Università degli Studi di Udine
Dipartimento di Scienze Matematiche, Informatiche e Fisiche
Dottorato di Ricerca in Informatica

Ph.D. Thesis

Crowdsourcing Relevance: Two Studies on Assessment

Candidate
Eddy Maddalena

Supervisor
Prof. Stefano Mizzaro

Cycle XXIX — A.Y. 2015-2016

Institute Contacts
Dipartimento di Scienze Matematiche, Informatiche e Fisiche
Università degli Studi di Udine
Via delle Scienze, 206
33100 Udine — Italia
+39 0432 558400
http://www.dimi.uniud.it/

This thesis is dedicated to all of those who contributed:
a few passionate researchers, and a few tens of thousands
unaware crowdsourcing workers spread around the world.

Acknowledgements

I want to thank all the colleagues and staff of the University of Udine (Italy) for giving me the opportunity to do my PhD. In particular, I would like to thank my supervisor Prof. Stefano Mizzaro who has always supported and encouraged me with remarkable patience. I owe him a huge gratitude for conveying to me such passion for my work.

Furthermore, a special thanks to all my co-authors, since working with them I have learned a lot of precious notions, and from them I could appreciate and catch some aspects, which allowed me to refine my method of work.

I would like to thank Prof. Mark Sanderson and Dr. Gianluca Demartini for inviting me to spend respectively six months in the RMIT (Royal Melbourne Institute of Technology) University in Melbourne (Australia), and four months in the Information School of the University of Sheffield (United Kingdom). Also, I want to thank all the colleagues and staff of these universities, for the fruitful collaboration and the pleasant time spent together both in work and spare time.

Additionally, I would like to thank the SEEK company, based in Melbourne (Australia) which allowed me to complete an internship, and in particular, Dr. Wilson Wong who worked with me on the project.

Last but not least, I would like to thank my family, my girlfriend and my friends for the daily support, the continuous encouragement and the patience demonstrated throughout the time spent listening to my paranoid complaints.

Abstract

Crowdsourcing has become an alternative approach to collect relevance judgments at large scale. In this thesis, we focus on some specific aspects related to time, scale, and agreement.

First, we address the issue of the time factor in gathering relevance label: we study how much time the judges need to assess documents. We conduct a series of four experiments which unexpectedly reveal us how introducing time limitations leads to benefits in terms of the quality of the results. Furthermore, we discuss strategies aimed to determine the right amount of time to make available to the workers for the relevance assessment, in order to both guarantee the high quality of the gathered results and the saving of the valuable resources of time and money.

Then we explore the application of magnitude estimation, a psychophysical scaling technique for the measurement of sensation, for relevance assessment. We conduct a large-scale user study across 18 TREC topics, collecting more than 50,000 magnitude estimation judgments, which result to be overall rank-aligned with ordinal judgments made by expert relevance assessors. We discuss the benefits, the reliability of the judgements collected, and the competitiveness in terms of assessor cost.

We also report some preliminary results on the agreement among judges. Often, the results of crowdsourcing experiments are affected by noise, that can be ascribed to lack of agreement among workers. This aspect should be considered as it can affect the reliability of the gathered relevance labels, as well as the overall repeatability of the experiments.

Credits

Part of this thesis is based on the following publications:

- Eddy Maddalena, Marco Basaldella, Dario De Nart, Dante Degl'Innocenti, Stefano Mizzaro, and Gianluca Demartini. Crowdsourcing Relevance Assessments: The Unexpected Benefits of Limiting the Time to Judge. In *Proceedings of the 4th AAAI Conference on Human Computation and Crowdsourcing (HCOMP 2016)*, Austin, Texas, 2016. This work forms the basis of Chapter 4.

- Andrew Turpin, Stefano Mizzaro, Eddy Maddalena and Falk Scholer. On Crowdsourcing Relevance Magnitudes for Information Retrieval Evaluation. In *ACM Transactions on Information Systems (TOIS)*, 35(3), doi: 10.1145/3002172, 2017. This work forms the basis of Chapter 5.

- Falk Scholer, Eddy Maddalena, Stefano Mizzaro and Andrew Turpin. Magnitudes of relevance relevance judgements, magnitude estimation, and crowdsourcing. *In Proceedings of The Sixth International Workshop on Evaluating Information Access (EVIA 2014)*, National Institute of Informatics, Tokyo, Japan, pages 9-16, 2014. This work forms the basis of Chapter 5.

- Eddy Maddalena, Stefano Mizzaro, Falk Scholer and Andrew Turpin. Judging Relevance Using Magnitude Estimation. In *Proceedings of the 37th European Conference on Information Retrieval (ECIR 2015)*, Vienna University of Technology, Austria, pages 215-220, 2015. This work forms the basis of Chapter 5.

- Andrew Turpin, Falk Scholer, Stefano Mizzaro and Eddy Maddalena. The Benefits of Magnitude Estimation Relevance Assessments for Information Retrieval Evaluation. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2015)*, Santiago de Chile, Chile, pages 565-574, 2015. This work forms the basis of Chapter 5.

Contents

Abbreviations

AP	Average Precision
ASP	Answer Set Programming
CLEF	Cross-Language Education and Function
CS	Crowdsourcing
DBSCAN	Density-based Spatial Clustering of Applications with Noise
DCG	Discounted Cumulative Gain
DoD	United States Department of Defense
ERR	Expected Reciprocal Rank
FIRE	Forum for Information Retrieval Evaluation
GMAP	Geometric Mean Average Precision
ICC	Intraclass Correlation Coefficient
INEX	Initiative for the Evaluation of XML Retrieval
IR	Information Retrieval
MAP	Mean Average Precision
mTurk	Amazon Mechanical Turk
NDCG	Normalized Discounted Cumulative Gain
NIST	National Institute of Standards and Technology
NLP	Natural Language Processing
NTCIR	NII Testbeds and Community for Information access Research
TREC	Text REtrieval Conference

1

Introduction

The massive diffusion of the Internet begun two decades ago considerably increased the amount of information available to us. The searching of material relevant to our needs is a daily challenge that companies, enterprises and organizations have to face for improving the quality of their services. In fact, the success search engines, social networks and online shopping sites depends on their capabilities in providing content that satisfies their users. These can choose their favourite systems among several available options, and, if they change their mind, they can easily and quickly move from a system to another that better fulfils their need. User satisfaction then plays a fundamental role in system's acceptance, and thus in its success.

In the 1960s, long before the advent of the Web, Information Retrieval (IR) was arising, a new research field aimed to support the work of librarians, who had the same need of current web users: to find specific information in document collections. The evolution of IR led to an automatization of the search processes. IR research has allowed the use of search techniques, which were improved over the years, until forming the basis of the search algorithms adopted by the Web platforms we daily use.

However, the adoption of an effective search algorithm is not enough to ensure the high-quality of a system. Every search context is different; the characteristics of the search domain, the nature of the document collection and the type of users directly influence the search process. IR effectiveness has to be constantly monitored in order to maintain a high search quality. For this reason, effectiveness evaluation is considered a topic of critical importance within IR research and development.

This thesis discusses system evaluation by means of test collections, which are benchmarks composed by a set of documents, a set of user needs (topics), and a series of relevance judgments for a subset of the document-topic pairs. Commonly, assessors who judge the relevance of test collection documents are experts and their judgments are considered ideal. Measurement system effectiveness is performed by running a system over a test collection and comparing the results of its searches with the judgements expressed by the assessor. A system is considered to be effective if it retrieves documents

that have been judged relevant also by the experts.

Unfortunately, the relevance assessment process is affected by several issues. For example, the process is highly subjective, not scalable, expensive, and very tiring since the experts spend a lot of time reading and judging large amounts of documents. Also, it is dependent many personal characteristics of the assessors, such as their educational level, domain knowledge, mood at judging time, and so on. Conscious of these limitations, in the recent years, researchers searched new ways to collect relevance judgments. One of these involved crowdsourcing, a phenomenon born on the web around ten years ago. It expects that people, having particular needs, create and upload simple tasks to ad-hoc web-based platforms. Users (also called workers) around the world can access the platform and complete those tasks. After the task completion, requesters pay a small amount to the workers for the work done. This business model seems to be particularly suitable for the creation of large test collections.

Researchers found crowdsourcing particularly interesting as an alternative way to collect relevance judgments. Crowdsourcing allows the creation of test collections by offering multiple benefits: the creation of test collections is more scalable thanks to the large amount of available workers, and can be split into smaller pieces to reduce the workload of each assessor, low cost allows money saving, tasks can be parallelized to save time, and redundancy of the gathered relevance scores make the entire process less subjective. This last point is particularly interesting because the analysis of the agreement among crowdsourcing workers can lead to discovering additional valuable information that would otherwise get lost. The research on crowdsourcing, and more general on human computation, attempts to maximise the positive impact of these benefits.

The idea of exploiting crowdsourcing to create test collections raises new issues. Unlike the experts, of whom we have secure guarantees, the workers are unknown; the information we can obtain about them is limited and sometimes not reliable, then, we have to think up techniques which guarantee a satisfactory quality level of their work. For example, workers can be required to answer explicit test questions or pass hidden quality check. Moreover, crowdsourcing tasks are usually designed to collect redundant data. For this reason, aggregation functions are necessary to transform results of more workers into final results. The choice of a good aggregation function can allow the exclusion of outliers, increasing the overall quality of the acquired data.

1.1 Motivations

This thesis aims to explore the crowdsourcing relevance assessment by focusing on three specific aspects: the *time factor*, the introduction of the *magnitude estimation technique* for expressing relevance judgements, and the *agreement* among workers. The three aspects and the reasons have for choosing them are briefly presented below:

1. The time factor in the relevance assessment process is an important aspect. The study of this aspect can show if optimizations of the timeframe available to workers allow to an overall improvement of their throughput. We discuss four experiments

in which we show how the time factor impacts the relevance assessment process. In particular, we discuss the benefits provided by the introduction of time limitations in the judging process. These studies aim to understand which is the right amount of time to be made available to assessors for judging a document. In fact, if the available time is not sufficient for performing an accurate judgement, the overall quality of the assessment is going to decrease. On the contrary, allowing too much time for assessment may either favour distractions or attract lazy workers. Furthermore, the optimisation of the amount of time available to the assessors, can both favour accurate relevance judgements and avoid waste of time and then lead to money saving.

2. Magnitude estimation is a psychophysical technique for the construction of measurement scales for the intensity of sensations. An observer is asked to assign numbers to a series of presented stimuli. The first number can be any value that seems appropriate, with successive numbers then being assigned so that their relative differences reflect the observer's subjective perceptions of differences in stimulus intensity. A key advantage of using magnitude estimation is that the responses are on a ratio scale of measurement, meaning that all mathematical operations may be applied to such data, and parametric statistical analysis can be carried out. In this thesis, we discuss the benefits obtainable by using magnitude estimation in relevance assessment.

