RMSE$_{\Delta\lambda}$	Judg.		Acc.	τ	RMSE$_{\Delta\lambda}$
2007	93	(1.9%)	0.939	0.879	0.088	16	(0.3%)	0.939	0.879	0.08
2009	160	(2.4%)	0.943	0.886	0.104	95	(1.4%)	0.924	0.848	0.086
2010	13	(0.5%)	0.893	0.786	0.045	12	(0.4%)	0.857	0.714	0.039
2011	120	(1.9%)	0.935	0.869	0.041	123	(1.9%)	0.935	0.869	0.03

DCG_l@5										
Year	Broad					Fine				
	Judg.		Acc.	τ	RMSE$_{\Delta\lambda}$	Judg.		Acc.	τ	RMSE$_{\Delta\lambda}$
2007	78	(1.6%)	0.955	0.909	0.078	69	(1.4%)	0.924	0.848	0.072
2009	426	(6.3%)	0.981	0.962	0.099	400	(5.9%)	0.943	0.886	0.081
2010	252	(9.2%)	0.821	0.643	0.055	2	(0.1%)	0.929	0.857	0.039
2011	200	(3.2%)	0.889	0.778	0.04	181	(2.9%)	0.889	0.778	0.029

$nDCG_l$@5										
Year	Broad					Fine				
	Judg.		Acc.	τ	RMSE$_{\Delta\lambda}$	Judg.		Acc.	τ	RMSE$_{\Delta\lambda}$
2007	226	(4.7%)	0.955	0.909	0.058	160	(3.3%)	0.939	0.879	0.066
2009	827	(12.3%)	0.933	0.867	0.09	876	(13%)	0.924	0.848	0.07
2010	209	(7.6%)	0.857	0.714	0.053	220	(8%)	0.821	0.643	0.06
2011	347	(5.5%)	0.941	0.882	0.031	516	(8.2%)	0.948	0.895	0.021

RBP_l@5										
Year	Broad					Fine				
	Judg.		Acc.	τ	RMSE$_{\Delta\lambda}$	Judg.		Acc.	τ	RMSE$_{\Delta\lambda}$
2007	260	(5.4%)	0.955	0.909	0.059	120	(2.5%)	0.97	0.939	0.07
2009	1,099	(16.3%)	0.933	0.867	0.09	1,123	(16.7%)	0.933	0.867	0.068
2010	353	(12.9%)	0.857	0.714	0.059	263	(9.6%)	0.857	0.714	0.065
2011	348	(5.5%)	0.948	0.895	0.029	560	(8.9%)	0.961	0.922	0.022

Table 8.3: Confidence and accuracy of estimated differences in MIREX 2007, 2009, 2010 and 2011 when judging documents until 95% average confidence.

For $\Delta RBP_l@k$, the weight is defined as:

$$w_d = \frac{\sum_{(\mathsf{A},\mathsf{B})\in[\mathcal{S}]^2} \left| p^{\mathsf{A}_d^{-1}-1} \cdot \mathbb{1}\left(\mathsf{A}_d^{-1} \leq k\right) - p^{\mathsf{B}_d^{-1}-1} \cdot \mathbb{1}\left(\mathsf{B}_d^{-1} \leq k\right) \right|}{\sum_{i=1}^{k} \mathrm{E}[R_{\mathsf{l}_i}] \cdot p^{i-1}}$$

For the usual case of having several queries, the above weight definitions can be used by considering d a query-document pair rather than just a document, that is, the weight of a document for a particular query.

As in the previous section, I simulated the evaluation in MIREX 2007, 2009, 2010 and 2011 as if we started with no judgments and then used these weights to choose which documents to judge at each time, stopping when the average confidence in the ranking reaches 95%. All relevance scores were initially estimated with M_{out} and updated with M_{jud} every 20 documents judged. Figure 8.1 shows how the confidence in the ranking increases as more documents are judged, and Figure 8.2 shows the accuracy of the estimated rankings.

CG_l@5						
Conf.	Broad			Fine		
	In bin		Acc.	In bin		Acc.
[0.5, 0.6)	10	(2.8%)	0.4	14	(4%)	0.643
[0.6, 0.7)	13	(3.7%)	0.923	9	(2.6%)	0.556
[0.7, 0.8)	12	(3.4%)	0.833	10	(2.8%)	0.7
[0.8, 0.9)	21	(6%)	0.762	26	(7.4%)	0.808
[0.9, 0.95)	18	(5.1%)	0.722	12	(3.4%)	0.5
[0.95, 0.99)	28	(8%)	0.929	29	(8.2%)	0.931
[0.99, 1]	250	(71%)	0.992	252	(71.6%)	0.996
E[Accuracy]			0.935			0.926

DCG_l@5						
Conf.	Broad			Fine		
	In bin		Acc.	In bin		Acc.
[0.5, 0.6)	8	(2.3%)	0.5	7	(2%)	0.286
[0.6, 0.7)	15	(4.3%)	0.6	18	(5.1%)	0.611
[0.7, 0.8)	19	(5.4%)	0.526	16	(4.5%)	0.625
[0.8, 0.9)	14	(4%)	0.786	15	(4.3%)	0.6
[0.9, 0.95)	14	(4%)	0.786	17	(4.8%)	0.706
[0.95, 0.99)	30	(8.5%)	0.933	22	(6.2%)	0.955
[0.99, 1]	252	(71.6%)	1	257	(73%)	1
E[Accuracy]			0.923			0.915

$nDCG_l$@5						
Conf.	Broad			Fine		
	In bin		Acc.	In bin		Acc.
[0.5, 0.6)	10	(2.8%)	0.5	10	(2.8%)	0.4
[0.6, 0.7)	14	(4%)	0.571	13	(3.7%)	0.615
[0.7, 0.8)	15	(4.3%)	0.667	21	(6%)	0.667
[0.8, 0.9)	19	(5.4%)	0.737	14	(4%)	0.714
[0.9, 0.95)	12	(3.4%)	0.917	12	(3.4%)	0.75
[0.95, 0.99)	21	(6%)	0.952	16	(4.5%)	1
[0.99, 1]	261	(74.1%)	1	266	(75.6%)	1
E[Accuracy]			0.935			0.929

RBP_l@5						
Conf.	Broad			Fine		
	In bin		Acc.	In bin		Acc.
[0.5, 0.6)	15	(4.3%)	0.467	11	(3.1%)	0.727
[0.6, 0.7)	8	(2.3%)	0.875	13	(3.7%)	0.692
[0.7, 0.8)	16	(4.5%)	0.688	15	(4.3%)	0.6
[0.8, 0.9)	19	(5.4%)	0.737	20	(5.7%)	0.85
[0.9, 0.95)	10	(2.8%)	0.7	12	(3.4%)	0.917
[0.95, 0.99)	17	(4.8%)	1	13	(3.7%)	0.846
[0.99, 1]	267	(75.9%)	1	268	(76.1%)	1
E[Accuracy]			0.938			0.946

Table 8.4: Accuracy vs. confidence in the sign of estimates in MIREX 2007, 2009, 2010 and 2011 when judging documents until 95% average confidence.

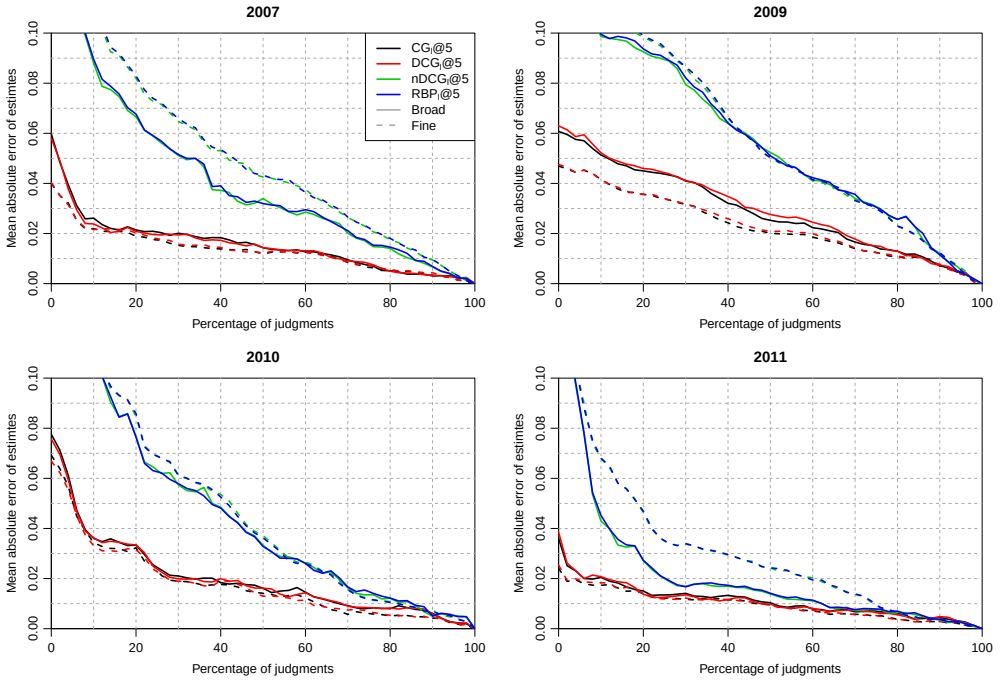

Figure 8.3: Accuracy of the absolute effectiveness estimates in MIREX 2007, 2009, 2010 and 2011 as the number of judgments increases.

Both confidence and accuracy are very high with virtually no judgments, though confidence seems overestimated when the number of judgments is small. As more judgments are made, confidence starts to underestimate accuracy.