3. The agreement between workers is an aspect that is sometimes neglected in crowdsourcing experiments. Aggregation functions only transform the results obtained by single individuals into the final result for the group, without considering how single workers agree among themselves. However, this aspect is interesting to study because the meaning of results obtained from workers who worked with a high agreement may be different from that of results obtained by workers who strongly disagreed. Agreement analysis adds valuable information to the computed aggregated results and can be used for many purposes, e.g., to discover problems in the task, to validate workers results, or to detect malicious workers.

1.2 Outline of the thesis

This thesis is organised in two parts: the first part discusses the background and related works (Chapters 2 and 3); while the second part discusses two experiments (Chapters 4 and 5) and some preliminary results (Chapter 6). Overall, the thesis is composed by seven chapters and one appendix. Short descriptions of each chapter's content are listed below:

Chapter 2 introduces crowdsourcing, discussing its typical form and some contrasting points of view about it. The chapter presents four open issues in crowdsourcing: workers' capabilities and honesty, answers aggregation techniques, disagreement among the workers, and crowdsourcing for mobile workers.

Chapter 3 presents the information retrieval field and the importance that the concept of relevance has for the field. The chapter also discusses effectiveness evaluation of IR systems and presents some initiatives which arisen to support the evaluation activity. The chapter then presents relevance assessment as a technique for creating test collections. Finally the chapter discusses the most used effectiveness metrics.

Chapter 4 discusses four experiments that we have performed to study the time factor in relevance assessment. The experiments involved crowsourcing workers from both India and United States, and studied the impact that time limitations have on the relevance assessment process. The material presented in the chapter is based on a published work [87].

Chapter 5 describes a large-scale crowdsourcing experiment aimed to verify the suitability of the magnitude estimation technique as an alternative way to express relevance judgments. The experiment requires that workers use this technique to judge the relevance of a set of documents. The relevance judgments collected by the workers are then compared with benchmarks to consolidate the obtained results. The material presented in the chapter is based on a published work [88].

Chapter 6 presents some results of a work, which aims to study the agreement among crowdsourcing workers. The chapter discusses how the analysis of agreement can lead additional information about a crowdsourcing task, and how it can be exploited to validate the results. Also, the chapter shows how topics with different levels of agreement differently affect information retrieval system evaluations.

Chapter 7 contains the conclusions and future work.

Appendix A describes how Answer Set Programming (ASP) can be used to facilitate the process of work subdivision among worker. This process is characterised by strict constraints that may be difficult to be satisfied by classic programming paradigms.

I

Background

2

Crowdsourcing

This chapter introduces the concept of Crowdsourcing (CS). Section 2.1 contains the definitions. Section 2.2 presents the five main forms of CS. Section 2.3 discuss some favourable and unfavourable thoughts concerning CS. Section 2.4 discusses some open issues in CS: worker's honesty (Section 2.4.1), results aggregation (Section 2.4.2), agreement among workers (Section 2.4.3), and CS for mobile devices (Section 2.4.4). Section 2.5 concludes the chapter.

2.1 Crowdsourcing definitions

In 2006, Howe published "The Rise of Crowdsourcing"[57], an article concerning a new phenomenon which has arisen in those years. Howe called it *crowdsourcing* as the the merging of the words *crowd* and *outsourcing*. The author described it as:

> *"Simply defined, crowdsourcing represents the act of a company or institution taking a function once performed by employees and outsourcing it to an undefined (and generally large) network of people in the form of an open call. This can take the form of peer-production (when the task is performed collaboratively), but is also often undertaken by sole individuals. The crucial prerequisite is the use of the open call format and the large network of potential laborers."*

Howe realised that a new modality for organising and assigning tasks and projects was becoming popular on the web. This was made possible by the technological improvements that affected the Internet in the 2000s, and by the awareness of its users, who began to realise that the web could become a suitable place for participating in collaborative projects. In the subsequent years, CS continuously evolved becoming more structured. New CS-based companies arose, and many researchers around the world focused their interest on the phenomenon.

CS has three main actors:

- the *requesters*, who are individual users or organisations having some work to be performed;
- the *contributors* (also called *workers*), who are people that offer for carrying out part of the work, for free or upon a payment;
- the *platforms*, also called *online labour marketplaces*, that are websites created in order to facilitate the crowd-work. Operating similarly to an agency, web platforms connect the requesters with available workers.

Although Howe was the first to realise the impact of CS on the web-oriented business, in the following years his definition has been considered by the academic community as too general and incomplete. Thus many researchers proposed alternative definitions. In 2012, Estellés-Arolas and González-Ladrón-De-Guevara [42] counted more than forty different definitions of CS. The authors presented a new consistent and exhaustive definition, based on those already existing:

> *"Crowdsourcing is a type of participative online activity in which an individual, an institution, a non-profit organization, or company proposes to a group of individuals of varying knowledge, heterogeneity, and number, via a flexible open call, the voluntary undertaking of a task. The undertaking of the task, of variable complexity and modularity, and in which the crowd should participate bringing their work, money, knowledge and/or experience, always entails mutual benefit. The user will receive the satisfaction of a given type of need, be it economic, social recognition, self-esteem, or the development of individual skills, while the crowdsourcer will obtain and utilize to their advantages that what the user has brought to the venture, whose form will depend on the type of activity undertaken."*

Even if the definition given by Estellés-Arolas and González-Ladrón-De-Guevara is widely accepted from the research community, the issue of the crowdsourcing definition remains a current theme [72, 43].

2.2 Forms of crowdsourcing

In the years following Howe's article [57], crowdsourcing increasingly developed, becoming more dynamic, reliable and suitable for new scenarios. Crowdsourcing offered various and heterogeneous possibilities and it was adopted in several types of applications. In order to explore these scenarios, several classifications of crowdsourcing were proposed. One of those is between *rewards crowdsourcing* and *equity crowdsourcing*:

- *rewards crowdsourcing* expects that the requesters pay an amount of money to the workers for performing one or more tasks;
- *equity crowdsourcing* typically consists of groups of workers who aim to achieve a common goal by working in a cooperative way without expecting any sort of

payment. Workers' motivations are mainly social (e.g., personal satisfaction, sense of belonging to a group of people, increasing of self-esteem or, simply, mere fun).

There are five major forms of crowdsourcing:

- the *crowdcontests* are online challenges in which requesters, through an open call, try to identify the best workers for a task. In the web platforms, the workers can first find the lists of available tasks with the relative descriptions and deadlines, and then choose which of those to perform. After the expiring of the deadline, the requester chooses the best worker across all the tasks performed and pays only the worker who carried it out. Some people criticise this model because it expects that a lot of workers perform tasks without getting paid [106]. Others appreciate it, highlighting its capability for talent selection [83]. An example of this is *99designs*[1], one of the most popular online graphic design marketplaces. People needing a design for a logo, a website layout, or any other graphical task, can start an open call with a detailed description of the task. Some workers can work on it until the deadline, and after that, the requester chooses the best creation among the completed works and provides the payment;

- the *macrotasks* require workers having specific skills. A requester creates a task and fixes a certain remuneration for his completion. Then, the requester hires a worker, who has to perform the task. The worker will get the compensation after completing the task. Usually, simple tasks are performed by single workers, and more complex tasks can require workers to create working teams. Typical areas of macrotasks crowdsoucing are: software development, writing and editing assistance, web-design and three-dimensional modelling. An example of platform that offers macrotasks is *Fiverr*[2]. This site connects people with freelancers who can fulfill macrotasks, such as, software developing, infographics designing, and music mixing.

- the *microtask crowdsourcing* is characterised by the division of big tasks into small units. Every worker performs one or more unit, and everyone gets paid for the completion of the task. Ad hoc online marketplaces manage the entire process. The task requester simply uploads a task on a platform. Every worker accesses the platform and chooses from a list the task that intends to perform. Typical examples of microtasks are: image labelling, transcription of business cards, participation to surveys and short text translation. Amazon Mechanical Turk (mTurk)[3] is probably the most famous platform for microtask crowdsourcing. It was used in 2008 by the designers Koblin and Kawashima for the creation of "Ten Thousand Cents"[4], a digital artwork that represents a $100 bill. Using a custom drawing tool, thousands of unaware individuals collaborated in isolation from one another, painting a tiny part of the bill, without knowledge of the overall

[1] `https://99designs.com/`
[2] `https://www.fiverr.com/`
[3] `https://www.mturk.com/`
[4] `http://www.tenthousandcents.com/`

Figure 2.1: "Ten Thousand Cents", the artwork made by unaware crowdsourcing workers in 2008.

task. Figure 2.1 shows the entire work, which involved workers from 51 different countries and had a cost of 100$.

- the *crowdfunding* allows fundraising for either profit or charity. Two interesting examples are Crowdcube[5] and Razoo[6]. The first one is a British crowdfunding company that allows everyday investors, professionals and venture capitalists to invest in different types of projects. Usually, the funding campaigns involve start-ups or emerging companies. The investor can invest as little or as much they like (starting from £10). Razoo is a nonprofit organisation which promotes fundraising for personal or charitable purposes. Through their donations, benefactors can support various campaigns, such as the rebuilding of houses after a flood, the donation for the development of hospitals in African or the training of guide dogs;

- the *self-organised crowds* allows a requester to create a challenge, that is posted on a crowdsourcing platform. Self-organized crowd teams compete to provide the best answer for the proposed challenge. Only the winning team gets the compensation. One of the major company which uses the self-organized crowds is Innocentive[7], a platform which allows customers to publish open calls concerning research and development problems in engineering, computer science, math, chemistry, life sciences, physical sciences, and business.

Macrotasks and microtasks are the crowdsourcing forms most used for research purposes [25]. This thesis exclusively focuses on microtasks crowdsourcing.

[5] `https://www.crowdcube.com`
[6] `https://www.razoo.com`
[7] `https://www.innocentive.com/`

2.3 Is really the crowd so good?

Some of the first questions that people ask when hearing for the first time about crowdsourcing are: "are the crowds so good? Why should I trust someone (a worker) that I do not know? Can a group of non experts guarantee as good performance as that of an expert?". All these questions are legitimate and the answers could be not trivial. The issue about the (in)capability of the crowds to make right and wise decisions has fascinated philosophers, thinkers and unfortunately also dictators since long. For some of them, the crowds offer a series of advantage that single individuals cannot offer, for others the crowds are stupid and dangerous and sometimes are considered the responsible of serious failures in the history of humanity. The American journalist James Surowiecki is probably one of the biggest supporters of the crowd's capabilities. In "The Wisdom of Crowds" [120] Surowiecki presents several case studies and anecdotes in which the ideas of a group of people result in better outcomes than those of a single expert. Surowiecki claims that in order to make crowdsourcing effective, four principles should be guaranteed:

- the *diversity of opinion*: each person in the crowd should have a personal interpretation of the known facts;
- the *independence*: people in the crowd should not influence each other;
- the *decentralization*: people can specialize and make the decisions on own local knowledge;
- an *aggregation* mechanism which turns private judgements into a collective decision.

The design of traditional crowdsourcing natively guarantees *diversity of opinion*, *independence* and *decentralization*. As we will see in Section 2.4.2 and in the next chapters, the *aggregation* phase is left to the requester. Analysing some historical failures in which the crowd produced disastrous judgments, Surowiecki realizes that some problems systematically affect the decision-making environment. For instance, in 2010 during the Duisburg Love Parade, 21 people died, and more than 500 were injured, because of overcrowding. Participants, who did not mind the Police announcing, continued to enter in an overcrowded festival area. The causes of the tragedy are attributed to a phenomenon called "crowd turbulence" [54]. In total Surowiecki identified five causes of crowdsourcing failures:

- the *homogeneity* of the crowd, that leads to a lack of variety in the solutions found;
- the *centralization* of information, which prevents the members of the group from obtaining sufficient data to find proper solutions;
- the *division* of information, when the information is not distributed among all members of the decisional group. Surowiecki claims that crowds work best when they choose for themselves what to work on and what information they need;

- in a contest where people can influence each other, and decisions have to be taken in a certain sequence, any decision may be directly affected by some others that may have been previously taken. This effect is called *imitation*. In this scenario, most charismatic or influential people can affect the choices of others. This behavior is unsafe because it may lead to fragile social outcomes;

- the *emotionality* of the crowd can manifest through factors such as the feeling of belonging, which can lead to peer pressure, the herd instinct, and, in extreme cases, collective hysteria.

Even though many people as Surowiecki believe that the crowd can be clever and wise, other researchers are sceptics and critic [123, 11]. In 1895, Gustave Le Bon, a social psychologist and sociologist, strongly criticized the crowds psicology. In "The Crowd: A Study of the Popular Mind" [15], the author described the psychological crowd as:

> *".. a provisional being formed of heterogeneous elements, which for a moment are combined, exactly as the cells which constitute a living body form by their reunion a new being which displays characteristics very different from those possessed by each of the cells singly."*

Human behaviour, according to Le Bon, changes considerably from a single person to a group of people. He believes that this change is negative and leads to a series of harmful effects. Le Bon considers the crowd psychology impulsive, irritable, unreasonable, with a complete lack of critical spirit and characterised by the amplification of the feelings. From the writings of Le Bon emerges a lack of consideration towards the collective thought considered a disincentive for people who wanted to control the crowds in order to harness their massive power. The works of Le Bon became popular few decades after being written and inspired the insane ideas of the totalitarian dictators of the Twentieth Century.

2.4 Open issues

Thanks to its adaptability in many different areas, crowdsourcing arises a considerable interest whit in the research community. Researchers studies crowdsourcing from different points of view: sociological, psychological, political, economical. In this thesis, we explore the phenomenon from the Information Science perspective. The next paragraphs list some the most debated issues of crowdsourcing that have emerged in the recent years: workers' capabilities and honesty, answer aggregation tecniques, (dis)agreement among workers, and Crowdsourcing for mobile users.

2.4.1 Workers' capabilities and honesty

The guarantee of minimum workers' capabilities level and the checking of their honesty are two issues that have plagued crowdsourcing since its rise. Unlike the traditional workplaces, where usually an employer can monitor the employees' commitment and

honesty, in crowdsourcing scenario the supervision process is not immediate and can require several forethoughts. The incapacity of a worker in solving easy problems; unskilled workers try to perform the task in good faith, and the potential errors they make are usually easy to detect and to manage. Several basic techniques guarantee that each worker has the minimum competencies for carrying out a task. Most of the times it is sufficient to ask the worker some test questions before the beginning of a task; enabling or not the worker to the task strictly depends on his answers.

A crowdsourcing problem, more serious than the workers' incapability, is represented by the workers' dishonesty. If neglected, this may irreparably compromise the results of an entire experiment. Malicious workers are exclusively interested in profit, by cashing as much as possible, as quickly and effortlessly as possible. Dishonest workers perform tasks hastily and approximately, neglecting the final quality of the work. Basic techniques are usually not effective to keep them away [3, 79]. Task designers have to think of ad hoc solutions, called *quality check*. One of these is the adoption of "gold standard" questions. This technique consists in asking workers some easy questions of which the answer is a priori known. The task checks automatically and in real time the correctness of the answers. A failure in the gold standards questions is a clear indication of lack of user's effort, which causes the anticipated termination of the task.

Due to its high negative impact in crowdsourcing, workers' dishonesty has been widely discussed in many works [52, 119, 138]. In 2008, Fischbacher and Föllmi-Heusi asked workers to roll a die in private, and then to report their rolls outcomes [45]. The retribution expected by the workers was proportional to the outcome of the rollings (except for the number six which was trated as a zero). According to statistics, the authors expected each number 17% of the times, but the outcomes reported were quite different: The numbers four and five, which rewarded workers with the highest gain, were reported respectively 27% and 35% of the times, and the number six only 6.4%. It was clear that some workers, to maximise their gain, cheated the rolls reporting.

2.4.2 Answer aggregation tecniques

A crucial phase of an crowdsourcing experiment is the aggregation of workers' results. This process consists in transform the results obtained from individual workers into the outcome of the group. This transformation can strongly affect the overall experiment results and can be a discriminant key for its success or failure. For this reason, the decision about which aggregation function to adopt in an experiment has to be taken carefully.

2.4.2.1 Aggregation techniques

Usually, tasks requiring the workers to provide information as numbers, choices, or labels do not need complex aggregation techniques; one of the followings options is a reasonable solution:

- *majority voting* is the most popular aggregation technique. This consists into considering the most selected label received by an item as the final one.

- *arithmetic mean* is a straightforward but effective solution especially when values inputted by workers are expressed in bounded scales a Likert scale [81].

- the *median* is an alternative to the mean that offers particular advantages for unbounded scales, where even a single outlier may strongly affect the aggregated value. For its capabilities in removing the outliers, the median is considered more "democratic" than the mean.

2.4.2.2 Advanced aggregation techniques

The aggregation techniques defined above are suitable for the majority of crowdsourcing tasks, but there are also other types of tasks which require more sophisticated algorithms for aggregation. For example, some tasks require the workers to type free text, and how to properly aggregate different texts into a final single text is a subjective operation which requires capabilities that even modern artificial intelligence is still far to have. . Natural Language Processing (NLP) researchers have handled these issues for a long time. Despite the fact that researchers have improved the algorithms for managing the human language, the quality of their results is still not comparable to the one obtainable by humans. This is one of the reasons that makes crowdsourcing attractive for NLP purposes [55, 101].

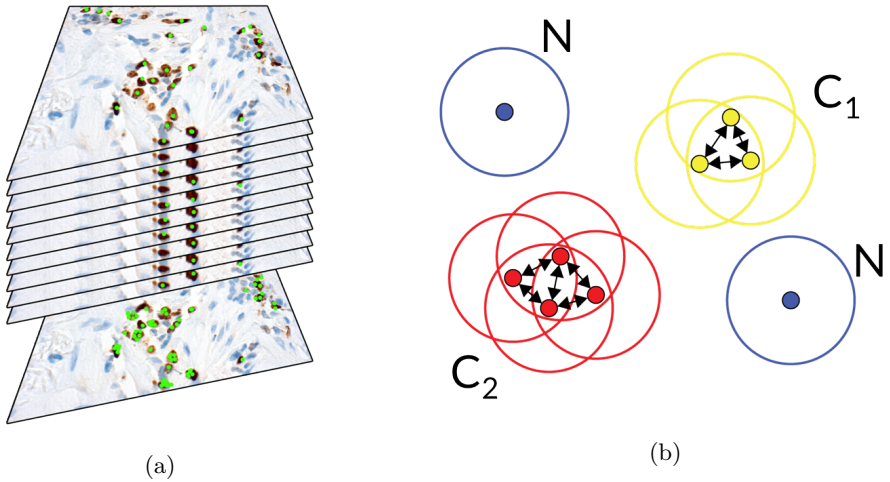
More advanced aggregation techniques have been proposed. Hosseini et al. [56] show that the usage of an *expectation maximization* model to aggregate crowd judgments yields to more stable evaluations compared to majority voting. More recently, it has been shown that better results can be obtained considering a measure of worker similarity based on the type of errors workers perform [125]. Thus, by identifying worker communities, better answer aggregation techniques can be defined.

To allow for standard and repeatable comparisons of aggregation techniques, ad-hoc benchmarks have been proposed. For example, SQUARE [110] serves as a comparative test collection for aggregating relevance judgments collected from the crowd. Hung et al. [58] proposed a benchmark based on simulated crowd answers, which allows to study how to fine tune parameters of crowd aggregation models. Nguyen et al. [97] developed a benchmarking tool aimed to evaluate aggregation techniques in different aspects, like accuracy, and sensitivity to spammers. By using Nguyen's tool, researchers can experiment with existing (or new) aggregation techniques.

2.4.2.3 An example of an advanced aggregation technique

The recent evolution of the Web together with several technological improvements that involved electronic devices and networks led to the stabilisation of the HTML5 standard, that has made simple the development of even more sophisticated tasks. These tasks require advanced activities, for example, video annotations, video games testing, and drawing of shapes on virtual canvases.

These tasks usually require sophisticated aggregation techniques. An example is to ask workers to perform an activity which is generally made by pathologists, consisting in



(a) (b)

Figure 2.2: (a) Merging of the detections of ten images in only one. (b) Operational schema of *DBSCAN*: `C1` and `C2` are two legitimate clusters.

the detection of particular cells, having specific characteristics, by examining biological images [33, 131].

For each image, the authors collected the detections of ten workers. Each worker was free to detect a different number of cells. We aggregated the detection obtained on every image, in order to compare the results of the crowd with a ground truth. Furthermore, we wanted to check if the quality of the work of a group would be higher than the average quality of the work of the individual worker, as suggested by Bachrach et al. [12]. The aggregation function required three steps:

1. project all the detections from an image, as shown in Fig. 2.2a;

2. run the Density-based Spatial Clustering of Applications with Noise (DBSCAN) algorithm [44], over that image. DBSCAN allows the clusterization of points on a cartesian plane by setting two important parameters: `minPts`, i.e., the minimum numbers of points to form a cluster, and `eps`, i.e., the maximum distance among the core point in a cluster and all other points. An example of DBSCAN groupings is shown in Fig.2.2b;

3. tune the algorithm in order to obtain performance similar to those of the gold standard.

When compared to the benchmark, the aggregation detection obtained by the adoption of the DBSCAN algorithm were on average better than the detection of the single workers.