Table 8.3 shows the results for all collections and measures when reaching 95% confidence in the ranking. The required judging effort is on average just 1.3% with $CG_l@k$, 3.8% with $DCG_l@k$, 7.8% with $nDCG_l@k$ and 9.7% with $RBP_l@k$. The accuracy of the ranking is similar in all cases, with an average of 0.92. The corresponding τ correlation is 0.84, and the rooted mean squared error between estimated and actual differences between systems is 0.06. Similarly, we are interested not only on the average accuracy of the estimates, but on how trustworthy each of them is. Table 8.4 reports again the number of estimates that fall inside each confidence bin and the average accuracy of all estimates within the bin. We see again that confidence is slightly overestimated in the [0.9,0.99) interval, but in the majority of cases confidence is above 99%, where almost all estimates are correct. Compared with Table 8.2, this is the clearest benefit when making some judgments; the fraction of estimates with more than 99% confidence goes up from 67% to 74%. The expected accuracy of the individual estimates also goes slightly up from 0.927 to 0.931.

8.4 Estimating Absolute Effectiveness

The third and final scenario to consider is that of estimating the absolute effectiveness scores of systems, making as few relevance judgments as possible to have a good estimate.

CG_l@5								
Year	Broad				Fine			
	Judg.		MAE$_\lambda$	ρ	Judg.		MAE$_\lambda$	ρ
2007	0	(0%)	0.06	0.93	0	(0%)	0.04	0.944
2009	0	(0%)	0.061	0.963	0	(0%)	0.047	0.95
2010	0	(0%)	0.078	0.905	0	(0%)	0.069	0.881
2011	0	(0%)	0.036	0.975	0	(0%)	0.024	0.967

DCG_l@5								
Year	Broad				Fine			
	Judg.		MAE$_\lambda$	ρ	Judg.		MAE$_\lambda$	ρ
2007	0	(0%)	0.058	0.93	0	(0%)	0.04	0.93
2009	0	(0%)	0.063	0.975	0	(0%)	0.048	0.95
2010	0	(0%)	0.076	0.881	0	(0%)	0.067	0.929
2011	0	(0%)	0.038	0.975	0	(0%)	0.025	0.965

$nDCG_l$@5								
Year	Broad				Fine			
	Judg.		MAE$_\lambda$	ρ	Judg.		MAE$_\lambda$	ρ
2007	140	(2.9%)	0.117	0.972	0	(0%)	0.186	0.93
2009	34	(0.5%)	0.174	0.975	0	(0%)	0.172	0.975
2010	0	(0%)	0.171	0.905	0	(0%)	0.165	0.929
2011	112	(1.8%)	0.122	0.969	0	(0%)	0.184	0.944

RBP_l@5								
Year	Broad				Fine			
	Judg.		MAE$_\lambda$	ρ	Judg.		MAE$_\lambda$	ρ
2007	120	(2.5%)	0.121	0.979	0	(0%)	0.187	0.93
2009	20	(0.3%)	0.177	0.968	0	(0%)	0.173	0.971
2010	0	(0%)	0.17	0.905	0	(0%)	0.165	0.881
2011	100	(1.6%)	0.126	0.967	0	(0%)	0.184	0.967

Table 8.5: Accuracy of estimated absolute scores in MIREX 2007, 2009, 2010 and 2011 when judging documents until expected error is ±0.05.

Likewise, we need to define a weight for each document so that we judge the one that will be the most informative for our purpose. This time, judging a document that does not tell us anything about the difference between systems can still be useful to estimate their absolute score. Intuitively, we want to judge those documents that help us reduce variance:

$$w_d = \sum_{\mathsf{A} \in \mathcal{S}} \mathrm{Var}\left[\overline{\Lambda}_{\mathcal{Q},\mathsf{A}}\right] \cdot \mathbb{1}\left(\mathsf{A}_d^{-1} \leq k\right)$$

That is, every system that retrieved the document contributes its variance to the weight. In the end, we choose the document with the highest weight, which is the one that will reduce variance the most across systems. As in the previous scenarios, I simulated the evaluation in MIREX 2007, 2009, 2010 and 2011 as if we started with no judgments and then used weights to choose which documents to judge. For the stopping condition we can fix a threshold on how much error we find acceptable in the estimates; let us assume a threshold of ±0.05. From (8.19), we therefore want $t_\alpha \sqrt{\mathrm{Var}\left[\overline{\Lambda}_{\mathcal{Q}}\right]} \leq 0.05$. At the 0.95

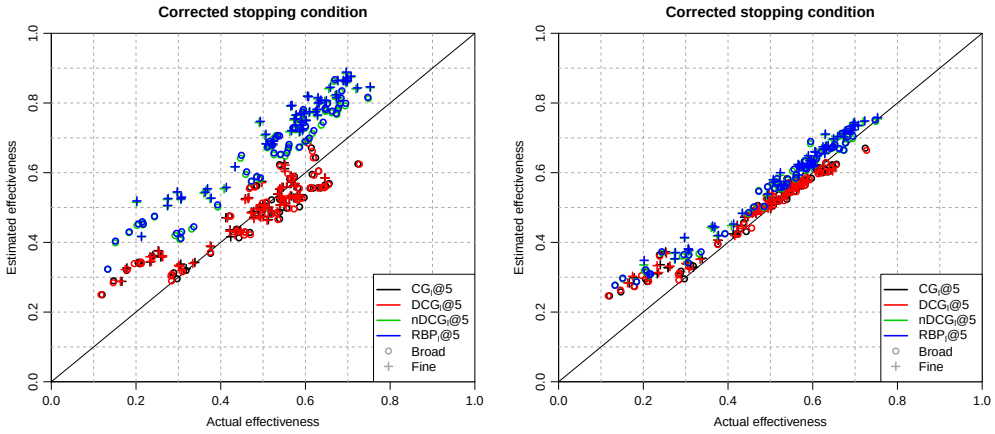

Figure 8.4: Estimated vs. actual absolute effectiveness scores in MIREX 2007, 2009, 2010 and 2011 when judging documents until expected error is ±0.05 with an uncorrected (left) or corrected (right) stopping condition.

confidence level, this means that our objective is $\sqrt{\text{Var}\left[\overline{\Lambda}_{\mathcal{Q}}\right]} \leq 0.0252$. When the average variance of the estimates satisfies this threshold, we can stop judging.

Figure 8.3 shows how the error of the estimates decreases as the number of judgments increases. It is clear that $CG_l@5$ and $DCG_l@5$ consistently produce smaller errors than $nDCG_l@5$ and $RBP_l@5$, and therefore need fewer judgments. This is because $nDCG_l@5$ and $RBP_l@5$ need to estimate not only the relevance of the top-5 documents retrieved, but also the relevance of all other documents to estimate the ideal ranking. Table 8.5 shows the accuracy of the estimates when judging documents until the threshold on variance is reached and therefore the expected mean absolute error (MAE) is ±0.05. We can see that in almost all cases no judgments are even needed because the estimates seem sufficiently good with the M_{out} model alone (see Section 8.2). The average error in $CG_l@5$ and $DCG_l@5$ is 0.052, meaning that the stopping condition worked as expected. However, the average error is 0.162 in $nDCG_l@5$ and $RBP_l@5$, which is thrice as expected. The reason for this could again be the need to estimate the ideal ranking, but in reality we can also see that λ scores are generally overestimated. Figure 8.4-left plots the estimated scores versus the actual scores, clearly showing that effectiveness is generally overestimated, especially in $nDCG_l@5$ and $RBP_l@5$. This behavior is similarly observed in probabilistic evaluation in Text IR, such as [Yilmaz and Aslam 2006, Aslam et al. 2006, Aslam and Yilmaz 2007]. When estimating the differences between systems in Section 8.3, these overestimations tended to cancel each other, but when estimating absolute scores they do not.

This means that we need to modify the stopping condition to account for this overestimation of effectiveness. As a first approach, we could come up with a correction factor on the estimates themselves and keep using the condition based on their variance. But this would be problematic as judgments are made, because at some point we will not need any correction at all. We can not base our corrections on the estimates because we do not know how erroneous they are, so we have to correct variance. Figure 8.5 shows the actual rooted variance of the estimates when the absolute error is at a certain level. For instance, the