2.4.3 The (dis)agreement among workers

The agreement among the workers is an interesting issue in crowdsourcing. Since workers should be completely independent and they can not influence each other (this cannot be totally guaranteed, but can be limited and monitored with proper checks, e.g., by studying the IP address of the workers), the eventual agreement among their results can represent a significant indication of high workers reliability, and it can be used for consolidating the gathered results. Analogously, an excessive disagreement can be a valuable warning for either a problem in the task design or a technical malfunction. For example, if we ask workers to estimate the price of a house shown in an image, and the variability between their estimates is low, we can be sure enough about the correctness of their answers. On the contrary, if the variability is high, we can investigate the reasons: Did the workers understand the instructions? Did they perform honestly? Is the cost of that house quite variable depending on its location? All such hypotheses are plausible and should be deeply studied since, if ignored, the overall quality of the task results could decrease. The agreement in crowdsourcing experiments is related to its difficulty. Workers who are asked to carry out complex tasks may have not the required skills to perform the task correctly; then it is reasonably expected many by chance answers that negatively impact to workers' agreement. Abraham et al. covered the problem of task difficulty and adaptivity for quality control [1]. Some researchers study the suitability of crowdsourcing for design very difficult tasks, e.g., Zaidan and Callison-Burch collected translations by crowdsourcing non-professional translators [137]. The authors demonstrated a variety of mechanisms that increase the translation quality to near professional levels. Historically, several measures of agreement have been proposed. Some of the most popular are:

- Cohen's kappa [74] and intraclass kappa coefficient [75] are widely used for quantifying the agreement between two raters who annotate by using binary and nominal scales (the scales of measurement are discussed in detail in Section 3.5.1). The weighted kappa [28] coefficient is especially useful when codes are ordered.

- Fleiss's kappa [46] measures the agreement among groups of annotators;

- Intraclass Correlation Coefficient (ICC) [73] compares the variability of different ratings of the same subject to the total variation across all ratings and all subjects.

Several works studied the relation between the level of agreement and exepected quality.

Dumitrache [38], collecting semantic annotation, observed that the disagreement between annotators can represent a signal of presence of ambiguity in the input text. Soberón et al. [114] exploited the disagreement between individual annotators and harnessed it as an useful information to signal vague or ambiguous text. Chklovski and Mihalcea [26] explores the agreement (and disagreement) of human annotators for word sense disambiguation. Inel et al. [59], Aroyo and Welty [10] proposed a framework to exploit different human responses to annotation tasks for analysing and understanding disagreement [59, 10]. Kairam and Heer [64] identified systematic areas of disagreement

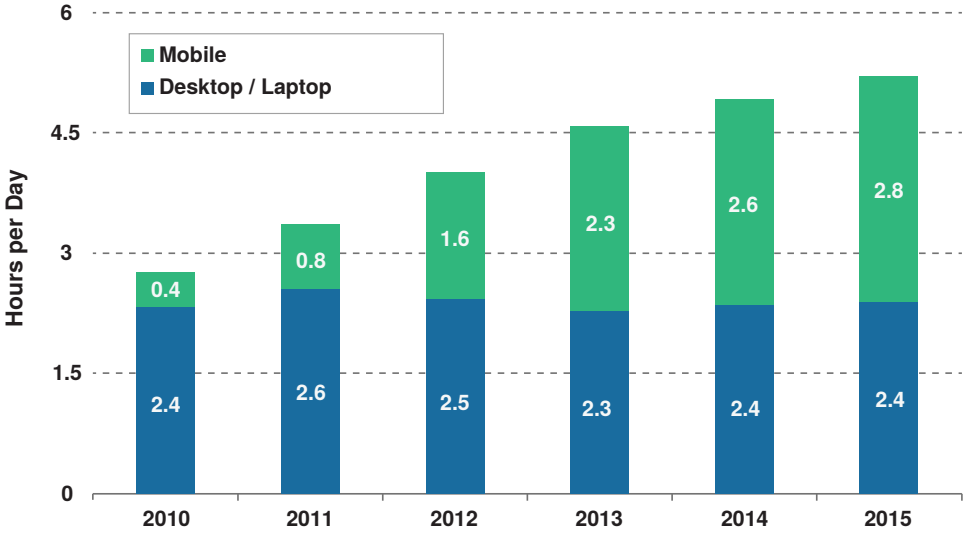


Figure 2.3: Total digital media time spent for each platform (Source: eMarketer).

between sub-groups of workers in a realistic task, in which workers annotated entities. The problem of workers' agreement was covered by Alonso et al., that described a process aimed to debug a crowdsourced labelling task with low inter-rater agreement [6]. Summarising, the analysis and the understanding of the meaning behind the workers' disagreement can have a crucial role in the interpretation of the outcomes of experiments.

2.4.4 Crowdsourcing for mobile users

The number of Internet accesses using mobile devices has continuously increased in the recent years. The *eMarketer*[8] company has published a survey that shows how from 2010 to 2015 the hours per day spent on the web by users using mobile devices increased from 0.4 to 2.8, while the number of hours spent using desktop devices remained constant. However, the current crowdsourcing platforms were designed before the massive diffusion of mobile devices. Their interfaces were conceived for desktop devices, and they have never been redesigned for interaction on mobile.

In order to verify this intuition, we conducted two studies aimed to measure the differences in terms of suitability of the existing crowdsourcing platforms toward the desktop and the mobile devices [34, 32]. The results confirm our intuition showing that there is still a considerable gap in the quality of the service that the existing crowdsourcing platforms offer to desktop and mobile users. Other researchers studied crowdsourcing for mobile users as well; Kucherbaev et al. [77] proposed *CrowdCafe*, a

[8] www.emarketer.com

crowdsourcing platform where people can perform micro-tasks using their smartphones while doing daily activities. Fuchs-Kittowski and Faust [47] proposed a general architecture and a classification scheme for mobile crowdsourcing systems, aiming to gain a better understanding of typical functionalities and design aspects to be considered during development and evaluation of such collaborative systems. Kumar et al. [78] propose *Wallah*, a crowdsourcing platform that promotes the diffusion of crowdsourcing in developing countries by overcoming technological limitations.

2.5 Conclusions

This chapter introduces Crowdsourcing and its five main forms: *crowdcontest*, *macrotask*, *microtask*, *crowdfunding* and *self-organised crowds*. Subsequently, the chapter presents different points of view of the capabilities of crowds, some of them positive [120] and others critical Bon [15]. Finally, the chapter analyzes in more detail some open issues of crowdsourcing: workers' honesty, the meaning of agreement among them, and results aggregation.

3

Relevance assesment

This chapter introduces *relevance assessment* which is a technique to gather relevance judgements in order to create test collection for IR evaluation. Section 3.1 defines IR and summarizes its history. Section 3.2 introduces the concept of relevance and discusses issues related to its definition. Section 3.3 focuses on the importance of evaluation in IR. Section 3.5 describes relevance assessment. Section 3.6 introduces evaluation metrics. The chapter ends with the conclusion (Section 3.7).

3.1 Information retrieval

In our daily lives, we look more and more frequently for information, and usually we do it online. We can check if our favourite shop offers exclusive discounts, consult train timetables for planning our trips, or find useful information for scheduling the next holidays. The massive diffusion of connected devices has given a considerable boost to the development of search systems, which now are able to find information in fast and effective ways. Search activities have become so natural that we began to consider it as a natural activity that accompanies us in our lives. Furthermore, some smartphone applications are programmed to perform automatic searches unbeknownst to us. It happens, for examples, when we walk on a street at dinner time, and our smartphone automatically provides to us a list of nearby restaurants. All these possibilities, unimaginable just a few years ago, have been made possible thanks to more than half century of IR research.

The term "Information Retrieval" has been coined in 1951 by Calvin N. Mooers who defined IR as [95]:

> *"... the finding or discovery process with respect to stored information ... useful to him [user]. Information retrieval embraces the intellectual aspects of the description of information and its specification for search, and also*

> *whatever systems, technique, or machines that are employed to carry out the operation."*

In those years, "machine literature searching" was studied for helping the work of librarians, who needed to automate the process of searching specific information in a large amount of data. The original idea of using automatic systems for retrieving documents to satisfy specific information needs arised a few years before Mooers's definition, precisely in 1945, when Vannevar Bush published his famous paper entitled "As We May Think"[19]. Between the 50s and 60s, the first IR systems were arisen. Those systems were the ancestors of current search engines and their operating principle was the same: a person (user) submits a request (query) to a search engine, which searches into a collection of documents for those that contain information that is more useful for the user and proposes them to the user. The subsequent evolution of computer technologies gave rise new challenges for the IR researchers who worked to make IR systems more efficient, capable of working faster and on larger document collections. In 2008, Manning et al. formalise a new definition of IR [89]:

> *"Information retrieval is finding material (usually documents) of an unstructured nature (usually text) that satisfies an information need from within large collections (usually stored on computers)."*

Manning's definition associates the search process to the concept of "Information Need": a document retrieved by a system is relevant if it contains information that is useful to what the user is looking for. Unfortunately, establishing if a document is relevant to a topic is an activity that leads to several issues, such as the subjectivity of the assessors. In the 90s the advent of the Web increased the interest for IR; in fact, the net makes available a vast collection of documents, which is growing day by day. It led to the arising of "Web Information Retrieval", which opened the era of modern commercial IR systems.

3.2 The concept of relevance

Even though relevance is a fundamental concept in IR [102, 124, 60] and it has been discussed in many papers, it is still not completely understood. The entire search process is based on the concept of relevance: an effective system has to provide to the user documents that are relevant for her/his request. But, what does "relevant" really mean? Can we define the concept of relevance? Many researchers have tried to find convincing answers to these questions. The first explicit studies about relevance are dated 1959. Before this date, the problem of finding relevant information was already known by the librarians, but the concept of relevance was treated only implicitly, behind the scenes of other studies. In 1959, Vickery presented at the ICSI debate a distinction between the "relevance to a subject" (relevance of a document to a query for what concerns the topical component) and the "user relevance" (that refers to what the user needs) [129, 128]. In modern IR the relevance to a subject is called *topicality*: a document is considered topically relevant to a query if it is about the same topic.

Table 3.1: The patterns in Saracevic's definition of relevance.

A	*B*	*C*	*D*	*E*
measure	correspondence	document	query	person
degree	utility	article	result	judge
dimension	connection	textual form	information	user
estimate	satisfaction	reference	used	requester
appraisal	fit	information	point of view	information
relation	bearing	provided	information	specialist
	matching	fact	requirement	
			statement	

Vice versa a document is considered relevant by the user if it contains the information that the user is looking for. Not necessarily a document having a high topicality with respect to a query (then presumably retrieved by a system) is relevant for the user that typed the query. For example, a document about the "Venice Film Festival" would be relevant to the query "Events in Venice"; however, the user may not consider relevant the document if it refers to an old version of the festival, or it is written in a language incomprehensible for him/her, or he/she already knows the information contained in the document and he/she is looking for a more detailed information.

After Vickery's work, many researchers tried to define the concept of relevance in various ways. In 1975, Saracevic created (and criticised) a general pattern of definition, assembling various attempts of relevance definitions formulated by researchers in the previous years [104]:

> *"Relevance is the* `A` *of* `B` *existing between a* `C` *and a* `D` *as determined by an* `E`*."*

Every slot (`A`, `B`, `C`, `D`, and `E`) in the pattern can be replaced with one of the terms in the columns of the schema of Table 3.1. Saracevic criticises all the definitions, considering them a sort of paraphrasing. Also, the author underlines that the definitions do not establish "primitive terms" before to proceed to the more complex ones.

Furthermore, Saracevic claimed that the definitions mix the concept of relevance with its measurements, do not clearly identify that are the relation between the entities in `C` and those in `D`, and do not distinguish between different types of relevance [105]. The last observation is particularly interesting, as Saracevic considers the existence of many types of relevance the main reason for the difficulties found trying to formally define the concept of relevance. Saracevic believes that relevance is a primitive concept that has not to be defined. Therefore, in the "long story of relevance" [93], a formal definition of relevace, universally accepted by all the experts has never been found. Mizzaro identifies two reasons responsible for the failures in finding a proper definition for relevance [94]:

1. there are many kinds of relevance, not just one [105];

2. researchers use inconsistent terminology.

The author argues that relevance can be represented as a point in a four-dimensional space, where the dimensions are *information resources*, *representation of the user's problem*, *time*, and*components*.

3.3 Evaluation

Some decades ago, IR systems were a prerogative of expert librarians. The first IR systems were affected by several limitations and inefficiencies. A simple misspelling or typo in a query could determinate a failure in the search process. Therefore the use of these systems required particular attention and it was reserved to experts. Users had to adapt to systems in order to avoid errors in searches. Today this scenario seems obsolete; we are used to perform many searches every day, using different systems, always expecting good results in a short time. If a search engine does not satisfy us, we can decide to use another one we believe to be better. Since the business model of commercial search engine is based on advertisement, IR companies put a lot of effort for improving the service quality which guarantees a high lever of user satisfaction that indirectly leads to earnings for the companies.

The improvement of an IR system can involve many aspects such as results visualisation, amount of time to answer a query, query suggestions, and so on. However, the most important aspect that can be improved is the retrieval capability: ideally, given a query, a "good" search engine should be able to retrieve exclusively those documents which are relevant to the query, and present them to the user (preferably) well sorted according to their respective relevance degree. Unfortunately, all these activities are not so simple to do. General solutions which are suitable for every search scenario do not exist. Every IR system has to be minutely tuned for working in specific contexts. The adjustments depend on multiple variables such the search domain, document types, query language, etc. Cleverdon and Keen [27] identify six measurable factors that should be taken into account when evaluating an IR system: *coverage*, *time*, *presentation*, *effort*, *precision*, and *recall*. Traditionally, IR evaluation allows to measure the effectiveness of searches. It can be done in two ways: by means of either user studies or test collections. The two methods are discussed in Sections 3.3.1 and in Section 3.3.2.

3.3.1 Effectiveness measured by means of user studies

The effectiveness measured by means of user studies consists of user tests which involve several participants, who are asked to use an IR system in test sessions. During the trials, the examiner may monitor the participants by using techniques like eye tracking, skin conductance response, perceived level of stress, and even more. A test can involve either natural or artificial (induced by the examiner) user information needs. Some tests can expect that the examiner asks the participants to fill out surveys aimed to gather information from users such user's satisfaction and perceived problems. The main advantage of this method consists in the amount of information that can be collected

about users and their behaviour when actually using a system. As it happens for other user-based studies, this method is affected by some problems such as difficult-repeatability and difficulties in recruiting large sets of participants in terms of both availability and costs [71].

3.3.2 Effectiveness measured by means of test collections

A widely-used approach in IR effectiveness measuring does not involve users, but it focuses on IR systems. This method, called *system-oriented evaluation*, requires test collections (also called benchmarks), which consist of [103, 133]:

- a set of search topics;
- a set of documents;
- human-generated assessments that indicate, for an answer document returned by a search system in response to a topic, whether the document is relevant or not.

The evaluation of an IR system consists in carrying out a set of searches over a set of test queries. A series of evaluation metrics then measure the similarities between the results produced by the system over the given queries and a benchmark of human annotations which is considered ideal. The metrics use these similarities for generating the final system's scores which are measurements that give an overview of the overall system effectiveness. The system's scores are also crucial for discovering differences among performances of multiple systems. In fact, metric scores analysis highlights peculiarities of each system, and their strengths and weaknesses.

Unlike the evaluation by means of user studies, the system-oriented evaluation produces measurements which are repeatable and extensible. A common practice consists in adopting evaluation through test collections for system tuning purposes.

The first evaluations based on test collection were developed in the 1950s. In those years, working on the "Cranfield projects", Cyril Cleverdon, for the first time, uses a manually labelled test collection for identifing, according to evaluation metrics, the best of a series of search algorithms [27, 63]. The Cranfield projects laid the foundation of research in IR evaluation. Subsequently, in the US, Salton adopted the same modality used for the Cranfield's experiments to evaluate IR systems based on "Vector Space Model" [37].

This thesis focuses only on the effectiveness of IR systems measured by using test collections, then aspects of evaluation by means of studies are not discussed.

3.4 Initiatives for test collection based evaluation

Several initiatives arose in order to support the research on IR. This thesis focuses on Text REtrieval Conference (TREC)[1], a campaign that since 1992 encourages IR research. Arisen by a collaboration between the National Institute of Standards and Technology (NIST) and the United States Department of Defense (DoD), TREC provides to

[1] `http://trec.nist.gov/`

researchers benchmarks for repeatable experiments. TREC organises IR challenges in which dozens of international groups participate with their own retrieval systems. At each TREC edition, NIST provides test collections and proposes some search tracks. Some of the most popular are:

- *Web Track*: explores web-specific retrieval tasks over collections of up to one billion web pages [29].
- *Federated Web Search Track*: supports techniques for the selection and combination of search results obtained from real on-line web search services [36].
- *Microblog Track*: investigates the nature of real-time information needs in context of microblogging environments [82].
- *Contextual Suggestion Track*: explores search techniques for information needs which are strictly dependent on context and user interest [31].
- *Session Track*: provides resources for simulating user interactions over a sequence of queries, rather than for a single query [22].

Participants run their systems over the collections and send to NIST the results of the performed searches (which are called runs). NIST experts judge the results according to a pooling strategy. Using specific metrics, NIST evaluates then the systems results, creating the final rank of the systems. The data of past TREC editions is available; the judgements of the experts and the systems results for free (downloadable from the TREC website), and the documents for free o for a fee.

TREC is not the only campaign; there are other initiatives similar to TREC, such:

- NII Testbeds and Community for Information access Research (NTCIR)[2], a Japanese series of evaluation workshops. NTCIR aims to encourage research in Information Access technologies by providing large-scale test collections and investigating evaluation methods for constructing reusable datasets;
- Cross-Language Education and Function (CLEF)[3] is the European counterpart of TREC that focuses on Cross-language information retrieval.
- Forum for Information Retrieval Evaluation (FIRE)[4] is a conference which aims to build a South Asian counterpart of TREC, CLEF, and NTCIR. FIRE continuously expanded in recent years also by exploring new domains such as plagiarism detection, legal information access, the spoken document retrieval.
- Initiative for the Evaluation of XML Retrieval (INEX)[5] promotes evaluation of focused retrieval by providing large test collections of structured documents, uniform evaluation measures, and a forum for organizations to compare their results.

[2] http://ntcir.nii.ac.jp/
[3] http://www.clef-initiative.eu/
[4] http://fire.irsi.res.in/fire/2016/home
[5] http://inex.mmci.uni-saarland.de

3.5 Relevance assessment

The activity in which an assessor establishes the relevance degree of a document with respect to a topic is called "relevance assessment". The process involves topics that consist in artificial information needs which are abstractions of real information needs. A real information need is difficult to simulate because it depends on many variables; some of them easily measurable, others are psychological and dependent on personal preferences, experience and skills.

3.5.1 Scales of measurement

Relevance assessment can be seen as a measurement in which an assessor judges the pertinence of a document with respect to a given topic. Relevance assessment requires that an assessor assigns documents relevance scores according to a predetermined scale.

For many years, researchers studied scales of measurements. Stevens [117] categorized four type of scales: "nominal", "ordinal", "interval", and "ratio". The four scales are described as follows:

- *nominal* scales expect a differentiation of the items on their names, categories or other qualitative classifications and allow the determination of equality. For example, animals can be classified as "fish", "amphibian", "reptile", "bird" or "mammal";
- *ordinal* scales allow for rank ordering (1^{st}, 2^{nd}, 3^{rd}, etc.). The measured items can be sorted, but the relative degree of difference between them is not known. The measuring of opinion is a good example of an ordinal scale: someone could "not agree", "partially agree", or "totally agree" with respect to a given topic;
- *interval* scales determine the equality of intervals or differences. The interval type only allows for the degree of difference between items and do not allow the ratio between them. For example, the Celsius scale for temperature measurements has two fixed points: freezing and boiling point of water. These points are subdivided into 100 intervals. The ratio between intervals is not meaningful: to say that 15 °C are "half as cold" as 30 °C does not make sense.
- *ratio* scales have a unique and meaningful fixed point zero. These scales are the most common in the physical world and are used for measuring mass, length, duration, and many other physics magnitudes.

Historically, relevance judgments were made by using *binary* scales. Therefore, every judged document was either considered relevant to the search topic or not. More recently, on the basis of the observation that searchers can generally distinguish between more than two levels of relevance, multiple levels ordinal scales have been proposed for relevance judgments gathering [61, 133]. Even though the usage of operationalizing relevance for IR systems evaluation has many years of history, this is affected by many open issues. For example, it is unclear how many relevance categories should be used

when using an ordinal scale [121]. Sormunen [115] re-assessed some TREC-7 and TREC-8 documents on a four-level relevance scale (`H` for highly relevant, `R` for Relevant, `M` for Marginally relevant, and `N` for Not relevant).

3.5.2 Magnitude estimation

Magnitude estimation is a psychophysical technique for the construction of measurement scales for the intensity of sensations. An observer is asked to assign numbers to a series of presented stimuli. The first number can be any value that seems appropriate, with successive numbers then being assigned so that their relative differences reflect the observer's subjective perceptions of differences in stimulus intensity [39]. A key advantage of using magnitude estimation is that the responses are on a ratio scale of measurement [48], meaning that all mathematical operations may be applied to such data, and parametric statistical analysis can be carried out. In contrast, for ordinal (or ranked category) scales certain operations are not defined; for example, the median can be used as a measure of central tendency for ordinal data, but the mean is not meaningful since the distance between the ranked categories is not defined [111].