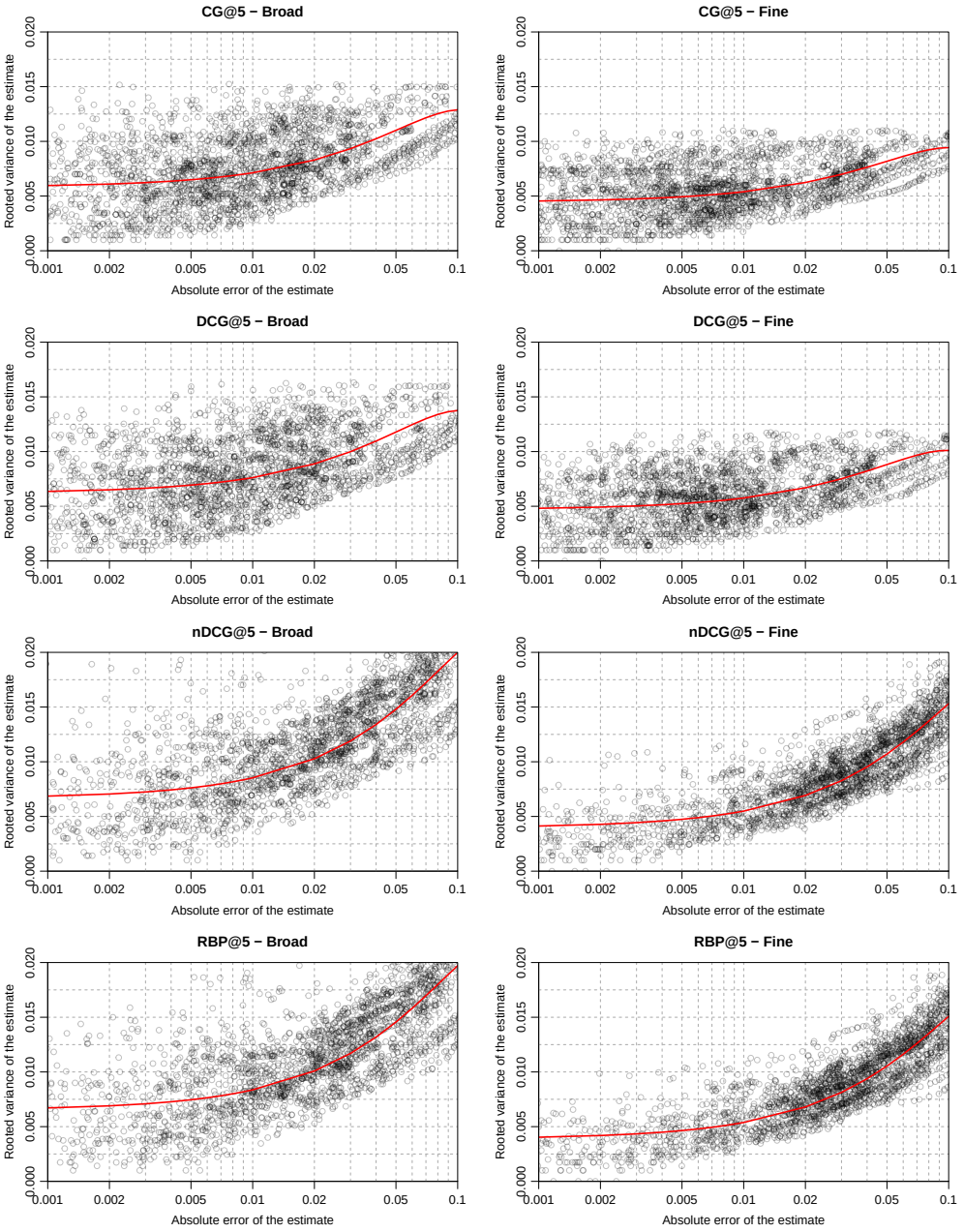

Figure 8.5: Rooted variance of estimates needed for absolute errors to be at a certain level.

top-left plot shows that when our goal is ± 0.01 we can use a threshold on variance of 0.0071^2 on average, or 0.011^2 if our goal is ± 0.05. These thresholds from Figure 8.5 can be used as a practical correction factor to use in the stopping condition. Table 8.6 shows the error of the estimates when judging documents until the *corrected* threshold on variance is reached and therefore the expected mean absolute error is ± 0.05 in practice. As shown, the error in $nDCG_l@5$ and $RBP_l@5$ does go down from 0.162 (see Table 8.5) to 0.049, but at the

CG_l@5								
Year	Broad				Fine			
	Judg.		MAE_λ	ρ	Judg.		MAE_λ	ρ
2007	720	(14.9%)	0.021	1	671	(13.9%)	0.021	0.993
2009	1960	(29.1%)	0.042	0.983	1660	(24.7%)	0.034	0.968
2010	382	(14%)	0.036	0.81	355	(13%)	0.032	0.857
2011	534	(8.5%)	0.02	0.967	539	(8.5%)	0.017	0.971

DCG_l@5								
Year	Broad				Fine			
	Judg.		MAE_λ	ρ	Judg.		MAE_λ	ρ
2007	696	(14.4%)	0.02	0.993	635	(13.2%)	0.021	0.993
2009	1956	(29.1%)	0.042	0.975	1592	(23.7%)	0.035	0.946
2010	393	(14.4%)	0.034	0.833	360	(13.2%)	0.031	0.976
2011	533	(8.4%)	0.022	0.975	521	(8.3%)	0.019	0.967

$nDCG_l$@5								
Year	Broad				Fine			
	Judg.		MAE_λ	ρ	Judg.		MAE_λ	ρ
2007	1643	(34%)	0.05	1	1730	(35.8%)	0.058	1
2009	3141	(46.7%)	0.056	0.964	3421	(50.8%)	0.049	0.975
2010	966	(35.3%)	0.056	0.905	1223	(44.7%)	0.044	0.881
2011	927	(14.7%)	0.031	0.983	1275	(20.2%)	0.046	0.971

RBP_l@5								
Year	Broad				Fine			
	Judg.		MAE_λ	ρ	Judg.		MAE_λ	ρ
2007	1615	(33.4%)	0.05	1	1660	(34.4%)	0.061	0.993
2009	3125	(46.4%)	0.056	0.964	3404	(50.6%)	0.05	0.979
2010	972	(35.5%)	0.056	0.952	1214	(44.4%)	0.045	0.929
2011	951	(15.1%)	0.033	0.977	1283	(20.3%)	0.046	0.975

Table 8.6: Accuracy of estimated absolute scores in MIREX 2007, 2009, 2010 and 2011 when judging documents until expected error is ± 0.05 with corrected thresholds on variance as per Figure 8.5.

expense of making several judgments. On average, both measures need about 35% of the documents to be judged.

Figure 8.4-right plots the estimated scores versus the actual scores when using the corrected thresholds in the stopping condition. Estimates are clearly more accurate with the corrected thresholds, though still slightly overestimated. This improvement is also reflected in the Spearman ρ correlation coefficients between estimated and actual scores. With uncorrected thresholds the average correlation was $\rho = 0.948$ (see Table 8.5), while with the corrected thresholds it goes up to $\rho = 0.966$ (see Table 8.6).

8.5 Discussion

Section 8.2 showed that systems can be ranked fairly well even in the absence of relevance judgments. Although confidence is a little overestimated, the overall accuracy of the ranking

is 0.92 on average and consistent across collections, effectiveness measures and relevance scales. Individual estimates can also be trusted, especially when confidence is over 99%. In those cases, which are the overwhelming majority, estimates were 99.5% accurate. This means that this evaluation method can be used by researchers to quickly check if one system is more effective than another. In general the estimate will be correct about 92% of the times, but if the confidence on the estimate is above 99%, which happens about two thirds of the times, then we can be almost certain that the estimate is indeed correct. In addition, we find that the few system pairs for which estimates are incorrect are not statistically significantly different anyway. If our confidence is not high enough, we can judge some documents to update estimates, and stop judging when the average confidence is above some threshold. Section 8.3 showed that very few judgments are required to reach an average confidence of 95%. On average, CG_l@5 required just 1.3% of judgments, while RBP_l@5 generally required the most, with an average of 9.7% of the total. Judging these few documents the expected accuracy of the ranking is 0.931.

When estimating absolute effectiveness scores the required number of judgments increases considerably, to an average of 25% when seeking errors of ± 0.05. Section 8.4 showed that confidence in absolute estimates is significantly overestimated, providing a correction factor to compute a more reliable stopping condition. In general, estimating $nDCG_l$@5 and RBP_l@5 requires more judging effort than CG_l@5 and DCG_l@5. This surely is because the latter only depend on the estimates for the retrieved documents, while the former also depend on the estimated ideal ranking. Moreover, expectation and variance are approximated with a Taylor series expansion in these measures, so accuracy and precision should be lower to begin with.

Both Section 8.3 and Section 8.4 defined a document weight to select which documents should be judged. In the first case, the weight was formulated to maximize the average confidence in the ranking of systems, while in the second case it was formulated to minimize the average variance of the estimates. It has to be noted that these weights, and the stopping conditions in general, can be changed to suit other needs. For instance, it can be desirable to obtain 95% accuracy not in the ranking, but in each of the individual estimated differences. Similarly, our confidence can be defined not in terms of the probability of a sign swap, but in terms of the probability of the difference being larger than some fixed threshold corresponding to a certain probability of user satisfaction from Chapter 3. Likewise, the confidence in the absolute scores can be defined in terms of the probability of system success from Chapter 4, fixing a minimum level of effectiveness in the same way.

8.6 Summary

This chapter presents probabilistic methods to evaluate systems with very few relevance judgments or with no judgments at all. Several effectiveness measures are defined probabilistically by means of random variables, and using the two models developed in Chapter 7 to estimate the relevance of documents, it is shown how to estimate both the ranking of systems and their absolute effectiveness scores. First, it is shown that even when there are no relevance judgments available the order of systems is correctly estimated 92% of the times. Second, it is shown that judging documents focused on estimating differences between sys-

tems, with as little as 2% of the judgments we can correctly estimate 93% of the differences. Finally, when focusing on estimating absolute scores, it is shown that with about 25% of the judgments we can estimate with an error of ± 0.05. Different measures are more or less efficient and accurate than others, but in all cases we see how to compute a stopping condition to decide when to stop judging or when to keep making relevance judgments.

These results clearly show that following a probabilistic approach to IR Evaluation can save many human resources in settings like the AMS evaluation in MIREX. Each edition of the AMS task requires the work of dozens of volunteers to perform relevance judgments, summing up to a total of several dozen hours of assessor time. In practice, though, collecting all these judgments takes several days, even weeks [Jones et al. 2007]. Using the methods showed here we would need judging effort orders of magnitude smaller, so all judgments can be made by a single human assessor in a matter of hours or just minutes; MIREX volunteers could work on other tasks for which building new collections is paramount. In addition, these techniques allow researchers to add relevance judgments when needed, increasing the accuracy and reliability of estimates, so that large test collections can be built over time and as needed.

Chapter 9
Conclusions and Future Work

Evaluation is a very important area of research in Information Retrieval that has received a lot of attention in recent years. However, it seems that the Music IR field has not been, until very recently, aware of the need to analyze the evaluation frameworks used. About a decade ago, the Music IR field adopted the knowledge body on Evaluation developed in Text IR throughout the years, but virtually no attention has been paid to whether that adoption is appropriate or not. Most importantly, all research on Text IR Evaluation has been practically ignored since then. These are very important issues because evaluation experiments are what drive research in one or another direction and therefore set the pace at which the whole field advances. If the conclusions drawn from evaluation experiments are wrong though, things might be happening the other way around.

This dissertation analyzes the validity, reliability and efficiency of IR evaluation experiments for the particular task of Audio Music Similarity, as performed in the annual MIREX evaluation campaign. The next sections outline the conclusions of this work and lines for future research.