Magnitude estimation has a long history, and is the most widely-used psychophysical ratio scaling technique [48]. Initially developed to measure perceptions of physical stimuli, such as the brightness of a light or the loudness of a sound, magnitude estimation has also been successfully applied to a wide range of non-physical stimuli in the social sciences (including occupational preferences, political attitudes, the pleasantness of odors, and the appropriateness of punishments for crimes [118]), in medical applications (such as levels of pain, severity of mental disorders, and emotional stress from life events [48]), in user experience research (for example, as a measure of usability in HCI [92], and for healthcare applications [62]), and in linguistics (including judging the grammaticality of sentences [13]).

Eisenberg [41] investigated magnitude estimation in the context of judging the relevance of document citations from a library database (including fields such as author, title, keywords and abstract), and concluded that participants are able to effectively use magnitude estimation in such a scenario. A related technique was used by Spink and Greisdorf [116], where participants in a user study were required to fill in a worksheet with information about the relevance of resources that were retrieved from a library database for personal research projects. This included indicating the level of relevance on a 4-level ordinal scale, providing feedback about other levels of relevance such as utility and motivation, and marking the level of relevance on a 77mm line. The line was then decoded into numbers at a 1mm resolution so was in effect a 78-level ordinal scale.

3.5.3 The time factor

The "Time Factor" is another interesting aspect of the assessment process. It consists in measuring the amount of time that assessors need to perform relevance judgments. This data can be important in order to optimise the assessment process.

In 1998, Ahituv et al. [2] showed that time pressure typically impacts performance of decision makers working with information. However, experience helps in dealing with

time constraints. Most crowd workers are used to optimize the time needed to complete tasks. Yilmaz et al. [136] found that the effort taken to judge a document correlates with the utility of a document to an end-user with an information need rather than with document relevance. Verma et al. [127] claimed that the effort to judge relevance should be included in IR system evaluation metrics together with the relevance of retrieved documents. Halvey and Villa [51] looked at the effort needed to judge the relevance of images, and measured the effect of image features and topic properties (e.g., difficulty) on effort and quality of judgments. The authors also looked at the difficulty of judging document relevance, showing how borderline relevant documents require more effort to judge and that document size has an impact on the judging difficulty as well [130]. Wang [134] observed that relevance assessor's speed increases when the perceived difficultly of the task is low and that the judging accuracy increases when perceived difficulty increases. Verma and Yilmaz [126] observed different judging times of similar interrater agreement on both desktop and mobile devices.

3.5.4 Relevance assessment using crowdsourcing

As shown in Chapter 2, crowdsourcing offers several advantages when needing to collect large quantities of annotations, in a cheaper way and in a short time. These features can be very useful to create test collections. Traditionally, such process involved human experts, then it results difficult to scale and expensive in terms of both time and economic resources. In order to overcome these difficulties, several attempts of replacing expert assessors with crowd workers have been done [23, 113, 68, 30]. The adoption of the crowdsourcing based paradigm for relevance assessment allows to reduce the impact of the above mentioned problems, but at the same time, it opens the door to new issues. If neglected, all the issues mentioned in Section 2.4 can negatively impact on the assessment process. Therefore, those should be carefully considered and managed.

Alonso and Mizzaro compared the relevance judgment gathered by MTurk workers with those made by TREC experts [4, 5]. The study showed that the agreement between each worker and the TREC benchmark was not high when measured individually, but it increased when the workers were grouped. Furthermore, workers tended to disagree with the original assessors slightly more when the document is relevant. The overall outcomes of their experiment suggests that crowdsourcing is a cheap, quick, and reliable alternative for relevance assessment. Blanco et al. [14] have shown that the use of crowdsourcing to collect relevance judgments is reliable and allows repeatable IR evaluation. Grady and Lease [49] analysed human factors for crowdsourcing assessments in order to measure the accuracy of the results, time required for the assessment, and the costs. Kazai et al. [68, 67] studied the effect of several factors (e.g., pay, effort, worker qualifications, motivations, and interest) on obtaining relevance assessments by MTurk workers within the INEX Book track.

Other researchers have attempted to measure relevance in various ways, including graded judgments [91], preference-based judgments [9], and multidimensional ones [139]. Collecting relevance labels through crowdsourcing has also been used in practice, for example in the TREC blog track [91], or in the judgment task of the TREC Crowdsourcing

Track [112]. Tonon et al. [122] used used standard and crowdsourced relevance judgements to extend existing test collections with additional judgments over time. Eickhoff et al. [40] conducted a large-scale comparative study aimed to investigate the performance of traditional tasks designs and a game-based alternative that is able to achieve high quality at significantly lower pay rates, facing fewer malicious submissions. Zuccon et al. [140] used a crowdsourcing platform as a means of engaging study participants. Their experimental methodology captures user interactions and searching behaviours at a lower cost and within a shorter period than traditional laboratory-based user studies.

3.6 Evaluation metrics

Evaluation metrics have an essential role in effectiveness evaluation of IR systems, and are used to compare systems' results with the documents' relevance value, which are annotated by human experts. IR researchers proposed many evaluation metrics; in a 2006 survey, Demartini and Mizzaro [35] collected and discussed more than 50 metrics, only considering those oriented to system effectiveness. Today more than 100 are listed [8]. A perfect metric suitable for all possible scenarios does not exist. Vice versa, every metric is characterised by its own advantages and limitations and can result appropriate for evaluating different aspects of retrieval behaviour [16, 100].

Every researcher has to be conscious of the features of the metrics that he/she uses; choosing an inadequate metric might lead to wasting research efforts improving systems toward a wrong target. Several approaches for metric classifications and comparisons have been proposed, these define axioms [84, 85] or constraints [7] that the metrics can respect or not. Metrics classifications are made on the basis of the upheld axioms and constraints: ideally similar metrics should uphold similar axioms/constraints. The following sections describe briefly the main evaluation metrics.

3.6.1 Precision, Recall and F–Measure

The simplest metrics are set-based measures. *Ret* is the set of documents retrieved by a system, and *Rel* is the set of relevant documents. The most known metrics are:

- *Precision*: fraction of documents retrieved by the IR system that are actually relevant:
$$Precision = \frac{|Rel \cap Ret|}{|Ret|} = P(Rel|Ret) \tag{3.1}$$
- *Recall*: fraction of relevant documents that are retrieved:
$$Recall = \frac{|Rel \cap Ret|}{|Rel|} = P(Ret|Rel) \tag{3.2}$$
- *F–Measure*: measure that combines precision and recall by computing their harmonic mean:
$$F\text{–}Measure = \frac{2 * Precision * Recall}{Precision + Recall} \tag{3.3}$$

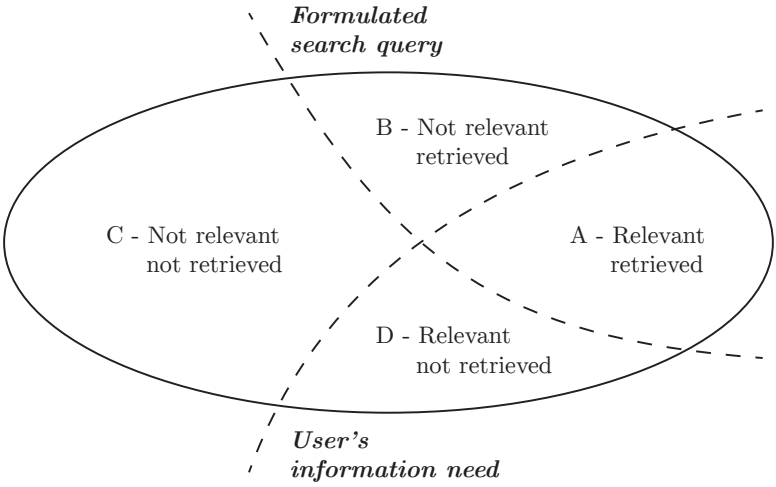


Figure 3.1: Rel ret

F–Measure is also used in a weighted version, computed by multiplying precision and recall respectively for α and $(1 - \alpha)$.

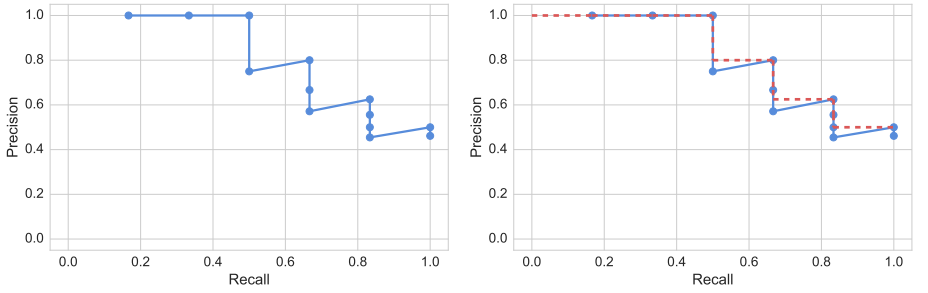
3.6.2 Precision Recall Curve

For each query, an IR system produces a rank in which precision and recall can be computed. The computed scores of precision and recall are combined into a "saw-tooth" graph (Figure 3.2a shows an example) called the precision-recall curve. This chart is then interpolated generating a step function similar to those shown in Figure 3.2b. The precision is measured on the interpolated curve at a fixed number of recall levels (usually the levels are 11, precisely 0, .1, ... , .9, 1). This operation is essential since the number of the steps of a curve depends on the query that generated it and precisely on the number of relevant documents retrieved. At this point, all the curves have the same number of recall levels and then a mean curve can be computed. As shown in Figure 3.3a, the mean curve looks smoothed and gives an overview of the overall system performance to multiple queries. As shown in Figure 3.3b precision-recall curves allow the comparison between multiple IR systems.

3.6.3 Average Precision (AP)

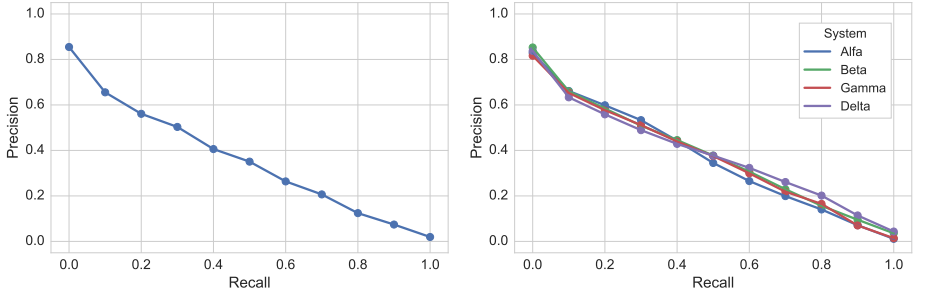
Average Precision (AP), computes the average of the precision value obtained by relevant documents which have been retrieved by an IR system.

$$AP = \int_0^1 p(r)dr \tag{3.4}$$



(a) (b)

Figure 3.2: Example of Precision - Recall curve, not interpolated (a) and interpolated (b).



(a) (b)

Figure 3.3: (a) Mean of Precision - Recall curves over multiple queries. (b) Comparison between the Precision - Recall curves of multiple systems.

3.6.4 Mean Average Precision (MAP)

AP operates on a single user need (query). For obtaining a wider overview of a system, we can compute AP, multiple times, over a set of queries, and then calculate the mean of all the AP scores. This is exactly what is done by Mean Average Precision (MAP). If Q is a set of queries, $Q = \{q_1, ..., q_n\}$:

$$MAP = \frac{\sum_{q=1}^{|Q|} APq}{n} \tag{3.5}$$

3.6.5 Geometric Mean Average Precision (GMAP)

Geometric Mean Average Precision (GMAP) is a variant of MAP. GMAP considers the geometric mean instead of the arithmetic mean. Proposed by Robertson in [100], it is specifically used to emphasise the lower end of the average precision scale, in order to shed light on poor performance of search engines. Given a set of n queries $(q1, ..., qn)$ and their relative AP scores $(AP_{q1}, ..., AP_{qn})$, the measure is computed as:

$$GMAP(AP_{q1}, ..., AP_{qn}) = \sqrt[n]{\prod_{i=1}^{n} AP_{qi}} \tag{3.6}$$

A single "zero" score in the formula is sufficient for reset the whole measure. To solve this problem Robertson proposes an alternative formula [100]:

$$GMAP(AP_{q1}, ..., AP_{qn}) = \exp\left(\frac{\sum_{i=1}^{n} \log(AP_{qi} + \epsilon)}{n}\right) - \epsilon \tag{3.7}$$

The two formulas are equivalent apart from a small constant ϵ which allows to solve the problem of smoothing. Researchers often use together MAP and GMAP in their evaluations since differences among their scores can be interesting and informative for the analysis [99].

3.6.6 P@N

Precision at N is the precision score obtained considering only the first N documents retrieved by a system. A "common practice" is to set N as the number of the documents shown by a web engine on the first result page.

3.6.7 R-Precision

Given R number of relevant documents returned by a system, we can compute *R-Precision* as precision value at the R-th position of the ranked list of the retrieved documents. For example, if given a query, a system retrieves five relevant documents, but only three of those are in the top five positions of the rank, then the R-Precision

score is 0.6 (3/5). If all the first documents returned by a system are relevant, R-Precision is one, and if all of those are not relevant, R-Precision is zero. P@5 and P@10 are common choices.

3.6.8 Normalized discounted cumulative gain (NDCG)

Discounted Cumulative Gain (DCG) is a metric that can work with more than two ordered relevance levels. It assumes that the most relevant documents for the user should be placed in the higher positions of the rank produced by an IR system. DCG measures the gain that a document brings to the user in relation to its logarithmically-discounted relevance degree (smoothing) which depends on its position on the rank of the systems results. Given a particular position p on the rank, and let rel_i be the relevance value of the document in position i of the rank, DCG can be computed as:

$$DCG_{@}p = rel_1 + \sum_{i=2}^{p} \frac{rel_i}{\log_2(i)} \tag{3.8}$$

Different queries submitted to a system directly affect in a different way both the number of relevant documents retrieved and the total length of the rank. For this reason, DCG in often normalised, becoming Normalized Discounted Cumulative Gain (NDCG), which is computed with the following formula:

$$NDCG_p = \frac{NDCG_p}{IDCG_p} \tag{3.9}$$

where IDCG is the ideal DGC, that consists in DGC computed on the documents sorted into the ideal order (the most relevant documents in higher positions of the rank, gradually followed by all the others).

3.6.9 Expected reciprocal rank (ERR)

Expected Reciprocal Rank (ERR) [24] is a metric, based on the "cascade" user model, that overcomes a limitation of NDCG represented by the fact that a document in a given position of the rank always has the same gain and discount independently of the documents shown above it. ERR implicitly discounts documents placed below very relevant documents in the rank. Also, ERR is defined as the expected reciprocal length of time that the user will take to find a relevant document.

3.7 Conclusions

In this chapter, we have presented IR and the concept of relevance, and discussed the importance of evaluation by means of both user studies and test collections. We have focused on the second one and on relevance assessment, a process which allows the gathering of relevance judgments by human experts. We have discussed some initiatives

for evaluation. Subsequently, we focused on the time factor in assessment activity. We then discussed measurement scales. Then the chapter discussed the relevance assessment through crowdsourcing. Finally, we have discussed the main IR evaluation metrics.

II

Two experiments (and some preliminary result)

4

Time factor in relevance assessment

This chapters discuses the time factor in the relevance assessment. Section 4.1 introduces the reasons that make the time factor interesting to analyse. Section 4.2 presents four research questions. Sections 4.3, 4.4, 4.6 and 4.5, discuss four experiments aimed to study how the time factor affects the relevance assessment process. Section 4.7 concludes the chapter discussing findings and limitations of this study. The material presented in the chapter is based on a published work [87].

4.1 Introduction

In this chapter we look at how to make individual crowdsourced relevance judgments more time efficient and, consequently, more cost-effective. We specifically analyse the time dimension involved in making relevance judgments by crowd workers and we study how much time workers need to make a judgment and what is the effect of reducing available judging time on the judgment quality. By identifying the optimal time needed to perform a high quality judgment, our proposed techniques allow to reduce the overall cost of generating an IR evaluation collection by means of crowdsourced relevance judgments, while maintaining an adequate quality.

We run extensive experiments by comparing different strategies to reduce the time available for a relevance judgment in crowdsourcing platforms. We experimentally show the trade-off of time/quality and which are the best choices, at the same cost, between asking for many quick judgments for the same topic-document pair as compared to fewer well-though judgments. We additionally compare different ways to enforce time constraints for judging tasks and their effect of judgment quality. Our main contributions are:

- A study on the time crowd workers take to make relevance judgments and what

are the effects of training and worker characteristics as well as the impact of topics and documents on judgment time and quality.

- An analysis of how judgment quality degrades by reducing/increasing the time available to make a judgment in a crowdsourcing setting.
- A comparison of alternative approaches to enforce time constrains on crowd judges (e.g., maximum time available vs. exact time to be spent on the task).
- The identification of the best allocation of monetary budget in a crowdsourcing platform for collecting relevance judgments, focusing on the trade-off between many quick judgments or fewer slow judgments.

4.2 Research questions

In order to optimize the cost of collecting crowdsourced relevance assessments, we assume a basic model, where the monetary cost c of an assessment for a topic is simply the product of the time t taken to judge each document, the number of judgments per document j, the number of documents n, and the reward r assigned for a judgment:

$$c = t * j * n * r. \tag{4.1}$$

By considering the reward r and the pool n constant, the total cost is then affected by the time and the number of judgments per document. We do not consider other parameters like the time to read the topic/query, the time to express the judgment, the time to switch to a new document and/or topic, etc. as we expect them not to be different over workers and documents. We measure the quality of a judgment by its agreement with editorial judgments (we will see some specific agreement measures below).

On the basis of this model, we can frame the following four research questions:

1. How much time t do crowd workers take to judge the relevance of a document if no time constrain is set?
2. What is the minimum amount of time t we can ask crowd workers to take in judging the relevance of a document? Does the judgment quality decrease when less time is available to make a judgment?
3. Which type of timeout is the most appropriate to solicit effective judgments? An *exact-time* timeout, where the document is shown for a certain amount of time and the judgment cannot be expressed before, or a *maximum-time* timeout, where the judgment can be expressed also before the expiration?
4. With a fixed budget c, what is the best trade-off between time available for a judgment t and number of judgments collected per document j? Is it better to ask for more judgments done quickly (higher j, lower t) or less judgments done with more time available (higher j, lower t)?

To answer these research questions, we ran a battery of four experiments, described in detail in the following four sections.

4.3 Experiment one

4.3.1 Aims

The time spent by a relevance assessor to perform a judgment varies quite a lot according to existing literature. For example, [130] report an average of 100 seconds for documents in the AQUAINT collection; [136] report that most expert judges take up to 140 seconds while crowd workers up to 90 seconds. The aim of this first experiment, that addresses RQ1, is to measure the range of time spent by individual workers in our setting and use the results to identify appropriate thresholds for timed relevance judging tasks.

4.3.2 Experimental design

We used five TREC-8 topics (403, 418, 420, 427, 448), selected based on the availability of multi-graded relevance judgments, of at least 2 documents per relevance level, and to avoid topics which are not anymore timely at present time. For each topic, we randomly selected eight documents having different lengths and different relevance levels. We selected two documents for each level, one long and one short. We define document length based on word count and uniformly sample documents sorted by length.

In our first experiment E1, each worker (we recruited highest quality workers as provided by CrowdFlower) was shown the TREC topic (title, description, and narrative) and, after an initial test question that verified that the topic had been understood, had the task of judging the relevance of the eight documents, shown in a permuted order such that any document appeared exactly 5 times in each of the eight positions. A worker could use as much time as he/she wanted on each document before going to the next one, and he/she was allowed to perform other units, but only on different topics (i.e., workers could not re-judge the same topic, to avoid a learning bias). We collected 5 judgments for each document in each position, so for each topic 5*8=40 workers were needed in this first experiment. Finally, we ran the experiment twice, for both India and U.S. based workers independently, for a total of 3200 judgments.