9.1 Conclusions

9.1.1 Validity

Chapter 2 showed that the Cranfield paradigm evaluates characteristics of the systems and not of the users, but researchers make the assumption that these system characteristics correspond to different facets of the search process that users undergo. In particular, it is assumed that system effectiveness corresponds to user satisfaction. In Chapter 3, I provided an empirical mapping from a system effectiveness score onto the probability of user satisfaction. With this mapping researchers can now evaluate systems from the perspective of user satisfaction, which is the ultimate goal of evaluating AMS systems.

The effectiveness-satisfaction mapping clearly showed that users disagree to some extent as to what makes a particular result relevant for some query. About 20% of the users are not satisfied by systems that supposedly returned perfect results according to effectiveness, and likewise users that are satisfied despite the system was supposed to retrieve no relevant

material at all. This result establishes a practical upper (and lower) bound on the expected utility of systems evaluated with the current MIREX setting. Systems are not expected to obtain an average effectiveness score larger than 0.8 or so. These results also serve as a measurement of how much systems can improve if they could use information about the particular users they are targeted to. Targeting arbitrary users as of now, we expect a fraction of them to disagree with the evaluation results, but if those users were the same ones that make the relevance judgments, and we included personal information about them as part of the query, systems could exploit that information to try satisfying them completely.

It is also shown that quite large differences in effectiveness need to be observed between two systems for users to actually note it and choose the supposedly better system. In particular, differences larger than about 0.4 are needed for the majority of users to prefer the more effective system. Below that threshold, the majority of users just can not decide. This establishes a threshold on the practical significance of an observed difference in effectiveness. Even if a system is better according to a test collection, it is likely that the difference is not perceived by actual users; in fact, some users actually prefer the supposedly worse system. In addition, it should be noted that differences above 0.4 have been observed in MIREX only about 20% of the times since 2006, meaning that at least 80% of the system comparisons did not have practical significance.

Chapter 4 considered the effectiveness-satisfaction mapping over the sample of queries in a test collection, showing that reporting only averages, as usually found in the literature, can be misleading. In fact, opposite conclusions can be drawn between two systems when looking at the distribution of user satisfaction instead of the distribution of system effectiveness. At the very least, user satisfaction is usually smaller than intuitively indicated by effectiveness. Looking at the full distribution also allows us to evaluate systems from the perspective of a search success, defined as the situation in which the results from a system satisfy the majority of users. This is particularly interesting for systems that are already good for some types of queries but not for others, as opposed to systems that perform average in all cases. Once again, it is shown that results based on system success may easily contradict conclusions based solely on effectiveness. In fact, the probability of system success is usually underestimated by effectiveness. With the objective of accurately describing the actual distributions of scores, it was shown that for collections with at least a few dozen queries the Empirical distribution is the best choice. For small collections though, the Normal and Beta distributions are more appropriate.

9.1.2 Reliability

Chapter 2 showed that our estimates of system effectiveness are subject to random error due to sampling of queries, assessors, etc. Even if we conclude that some system is better than another according to a test collection, it can be the case that in reality it is the other way around. The usual indicator of confidence when drawing such conclusions is statistical significance; if a difference is found to be significant, we are confident that it is correct.

Different statistical significance tests make different assumptions about distributions and sampling methods, which are generally violated in IR evaluation experiments. To analyze how these tests behave for actual AMS evaluation data, Chapter 5 analyzed several tests in terms of their optimality under different evaluation scenarios. Results suggest that all

tests commit fewer Type I errors than expected, with the t-test and the Wilcoxon test being the safest of all. The bootstrap test is the most powerful test, and given that errors seem lower than expected, it is therefore suggested as the test of choice. In terms of exactness, the Wilcoxon and bootstrap test perform better than the others. The permutation test, which is theoretically supposed to be exact, is not shown to be optimal under any criterion, begging a deeper analysis of its assumptions in the case of IR Evaluation.

The results of Chapter 5 also allow researchers to interpret p-values from a practical point of view, though only for the particular case of AMS evaluation. It provides an indication of how likely it is for a result to hold with a different collection and, when it does not, what type of discrepancy is more likely. In this line of interpreting statistical significance, the chapter presents a discussion of the difference between practical and statistical significance, usually overlooked in fields like Information Retrieval that evolve in an iterative, almost mechanical manner.

Large test collections allow us to increase the reliability of our estimates, but they are also more expensive to build. Chapter 6 employs Generalizability Theory to investigate the optimal size of test collections for the evaluation of AMS systems. It is shown that MIREX collections are generally larger than necessary. In particular, the number of queries can be safely reduced from the usual 100 to less than 50 and still get reliable estimates of differences between systems. When our objective are absolute scores, the number of queries needs to be larger, but it can also be reduced to about 70 queries. It is also shown that for a fixed budget it is more reliable to increase the number of queries than the number of assessors per query and that, in any case, there is virtually no improvement in using more than two assessors. Similarly, it is shown that evaluating with a deeper cutoff, say $k = 10$, is slightly less reliable for a fixed budget, but still reliable enough to explore different user models.

In this chapter I also discuss the use of Generalizability Theory not only to assess how reliable an existing collection is, but to analyze a collection while it is built. This way, resources can be spared because the theory can tell us if the collection is sufficiently reliable in its current state and, if not, how many more queries or assessors are necessary.

9.1.3 Efficiency

Chapter 7 introduced the probabilistic framework for IR Evaluation, by which the effectiveness of systems according to a test collection is estimated with some degree of uncertainty, but with the advantage that the number of relevance judgments needed can be greatly reduced to save resources. The central point for probabilistic evaluation is that the relevance of a document for some query is estimated based on available information such as the output of systems, metadata, or known judgments. This chapter presents two models that estimate relevance under two different situations. The first model can be used even when there are no available judgments at all, so it can provide initial estimates of performance. The second model exploits information of relevance judgments when they become available, producing estimates much more accurate than the first model.

Based on this probabilistic definition of relevance, Chapter 8 shows how to estimate effectiveness scores for some measures, and how to select what documents to judge depending on what our purpose is. First, it is shown that the first model to estimate relevance can be used to reliably rank systems without relevance judgments; the estimated rankings are

correct in 92% of the system comparisons. Indeed, Chapter 7 showed that this model produces errors comparable to the errors expected just due to assessor disagreement, so these discrepancies with the actual ranking can be ignored for all practical purposes. In the second scenario it was shown how to select documents to minimize judging effort when our goal is to estimate system differences. Some measures are more efficient in this case, but with as little as 2% of the usual judgments it is possible to correctly estimate 93% of the differences on average, suggesting a tremendous save in human resources for the annual MIREX evaluations. For the third scenario the chapter describes how to select documents to minimize judging effort when our goal is to estimate absolute effectiveness scores. After correcting for overestimations of confidence, it is shown that with 25% of the usual judgments we can estimate absolute scores with an error of ± 0.05.

Throughout the dissertation, several effectiveness measures and relevance scales are compared in terms of validity, reliability and efficiency. Some measures and scales work better than others under different circumstances and with different objectives; the reader is referred to the *Discussion* section of each chapter for details. In general, it is shown that the Fine scale works better than the Broad scale, and that binary scales tend to perform worse. In terms of measures, $CG_l@k$, $DCG_l@k$ and $RBP_l@k$ perform better overall.

9.2 Future Work

Several lines for further work can be identified at this point. First, it is suggested to carry out user studies to better understand how users behave in the AMS task under different application scenarios such as music recommendation or playlist generation. It should be studied what the most appropriate user model is in these cases, whether evaluating only five documents is suitable or not, what the consequences are of using audio clips instead of full songs, what the effects on the user are of the diversity of genre and artist in the retrieved documents, etc. Most importantly, it should be studied how to incorporate dynamic and static user information into the queries so that systems, and also evaluations, can better try to adapt to particular users.

It is also proposed to study alternative forms of ground truth to better capture the utility of documents in AMS. The immediate question is whether absolute relevance judgments, as currently performed, are more appropriate than others such as preference judgments. Another point is the establishment of clear guidelines, maybe user-dependent, on what constitutes a relevant document for this task, pondering various aspects such as instrumentation or lyrics. In that line, other facets besides user satisfaction may be considered.

In terms of reliability, I did not consider corrections for multiple comparisons in the statistical significance tests. The significance level dictates the expected error rate in significance testing, but it is important to realize that these tests are performed independent of each other. At the usual $\alpha = 0.05$ significance level, it is expected that from 100 comparisons we incorrectly have 5 as significant, but the probability of at least one being actually incorrect is 99.4%. That is, almost certainly we are incorrectly achieving significance at least once. Similarly, there are different ways to account for multiple comparisons. Some methods are more powerful or conservative than others, and they make again different assumptions

about the distributions. In the current MIREX setting, the Friedman test is used along with Tukey's HSD procedure. It should be analyzed which multiple comparisons procedure is more appropriate for AMS data.

Another line for further work is the development of better models to predict relevance. Three sets of features were studied here, from which two different models were developed. But they are by no means the only possible models we can come up with. Although they worked reasonably well, they tended to overestimate relevance. I find it particularly interesting to study audio-related features, as well as contextual information such as tags from music-related social media or different statistics from music services. Also, given that very few judgments were needed in most cases, our estimates were actually based on a handful of known judgments. Different models should be studied to predict relevance in these cases, because our confidence should be lower than if we had hundreds or even thousands of them. Similarly, I only analyzed estimates for the top-5 documents; different models should be studied for an arbitrary evaluation depth.

Finally, I personally encourage similar research for other Music IR tasks. It should be particularly pursued the development of low-cost evaluation methods for other tasks in need of new or improved test collections.

Appendix A
Models to Estimate Relevance

For arbitrary scenarios of AMS evaluation, we can use the M_{out} and M_{jud} models fitted in Section 7.2 with all available data from MIREX 2007, 2009, 2010 and 2011. Table A.1 lists the fitted parameters, for both models and both similarity scales, for their use in future AMS evaluation experiments. As an example, let us use M_{out} to estimate the Broad score of a document whose true score is 2 and has the following feature vector:

$$\boldsymbol{\theta}_d = (fSYS = 0.25, OV = 0.8053, fART = 0.0217, sGEN = 1, fGEN = 0.8478)$$

Plugging these features and the parameters in Table A.1 into equation (7.3) we have:

$$\begin{aligned}
log\frac{P(R_d \geq 2|\boldsymbol{\theta}_d)}{P(R_d < 2|\boldsymbol{\theta}_d)} &= -3.5205- \\
&- 19.7968 \cdot 0.25 - 0.3227 \cdot 0.8053 + 29.6378 \cdot 0.25 \cdot 0.8053+ \\
&+ 3.2530 \cdot 0.0217 + 1.8975 \cdot 1+ \\
&+ 5.4055 \cdot 0.8478 - 2.9606 \cdot 1 \cdot 0.8478 = \\
&= 1.2781 \\
log\frac{P(R_d \geq 1|\boldsymbol{\theta}_d)}{P(R_d < 1|\boldsymbol{\theta}_d)} &= -1.4351- \\
&- 19.7968 \cdot 0.25 - 0.3227 \cdot 0.8053 + 29.6378 \cdot 0.25 \cdot 0.8053+ \\
&+ 3.2530 \cdot 0.0217 + 1.8975 \cdot 1+ \\
&+ 5.4055 \cdot 0.8478 - 2.9606 \cdot 1 \cdot 0.8478 = \\
&= 3.3635
\end{aligned}$$

Next, we use the inverse logit function:

$$\begin{aligned}
P(R_d \geq 2|\boldsymbol{\theta}_d) &= \frac{e^{1.2781}}{1 + e^{1.2781}} = 0.7821 \\
P(R_d \geq 1|\boldsymbol{\theta}_d) &= \frac{e^{3.3635}}{1 + e^{3.3635}} = 0.9665
\end{aligned}$$

Parameter	Broad		Fine	
	M_{out}	M_{jud}	M_{out}	M_{jud}
fSYS	-19.7968	0.9789	-17.4721	0.7954
OV	-0.3227	–	0.1336	–
fSYS:OV	29.6378	–	26.4550	–
fART	3.2530	–	2.9111	–
sGEN	1.8975	–	2.0443	–
fGEN	5.4055	–	5.4544	–
sGEN:fGEN	-2.9606	–	-3.4851	–
a*SYS*	–	1.1964	–	0.0128
a*ART*	–	7.1813	–	0.2078
α_1	-1.4351	-5.2165	-0.5092	-2.7554
α_2	-3.5205	-11.9251	-1.2231	-4.9168
α_3	–	–	-1.7919	-7.0128
α_4	–	–	-2.2787	-9.0010
α_5	–	–	-2.7216	-10.8548
α_6	–	–	-3.1956	-12.7158
α_7	–	–	-3.8044	-14.6722
α_8	–	–	-4.6928	-16.8831
α_9	–	–	-5.9567	-19.2536

Table A.1: Parameters fitted for the ordinal logistic regression models using all judgments from MIREX 2007, 2009, 2010 and 2011.

Plugging into equation (7.4):

$$
\begin{aligned}
P(R_d = 2|\boldsymbol{\theta}_d) &= 0.7821 \\
P(R_d = 1|\boldsymbol{\theta}_d) &= 0.9665 - 0.7821 = 0.1844 \\
P(R_d = 0|\boldsymbol{\theta}_d) &= 1 - 0.9665 = 0.0335
\end{aligned}
$$

Finally, plugging into equations (7.1) and (7.2) we can compute expectation and variance:

$$
\begin{aligned}
\mathrm{E}[R_d] &= 0.1844 + 0.7821 \cdot 2 = 1.7486 \\
\mathrm{Var}[R_d] &= 0.1844 + 0.7821 \cdot 2^2 - 1.6577^2 = 0.2552
\end{aligned}
$$

Bibliography

A. Al-Maskari, M. Sanderson, and P. Clough (2007). The Relationship between IR Effectiveness Measures and User Satisfaction. In *International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 773–774. Cited on pages 16, 26 and 27.

J. Allan, J. A. Aslam, B. Carterette, V. Pavlu, and E. Kanoulas (2008). Million Query Track 2008 Overview. In *Text REtrieval Conference*. Cited on page 88.

J. Allan, B. Carterette, J. A. Aslam, V. Pavlu, B. Dachev, and E. Kanoulas (2007). Million Query Track 2007 Overview. In *Text REtrieval Conference*. Cited on page 88.

J. Allan, B. Carterette, and J. Lewis (2005). When Will Information Retrieval Be 'Good Enough'? In *International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 433–440. Cited on pages 16, 30 and 32.

J. Allan and B. Croft (2003). Challenges in Information Retrieval and Language Modeling. *ACM SIGIR Forum*, 37(1):31–47. Cited on page 4.

J. Allan, B. Croft, A. Moffat, and M. Sanderson (2012). Frontiers, Challenges and Opportunities for Information Retrieval: Report from SWIRL 2012. *ACM SIGIR Forum*, 46(1):2–32. Cited on page 4.

O. Alonso and S. Mizzaro (2012). Using Crowdsourcing for TREC Relevance Assessment. *Information Processing and Management*, 48(6):1053–1066. Cited on page 18.

D. R. Anderson, K. P. Burnham, and W. L. Thompson (2000). Null Hypothesis Testing: Problems, Prevalence, and an Alternative. *Journal of Wildfire Management*, 64(4):912–923. Cited on page 79.

J. A. Aslam, V. Pavlu, and R. Savell (2003). A Unified Model for Metasearch, Pooling and System Evaluation. In *ACM International Conference on Information and Knowledge Management*, pages 484–491. Cited on page 18.

J. A. Aslam, V. Pavlu, and E. Yilmaz (2006). A Statistical Method for System Evaluation Using Incomplete Judgments. In *International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 541–548. Cited on page 123.

J. A. Aslam and E. Yilmaz (2007). Inferring Document Relevance from Incomplete Information. In *ACM International Conference on Information and Knowledge Management*, pages 633–642. Cited on pages 19, 101, 102 and 123.

J.-J. Aucouturier and F. Pachet (2002). Music Similarity Measures: What's the Use? In *International Conference on Music Information Retrieval*, pages 157–163. Cited on page 5.

R. Baeza-Yates and B. Ribeiro-Neto (2011). *Modern Information Retrieval: The Concepts and Technology behind Search*. Addison-Wesley. Cited on page 1.

P. Bailey, N. Craswell, I. Soboroff, P. Thomas, A. P. de Vries, and E. Yilmaz (2008). Relevance Assessment: Are Judges Exchangeable and Does it Matter? In *International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 667–674. Cited on pages 15 and 18.

R. J. Beckman and G. L. Tietjen (1978). Maximum Likelihood Estimation for the Beta Distribution. *Journal of Statistical Computation and Simulation*, 7(3-4):253–258. Cited on page 55.

P. N. Bennett, B. Carterette, O. Chapelle, and T. Joachims (2008). Beyond Binary Relevance: Preferences, Diversity and Set-Level Judgments. *ACM SIGIR Forum*, 42(2):53–58. Cited on page 17.

D. Bodoff (2008). Test Theory for Evaluating Reliability of IR Test Collections. *Information Processing and Management*, 44(3):1117–1145. Cited on pages 87 and 94.

D. Bodoff and P. Li (2007). Test Theory for Assessing IR Test Collections. In *International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 367–374. Cited on pages 17 and 86.

D. Bollen, B. P. Knijnenburg, M. C. Willemsen, and M. Graus (2010). Understanding Choice Overload in Recommender Systems. In *ACM Conference on Recommender Systems*, pages 63–70. Cited on page 98.

L. Boytsov, A. Belova, and P. Westfall (2013). Deciding on an Adjustment for Multiplicity in IR Experiments. In *International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 403–412. Cited on page 78.

R. L. Brennan (2001). *Generalizability Theory*. Springer. Cited on pages 85, 86, 87 and 95.

C. Buckley, D. Dimmick, I. Soboroff, and E. M. Voorhees (2007). Bias and the Limits of Pooling for Large Collections. *Journal of Information Retrieval*, 10(6):491–508. Cited on pages 15 and 18.

C. Buckley and E. M. Voorhees (2000). Evaluating Evaluation Measure Stability. In *International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 33–34. Cited on pages 17 and 86.

C. Buckley and E. M. Voorhees (2004). Retrieval Evaluation with Incomplete Information. In *International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 25–32. Cited on pages 15, 16 and 18.

S. Buettcher, G. V. Cormack, and C. L. Clarke (2010). *Information Retrieval: Implementing and Evaluating Search Engines*. The MIT Press. Cited on page 1.

C. Burges, T. Shaked, E. Renshaw, A. Lazier, M. Deeds, N. Hamilton, and G. Hullender (2005). Learning to Rank Using Gradient Descent. In *International Conference on Machine Learning*, pages 89–96. Cited on pages 21, 22, 23 and 27.

P. Cano, E. Gómez, F. Gouyon, P. Herrera, M. Koppenberger, B. Ong, X. Serra, S. Streich, and N. Wack (2006). ISMIR 2004 Audio Description Contest. Technical Report MTG-TR-2006-02, Universitat Pompeu Fabra. Cited on page 3.

B. Carterette (2007). Robust Test Collections for Retrieval Evaluation. In *International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 55–62. Cited on pages 19 and 102.

B. Carterette (2011). System Effectiveness, User Models, and User Utility: A General Framework for Investigation. In *International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 903–912. Cited on page 16.

B. Carterette (2012). Multiple Testing in Statistical Analysis of Systems-Based Information Retrieval Experiments. *ACM Transactions on Information Systems*, 30(1). Cited on pages 18 and 78.

B. Carterette and J. Allan (2007). Semiautomatic Evaluation of Retrieval Systems using Document Similarities. In *ACM International Conference on Information and Knowledge Management*, pages 873–876. Cited on page 19.

B. Carterette, J. Allan, and R. Sitaraman (2006). Minimal Test Collections for Retrieval Evaluation. In *International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 268–275. Cited on pages 19, 101, 102, 112 and 117.

B. Carterette, P. N. Bennett, D. M. Chickering, and S. T. Dumais (2008). Here or There: Preference Judgments for Relevance. In *European Conference on Information Retrieval*, pages 16–27. Cited on page 17.

B. Carterette, E. Gabrilovich, V. Josifovski, and D. Metzler (2010a). Measuring the Reusability of Test Collections. In *ACM International Conference on Web Search and Data Mining*, pages 231–240. Cited on page 15.

B. Carterette and R. Jones (2007). Evaluating Search Engines by Modeling the Relationship between Relevance and Clicks. In *Annual Conference on Neural Information Processing Systems*. Cited on page 103.

B. Carterette, E. Kanoulas, V. Pavlu, and H. Fang (2010b). Reusable Test Collections Through Experimental Design. In *International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 547–554. Cited on page 15.

B. Carterette, V. Pavlu, E. Kanoulas, J. A. Aslam, and J. Allan (2009). If I Had a Million Queries. In *European Conference on Information Retrieval*, pages 288–300. Cited on page 19.

B. Carterette and M. D. Smucker (2007). Hypothesis Testing with Incomplete Relevance Judgments. In *ACM International Conference on Information and Knowledge Management*, pages 643–652. Cited on page 18.

B. Carterette and I. Soboroff (2010). The Effect of Assessor Error on IR System Evaluation. In *International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 539–546. Cited on page 15.

G. Casella and R. L. Berger (2002). *Statistical Inference*. Thomson Learning. Cited on page 110.

O. Chapelle, D. Metzler, Y. Zhang, and P. Grinspan (2009). Expected Reciprocal Rank for Graded Relevance. In *ACM International Conference on Information and Knowledge Management*, pages 621–630. Cited on pages 17 and 23.

C. W. Cleverdon (1991). The Significance of the Cranfield Tests on Index Languages. In *International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 3–12. Cited on pages 2, 9 and 20.

J. Cohen (1994). The Earth is Round (p¡.05). *American Psychologist*, 49(12):997–1003. Cited on page 80.

W. Conover (1999). *Practical Nonparametric Statistics*. Wiley. Cited on page 66.

G. V. Cormack and T. R. Lynam (2006). Statistical Precision of Information Retrieval Evaluation. In *International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 533–540. Cited on pages 14 and 15.

G. V. Cormack and T. R. Lynam (2007). Validity and Power of t-test for Comparing MAP and GMAP. In *International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 753–754. Cited on pages 77 and 78.

G. V. Cormack, C. R. Palmer, and C. L. Clarke (1998). Efficient Construction of Large Test Collections. In *International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 282–289. Cited on page 18.

H. Cramér (1928). On the Composition of Elementary Errors II. *Scandinavian Actuarial Journal*, 11(1):141–180. Cited on page 56.

B. Croft, D. Metzler, and T. Strohman (2009). *Search Engines: Information Retrieval in Practice*. Cited on page 1.

S. J. Cunningham, D. Bainbridge, and J. S. Downie (2012). The Impact of MIREX on Scholarly Research (2005-2010). In *International Society for Music Information Retrieval Conference*, pages 259–264. Cited on page 6.

S. Doraisamy and S. Rüger (2003). Robust Polyphonic Music Retrieval with N-grams. *Journal of Intelligent Systems*, 21(1):53–70. Cited on page 5.

J. S. Downie (2002). Interim Report on Establishing MIR/MDL Evaluation Frameworks: Commentary on Consensus Building. In *ISMIR Panel on Music Information Retrieval Evaluation Frameworks*, pages 43–44. Cited on page 3.

J. S. Downie (2003a). Music Information Retrieval. *Annual Review of Information Science and Technology*, 37(1):295–340. Cited on page 5.

J. S. Downie (2003b). *The MIR/MDL Evaluation Project White Paper Collection*. 3rd edition. Cited on page 3.

J. S. Downie (2004). The Scientific Evaluation of Music Information Retrieval Systems: Foundations and Future. *Computer Music Journal*, 28(2):12–23. Cited on page 3.

J. S. Downie, A. F. Ehmann, M. Bay, and M. C. Jones (2010). The Music Information Retrieval Evaluation eXchange: Some Observations and Insights. In W. R. Zbigniew and A. A. Wieczorkowska, editors, *Advances in Music Information Retrieval*, pages 93–115. Springer. Cited on pages 4, 25, 29, 90 and 104.

B. Efron and R. J. Tibshirani (1998). *An Introduction to the Bootstrap*. Chapman & Hall/CRC. Cited on page 66.

R. A. Fisher (1925). *Statistical Methods for Research Workers*. Cosmo Publications. Cited on page 82.

A. Flexer (2006). Statistical Evaluation of Music Information Retrieval Experiments. *Journal of New Music Research*, 35(2):113–120. Cited on page 76.

A. Flexer and D. Schnitzer (2010). Effects of Album and Artist Filters in Audio Similarity Computed for Verly large Music Databases. *Computer Music Journal*, 34(3):20–28. Cited on pages 104 and 106.

A. Gelman and H. Stern (2006). The Difference Between "Significant" and "Not Significant" is not Itself Statistically Significant. *The American Statistician*, 60(4):328–331. Cited on page 80.

S. Geman, E. Bienenstock, and R. Doursat (1992). Neural Networks and the Bias/Variance Dilema. *Neural Computation*, 4(1):1–58. Cited on page 13.

P. Good (2005). *Permutation, Parametric, and Bootstrap Tests of Hypotheses*. Springer. Cited on page 66.

J. Guiver, S. Mizzaro, and S. Robertson (2009). A Few Good Topics: Experiments in Topic Set Reduction for Retrieval Evaluation. *ACM Transactions on Information Systems*, 27(4):1–26. Cited on page 16.

C. Gull (1956). Seven Years of Work on the Organisation of Materials in a Special Library. *American Documentation*, 7(4):320–329. Cited on page 11.

D. K. Harman (1993). Overview of the Second Text REtrieval Conference (TREC-2). In *Text REtrieval Conference*. Cited on page 19.

D. K. Harman (2011). Information Retrieval Evaluation. *Synthesis Lectures on Information Concepts, Retrieval, and Services*, 3(2):1–119. Cited on pages 2, 6 and 9.

M. A. Hearst and J. O. Pedersen (1996). Reexamining the Cluster Hypothesis: Scatter/Gather on Retrieval Results. In *International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 76–84. Cited on page 105.

P. Herrera and F. Gouyon (2013). MIRrors: Music Information Research Reflects on its Future. *Journal of Intelligent Information Systems*. Cited on page 4.

W. Hersh, A. Turpin, S. Price, B. Chan, D. Kraemer, L. Sacherek, and D. Olson (2000). Do Batch and User Evaluations Give the Same Results? In *International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 17–24. Cited on page 16.

X. Hu and N. Kando (2012). User-Centered Measures vs. System Effectiveness in Finding Similar Songs. In *International Society for Music Information Retrieval Conference*, pages 331–336. Cited on page 16.

S. B. Huffman and M. Hochster (2007). How Well does Result Relevance Predict Session Satisfaction? In *International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 567–573. Cited on page 16.

D. Hull (1993). Using Statistical Testing in the Evaluation of Retrieval Experiments. In *International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 329–338. Cited on page 76.

R. Hyndman and Y. Fan (1996). Sample Quantiles in Statistical Packages. *American Statistician*, 50(4):361–365. Cited on page 53.

J. P. A. Ioannidis (2005). Why Most Published Research Findings Are False. *PLoS Medicine*, 2(8). Cited on pages 18 and 79.

P. G. Ipeirotis, F. Provost, and J. Wang (2010). Quality Management on Amazon Mechanical Turk. In *ACM SIGKDD Workshop on Human Computation*, pages 64–67. Cited on page 18.

K. Järvelin (2011). IR Research: Systems, Interaction, Evaluation and Theories. *ACM SIGIR Forum*, 45(2):17–31. Cited on pages 14 and 46.

K. Järvelin and J. Kekäläinen (2000). IR Evaluation Methods for Retrieving Highly Relevant Documents. In *International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 41–48. Cited on page 20.

K. Järvelin and J. Kekäläinen (2002). Cumulated Gain-Based Evaluation of IR Techniques. *ACM Transactions on Information Systems*, 20(4):422–446. Cited on pages 17, 21 and 22.

D. H. Johnson (1999). The Insignificance of Statistical Significance Testing. *Journal of Wildfire Management*, 63(3):763–772. Cited on page 80.

M. C. Jones, J. S. Downie, and A. F. Ehmann (2007). Human Similarity Judgments: Implications for the Design of Formal Evaluations. In *International Conference on Music Information Retrieval*, pages 539–542. Cited on pages 25, 45, 46 and 127.

E. Kanoulas and J. A. Aslam (2009). Empirical Justification of the Gain and Discount Function for nDCG. In *ACM International Conference on Information and Knowledge Management*, pages 611–620. Cited on pages 17 and 68.

P. Kantor and E. M. Voorhees (1996). The TREC-5 Confusion Track. In *Text REtrieval Conference*. Cited on page 20.

J. Kekäläinen (2005). Binary and Graded Relevance in IR Evaluations: Comparison of the Effects on Ranking of IR Systems. *Information Processing and Management*, 41(5):1019–1033. Cited on pages 14 and 17.

A. Kittur, E. H. Chi, and B. Suh (2008). Crowdsourcing User Studies With Mechanical Turk. In *Annual ACM SIGCHI Conference on Human Factors in Computing Systems*, pages 453–456. Cited on page 18.

A. Kittur, J. V. Nickerson, M. S. Bernstein, E. Gerber, A. D. Shaw, J. Zimmerman, M. Lease, and J. Horton (2013). The Future of Crowd Work. In *ACM Conference on Computer Supported Cooperative Work and Social Computing*, pages 1301–1318. Cited on page 18.

A. Kolmogorov (1933). Sulla Determinazione Empirica di una Legge di Distribuzione. *Giornale dell'Istituto Italiano degli Attuari*, 4:83–91. Cited on page 55.

F. Lancaster (1968). Evaluation of the MEDLARS Demand Search Service. Technical report, U.S. Department of Health, Education, and Welfare. Cited on page 2.

O. Lartillot, R. Miotto, N. Montecchio, N. Orio, D. Rizo, and M. Schedl (2011). MusiClef: A Benchmark Activity in Multimodal Music Information Retrieval. In *International Society for Music Information Retrieval Conference*. Cited on page 4.

J. Le, A. Edmonds, V. Hester, and L. Biewald (2010). Ensuring Quality in Crowdsourced Search Relevance Evaluation: The Effects of Training Question Distribution. In *ACM SIGIR Workshop on Crowdsourcing for Search Evaluation*, pages 17–20. Cited on page 33.

M. Lease and E. Yilmaz (2011). Crowdsourcing for Information Retrieval. *ACM SIGIR Forum*, 45(2):66–75. Cited on page 18.

J. H. Lee (2010). Crowdsourcing Music Similarity Judgments using Mechanical Turk. In *International Society for Music Information Retrieval Conference*, pages 183–188. Cited on pages 32 and 33.

E. Lehmann and G. Casella (1998). *Theory of Point Estimation*. Springer. Cited on page 13.

M. Lesk, D. K. Harman, E. A. Fox, H. Wu, and C. Buckley (1997). The SMART Lab Report. *ACM SIGIR Forum*, 31(1):2–22. Cited on page 2.

S. Lippens, J. Martens, M. Leman, B. Baets, H. Meyer, and G. Tzanetakis (2004). A Comparison of Human and Automatic Musical Genre Classification. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 233–236. Cited on pages 46 and 108.

I. Liu and A. Agresti (2005). The Analysis of Ordered Categorical Data: An Overview and a Survey of Recent Developments. *Sociedad Estadística e Investigación Operativa Test*, 14(1):1–73. Cited on page 103.

B. Logan and A. Salomon (2001). A Music Similarity Function Based on Signal Analysis. In *IEEE International Conference on Multimedia and Expo*, page 190. Cited on page 5.

J. S. Long (1997). *Regression Models for Categorical and Limited Dependent Variables*. Sage Publications, 1st edition. Cited on page 103.

M. I. Mandel and D. P. Ellis (2005). Song-Level Features And Support Vector Machines For Music Classification. In *International Conference on Music Information Retrieval*, pages 594–599. Cited on page 105.

C. D. Manning, P. Raghavan, and H. Schütze (2008). *Introduction to Information Retrieval*. Cambridge University Press. Cited on page 1.

W. Mason and D. J. Watts (2009). Financial Incentives and the "Performance of Crowds". In *ACM SIGKDD Workshop on Human Computation*, pages 77–85. Cited on page 33.

B. Mcfee, L. Barrington, and G. Lanckriet (2012). Learning Content Similarity for Music Recommendation. *IEEE Transactions on Audio, Speech and Language Processing*, 20(8):2207–2218. Cited on page 5.

B. McFee, T. Bertin-Mahieux, D. P. Ellis, and G. Lanckriet (2012). The Million Song Dataset Challenge. In *WWW International Workshop on Advances in Music Information Research*, pages 909–916. Cited on pages 4 and 46.

A. Moffat, W. Webber, and J. Zobel (2007). Strategic System Comparisons via Targeted Relevance Judgments. In *International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 375–382. Cited on page 18.

A. Moffat and J. Zobel (2008). Rank-Biased Precision for Measurement of Retrieval Effectiveness. *ACM Transactions on Information Systems*, 27(1). Cited on pages 17, 22 and 27.

A. Moffat, J. Zobel, and D. Hawking (2005). Recommended Reading for IR Research Students. *ACM SIGIR Forum*, 39(2):3–14. Cited on page 4.

G. Peeters, J. Urbano, and G. J. Jones (2012). Notes from the ISMIR 2012 Late-Breaking Session on Evaluation in Music Information Retrieval. In *International Society for Music Information Retrieval Conference*. Cited on pages 2, 4 and 6.

T. Pohle (2010). *Automatic Characterization of Music for Intuitive Retrieval*. PhD thesis, Johannes Kepler University. Cited on pages 104 and 106.

T. Poibeau and L. Kosseim (2001). Proper Name Extraction from Non-Journalistic Texts. *Language and Computers - Studies in Practical Linguistics*, 37:144–157. Cited on page 16.

S. Robertson (2008). On the History of Evaluation in IR. *Journal of Information Science*, 34(4):439–456. Cited on page 2.

S. Robertson (2011). On the Contributions of Topics to System Evaluation. In *European Conference on Information Retrieval*, pages 129–140. Cited on page 16.

S. Robertson and E. Kanoulas (2012). On Per-Topic Variance in IR Evaluation. In *International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 891–900. Cited on page 15.

S. Robertson, E. Kanoulas, and E. Yilmaz (2010). Extending Average Precision to Graded Relevance Judgments. In *International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 603–610. Cited on pages 17, 23, 24 and 28.

J. Rzeszotarski and A. Kittur (2011). Instrumenting the Crowd: Using Implicit Behavioral Measures to Predict Task Performance. In *ACM Symposium on User Interface Software and Technology*. Cited on page 18.

T. Sakai (2004). New Performance Metrics Based on Multigrade Relevance: Their Application to Question Answering. In *NTCIR Workshop on Research in Information Access Technologies, Information Retrieval, Question Answering and Summarization*. Cited on pages 17 and 22.

T. Sakai (2006). Evaluating Evaluation Metrics Based on the Bootstrap. In *International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 525–532. Cited on pages 18 and 77.

T. Sakai (2007). On the Reliability of Information Retrieval Metrics Based on Graded Relevance. *Information Processing and Management*, 43(2):531–548. Cited on pages 14, 17 and 86.

T. Sakai and N. Kando (2008). On Information Retrieval Metrics Designed for Evaluation with Incomplete Relevance Assessments. *Journal of Information Retrieval*, 11(5):447–470. Cited on pages 15 and 18.

J. Salamon and J. Urbano (2012). Current Challenges in the Evaluation of Predominant Melody Extraction Algorithms. In *International Society for Music Information Retrieval Conference*, pages 289–294. Cited on page 17.

M. Sanderson (2010). Test Collection Based Evaluation of Information Retrieval Systems. *Foundations and Trends in Information Retrieval*, 4(4):247–375. Cited on pages 6 and 9.

M. Sanderson and B. Croft (2012). The History of Information Retrieval Research. *Proceedings of the IEEE*, 100:1444–1451. Cited on page 1.

M. Sanderson, M. L. Paramita, P. Clough, and E. Kanoulas (2010). Do User Preferences and Evaluation Measures Line Up? In *International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 555–562. Cited on pages 16, 17, 18, 30, 32 and 33.

M. Sanderson, A. Turpin, Y. Zhang, and F. Scholer (2012). Differences in Effectiveness Across Sub-collections. In *ACM International Conference on Information and Knowledge Management*, pages 1965–1969. Cited on page 15.

M. Sanderson and J. Zobel (2005). Information Retrieval System Evaluation: Effort, Sensitivity, and Reliability. In *International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 162–169. Cited on pages 17, 77 and 86.

T. Saracevic (1995). Evaluation of Evaluation in Information Retrieval. In *International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 138–146. Cited on page 6.

J. Savoy (1997). Statistical Inference in Retrieval Effectiveness Evaluation. *Information Processing and Management*, 33(4):495–512. Cited on page 76.

N. Scaringella, G. Zoia, and D. Mlynek (2006). Automatic Genre Classification of Music Content: a Survey. *IEEE Signal Processing Magazine*, 23(2):133–141. Cited on page 108.

L. Schamber (1994). Relevance and Information Behavior. *Annual Review of Information Science and Technology*, 29:3–48. Cited on page 14.

M. Schedl, A. Flexer, and J. Urbano (2013a). The Neglected User in Music Information Retrieval Research. *Journal of Intelligent Information Systems*. Cited on pages 14, 46 and 96.

M. Schedl, D. Hauger, and J. Urbano (2013b). Harvesting Microblogs for Contextual Music Similarity Estimation: A Co-occurrence-based Framework. *Journal of Multimedia Systems*. Cited on page 106.

F. Scholer and A. Turpin (2008). Relevance Thresholds in System Evaluations. In *International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 693–694. Cited on page 16.

M. A. Seaman, J. R. Levin, and R. C. Serlin (1991). New Developments in Pairwise Multiple Comparisons: Some Powerful and Practicable Procedures. *Psychological Bulletin*, 110(3):577–586. Cited on page 78.

X. Serra, M. Magas, E. Benetos, M. Chudy, S. Dixon, A. Flexer, E. Gómez, F. Gouyon, P. Herrera, S. Jorda, O. Paytuvi, G. Peeters, J. Schlüter, H. Vinet, and G. Widmer (2013). Roadmap for Music Information ReSearch. Technical report, MIReS Consortium. Cited on page 4.

K. Seyerlehner, G. Widmer, and P. Knees (2010a). A Comparison of Human, Automatic and Collaborative Music Genre Classification and User Centric Evaluation of Genre Classification Systems. In *International Workshop on Adaptive Multimedia Retrieval*, pages 17–18. Cited on pages 46 and 108.

K. Seyerlehner, G. Widmer, and T. Pohle (2010b). Fusing Block-level Features for Music Similarity Estimation. In *International Conference on Digital Audio Effects*. Cited on page 5.

W. R. Shadish, T. D. Cook, and D. T. Campbell (2002). *Experimental and Quasi-Experimental Designs for Generalized Causal Inference*. Houghton-Mifflin. Cited on page 13.

R. J. Shavelson and N. M. Webb (1991). *Generalizability Theory: A Primer*. Sage Publications. Cited on pages 85 and 87.

M. D. Smucker, J. Allan, and B. Carterette (2007). A Comparison of Statistical Significance Tests for Information Retrieval Evaluation. In *ACM International Conference on Information and Knowledge Management*, pages 623–632. Cited on pages 18, 65, 67, 70 and 77.

M. D. Smucker, J. Allan, and B. Carterette (2009). Agreement Among Statistical Significance Tests for Information Retrieval Evaluation at Varying Sample Sizes. In *International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 630–631. Cited on pages 70 and 77.

M. D. Smucker and C. L. Clarke (2012a). The Fault, Dear Researchers, is Not in Cranfield, But in Our Metrics, that They Are Unrealistic. In *European Workshop on Human-Computer Interaction and Information Retrieval*, pages 11–12. Cited on page 16.

M. D. Smucker and C. L. Clarke (2012b). Time-Based Calibration of Effectiveness Measures. In *International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 95–104. Cited on page 16.

R. Snow, B. O'Connor, D. Jurafsky, and A. Y. Ng (2008). Cheap and Fast—But is it Good? Evaluating Non-Expert Annotations for Natural Language Tasks. In *Conference on Empirical Methods in Natural Language Processing*, pages 254–263. Cited on page 18.

I. Soboroff, C. Nicholas, and P. Cahan (2001). Ranking Retrieval Systems Without Relevance Judgments. In *International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 66–73. Cited on pages 18 and 114.

Student (1908). The Probable Error of a Mean. *Biometrika*, 6(1):1–25. Cited on page 65.

J. Tague-Sutcliffe (1992). The Pragmatics of Information Retrieval Experimentation, Revisited. *Information Processing and Management*, 28(4):467–490. Cited on pages 6, 12, 17 and 18.

J. R. Taylor (1997). *An Introduction Error Analysis: The Study of Uncertainties in Physical Measurements*. University Science Books. Cited on page 13.

W. M. Trochim and J. P. Donnelly (2007). *The Research Methods Knowledge Base*. Atomic Dog Publishing, 3rd edition. Cited on pages iii, 12, 13, 17 and 18.

A. Turpin and W. Hersh (2001). Why Batch and User Evaluations Do Not Give the Same Results. In *International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 225–231. Cited on page 16.

R. Typke, M. den Hoed, J. de Nooijer, F. Wiering, and R. C. Veltkamp (2005a). A Ground Truth for Half a Million Musical Incipits. *Journal of Digital Information Management*, 3(1):34–39. Cited on page 24.

R. Typke, R. C. Veltkamp, and F. Wiering (2006). A Measure for Evaluating Retrieval Techniques based on Partially Ordered Ground Truth Lists. In *IEEE International Conference on Multimedia and Expo*, pages 1793–1796. Cited on page 24.

R. Typke, F. Wiering, and R. C. Veltkamp (2005b). A Survey of Music Information Retrieval Systems. In *International Conference on Music Information Retrieval*, pages 153–160. Cited on page 5.

J. Urbano (2011). Information Retrieval Meta-Evaluation: Challenges and Opportunities in the Music Domain. In *International Society for Music Information Retrieval Conference*, pages 609–614. Cited on page 6.

J. Urbano, J. S. Downie, B. Mcfee, and M. Schedl (2012). How Significant is Statistically Significant? The case of Audio Music Similarity and Retrieval. In *International Society for Music Information Retrieval Conference*, pages 181–186. Cited on page 32.

J. Urbano, J. Lloréns, J. Morato, and S. Sánchez-Cuadrado (2011a). Melodic Similarity through Shape Similarity. In S. Ystad, M. Aramaki, R. Kronland-Martinet, and K. Jensen, editors, *Exploring Music Contents*, pages 338–355. Springer. Cited on page 5.

J. Urbano, M. Marrero, and D. Martín (2013a). A Comparison of the Optimality of Statistical Significance Tests for Information Retrieval Evaluation. In *International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 925–928. Cited on pages 18 and 77.

J. Urbano, M. Marrero, and D. Martín (2013b). On the Measurement of Test Collection Reliability. In *International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 393–402. Cited on pages 17, 87, 88 and 98.

J. Urbano, M. Marrero, D. Martín, and J. Lloréns (2010a). Improving the Generation of Ground Truths based on Partially Ordered Lists. In *International Society for Music Information Retrieval Conference*, pages 285–290. Cited on page 24.

J. Urbano, M. Marrero, D. Martín, J. Morato, K. Robles, and J. Lloréns (2011b). The University Carlos III of Madrid at TREC 2011 Crowdsourcing Track. In *Text REtrieval Conference*. Cited on pages 18 and 33.

J. Urbano, D. Martín, M. Marrero, and J. Morato (2011c). Audio Music Similarity and Retrieval: Evaluation Power and Stability. In *International Society for Music Information Retrieval Conference*, pages 597–602. Cited on page 78.

J. Urbano, J. Morato, M. Marrero, and D. Martín (2010b). Crowdsourcing Preference Judgments for Evaluation of Music Similarity Tasks. In *ACM SIGIR Workshop on Crowdsourcing for Search Evaluation*, pages 9–16. Cited on pages 32 and 33.

J. Urbano and M. Schedl (2012). Towards Minimal Test Collections for Evaluation of Audio Music Similarity and Retrieval. In *WWW International Workshop on Advances in Music Information Research*, pages 917–923. Cited on page 102.

J. Urbano and M. Schedl (2013). Minimal Test Collections for Low-Cost Evaluation of Audio Music Similarity and Retrieval Systems. *International Journal of Multimedia Information Retrieval*, 2(1):59–70. Cited on page 102.

J. Urbano, M. Schedl, and X. Serra (2013c). Evaluation in Music Information Retrieval. *Journal of Intelligent Information Systems*. Cited on page 6.

R. von Mises (1931). *Wahrscheinlichkeitsrechnung und ihre Anwendungen in der Statistik und theoretischen Physik*. Cited on page 56.

E. M. Voorhees (1998). Variations in Relevance Judgments and the Measurement of Retrieval Effectiveness. In *International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 315–323. Cited on page 86.

E. M. Voorhees (2000). Variations in Relevance Judgments and the Measurement of Retrieval Effectiveness. *Information Processing and Management*, 36(5):697–716. Cited on pages 14 and 15.

E. M. Voorhees (2001). Evaluation by Highly Relevant Documents. In *International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 74–82. Cited on pages 17 and 20.

E. M. Voorhees (2002a). The Philosophy of Information Retrieval Evaluation. In *Workshop of the Cross-Language Evaluation Forum*, pages 355–370. Cited on pages 2, 6, 9, 10, 15 and 16.

E. M. Voorhees (2002b). Whither Music IR Evaluation Infrastructure: Lessons to be Learned from TREC. In *JCDL Workshop on the Creation of Standardized Test Collections, Tasks, and Metrics for Music Information Retrieval (MIR) and Music Digital Library (MDL) Evaluation*, pages 7–13. Cited on page 3.

E. M. Voorhees (2009). Topic Set Size Redux. In *International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 806–807. Cited on pages 77 and 86.

E. M. Voorhees and C. Buckley (2002). The Effect of Topic Set Size on Retrieval Experiment Error. In *International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 316–323. Cited on pages 17 and 86.

E. M. Voorhees and D. K. Harman (2005). *TREC: Experiment and Evaluation in Information Retrieval*. MIT Press. Cited on pages 3 and 16.

W. Webber, A. Moffat, and J. Zobel (2008a). Statistical Power in Retrieval Experimentation. In *ACM International Conference on Information and Knowledge Management*, pages 571–580. Cited on page 18.

W. Webber, A. Moffat, J. Zobel, and T. Sakai (2008b). Precision-At-Ten Considered Redundant. In *International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 695–696. Cited on page 14.

F. Wilcoxon (1945). Individual Comparisons by Ranking Methods. *Biometrics Bulletin*, 1(6):80–83. Cited on page 65.

T. Yee (2010). The VGAM Package for Categorical Data Analysis. *Journal of Statistical Software*, 32(10):1–34. Cited on page 103.

T. Yee and C. Wild (1996). Vector Generalized Additive Models. *Journal of the Royal Statistical Society*, 58(3):481–493. Cited on page 103.

E. Yilmaz and J. A. Aslam (2006). Estimating Average Precision with Incomplete and Imperfect Information. In *ACM International Conference on Information and Knowledge Management*, pages 102–111. Cited on pages 19 and 123.

E. Yilmaz, E. Kanoulas, and J. A. Aslam (2008). A Simple and Efficient Sampling Method for Estimating AP and NDCG. In *International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 603–610. Cited on page 19.

E. Yilmaz, M. Shokouhi, N. Craswell, and S. Robertson (2010). Expected Browsing Utility for Web Search Evaluation. In *ACM International Conference on Information and Knowledge Management*, pages 1561–1564. Cited on page 16.

H. Zaragoza, B. B. Cambazoglu, and R. Baeza-Yates (2010). Web Search Solved? All Result Rankings the Same? In *ACM International Conference on Information and Knowledge Management*, pages 529–538. Cited on page 16.

S. T. Ziliak and D. N. McCloskey (2008). *The Cult of Statistical Significance: How the Standard Error Costs Us Jobs, Justice, and Lives*. University of Michigan Press. Cited on pages 18 and 79.

J. Zobel (1998). How Reliable are the Results of Large-Scale Information Retrieval Experiments? In *International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 307–314. Cited on pages 15, 16, 18, 77 and 86.