4.3.3 Results

4.3.3.1 Judgment time

Figure 4.1 shows the task execution time distribution. As expected, in both runs, many workers took little time to complete the judgment with a tail of very long execution times. Figure 4.2 shows the distributions of time spent by individual workers over each topic and document position. We can observe that for some topics (e.g., 420) the first document to be judged takes more time because of learning effects. On average there is little delay for the first document to be judged as compared to the others. Around 97%

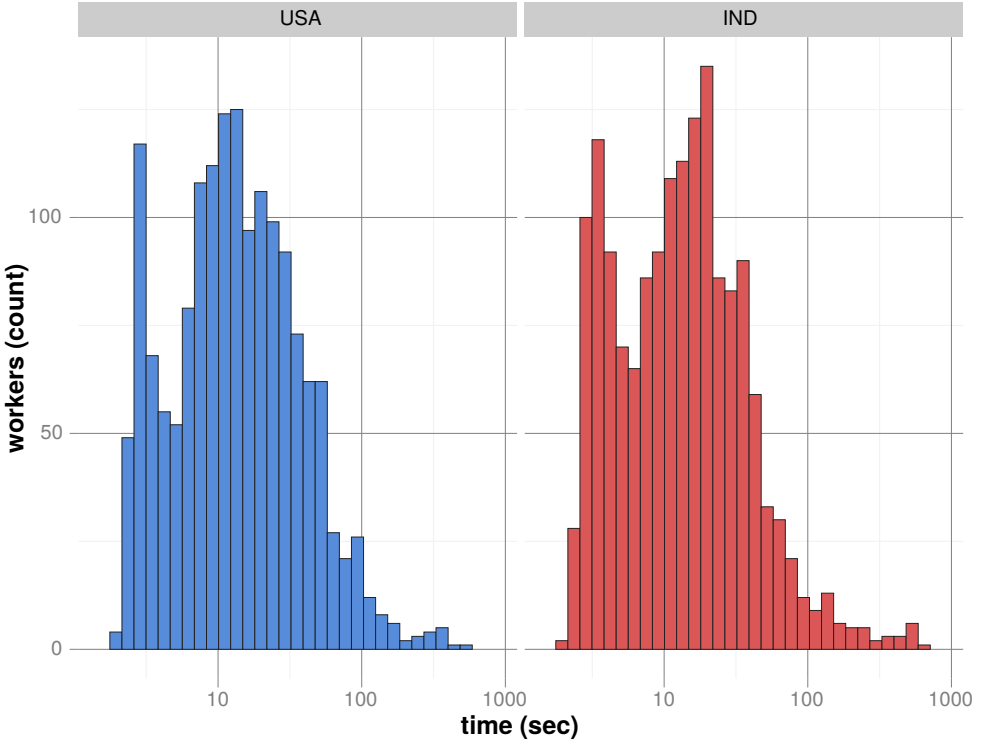


Figure 4.1: E1: Time required by workers to judge a document (log scale).

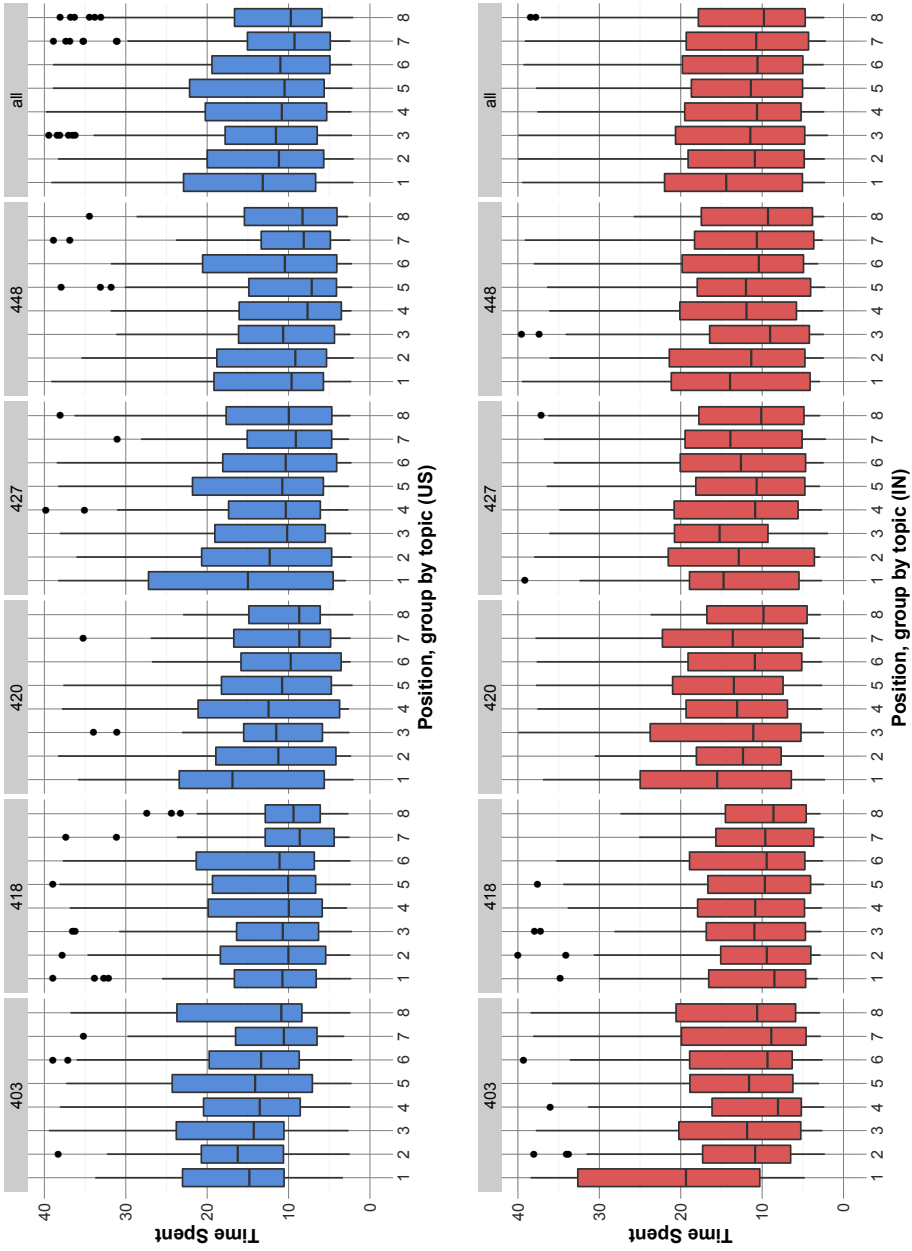


Figure 4.2: E1: Time spent by individual workers to judge document relevance broken down over topics and judging order for US based workers (in blue) and India based workers (in red).

of the recorded times are in the range between 2 and 100 seconds; about 80% are below 35 seconds and about 60% spent 5–35 seconds (see also Table 4.1).

Table 4.1: E1: Distributions of time spent for US- and India-based workers.

	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
US	1.98	6.62	13.00	24.20	27.20	580.00
IN	1.90	5.46	13.30	25.40	25.70	634.00

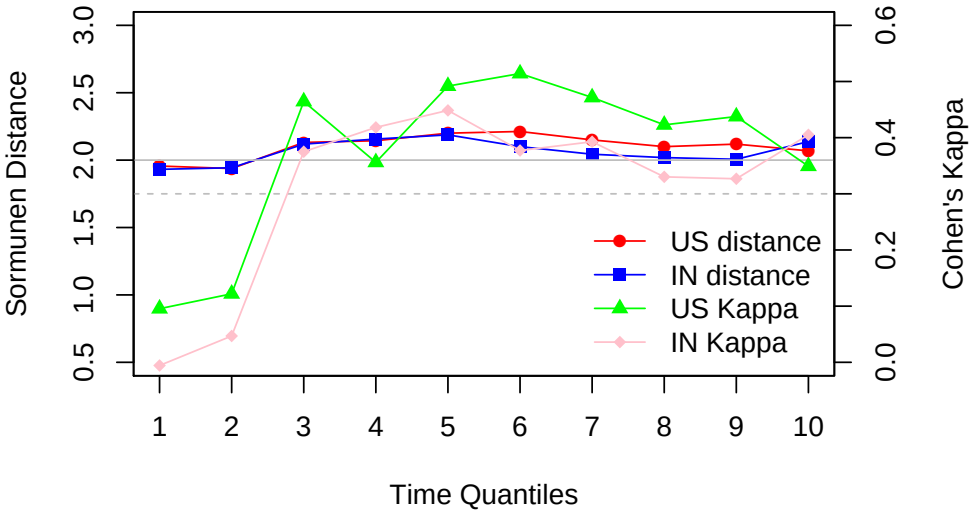
We found no correlation between judging time and topic or document length (measured as number of chars or words, or with the ARI readability index [109]). This is consistent with the findings of [9] who looked at these dimensions within the TREC 2012 Crowdsourcing track collection. We also did not observe correlation of time spent with relevance level, nor with agreement rates with Sormunen and TREC judgments.

4.3.3.2 Judgment Quality

We now study the quality of the relevance judgments obtained from the crowd with no time constrains imposed on the task. We measure quality as the level of agreement (measured with Cohen's Kappa) against Sormunen's 4-level judgments. We also measure agreement against binary TREC judgments, in three ways: by considering TREC relevant as H and nonrelevant as N, and by thresholding the 4 levels into 2 levels (both N-MRH and NM-RH). As additional measure of quality we also considered the average distance of the category selected by the worker to Sormunen's category (i.e., accuracy) which showed analogous results. The same metrics have been previously used to measure crowd judgment quality [98, 66]. We compute such quality metrics both using the crowd judgments considered individually as well as with judgments aggregated together following the standard majority vote approach.

Figure 4.3 shows how quality changes over execution time for individual workers. The horizontal axis shows the deciles of time spent (deciles values are shown in the table below the plot). The vertical axis reports the quality, measured both with Sormunen distance and Cohen's Kappa. Note that for our dataset (documents with uniform distribution of relevance) a random assessor would obtain an average distance of 1.75 (dashed horizontal line). The two quality measures consistently show that the highest quality values are obtained in the central part of the curve, corresponding to around 5–50 seconds, and especially 5–25 seconds (values in bold in the table). This is consistent with Figure 4.2 and confirms that the time interval 5–30 seconds covers most worker activities. In the following experiments we will focus on such time interval to identify the execution time that leads to highest quality judgments.

We also look at judgment quality variations over the document order presented to workers. Table 4.2 shows that, as expected, Kappa values of judgments aggregated by majority vote are higher than those of individual judgments. We also note that the first two judgments are typically of lower quality, most likely because of learning effects.



	0%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
U.S.	2.0	3.2	**5.1**	7.6	10	13	17	**23**	32	51	580
IN	1.9	3.4	**4.5**	7.0	9.9	13	17	**22**	31	46	630

Figure 4.3: E1: Variation of quality over time spent by individual workers (binned by deciles). Quality is measured both as distance from Sormunen label and Cohen's Kappa. The two quality measures have Kendall's correlations of 0.82 (U.S.) and 0.87 (IN).

4.4 Experiment two

4.4.1 Aims

Based on the analysis of the time taken by crowd workers to judge the relevance of a document (see Section 4.3), we now study the effect on judgment quality of reducing the time available to crowd workers to look at the document to be judged. To understand which is the minimum amount of time required to perform relevance judgments by crowd workers (RQ2) we designed the following experiment (E2).

4.4.2 Experimental design

We display a document to crowd workers for a predefined amount of time and ask for a best effort relevance judgment. Given the results from E1 (Section 4.3), we set the following *timeouts* (i.e., time after which the document disappears and a judgment has to be made): 3, 7, 15, 30 seconds as we observed best quality to be around the

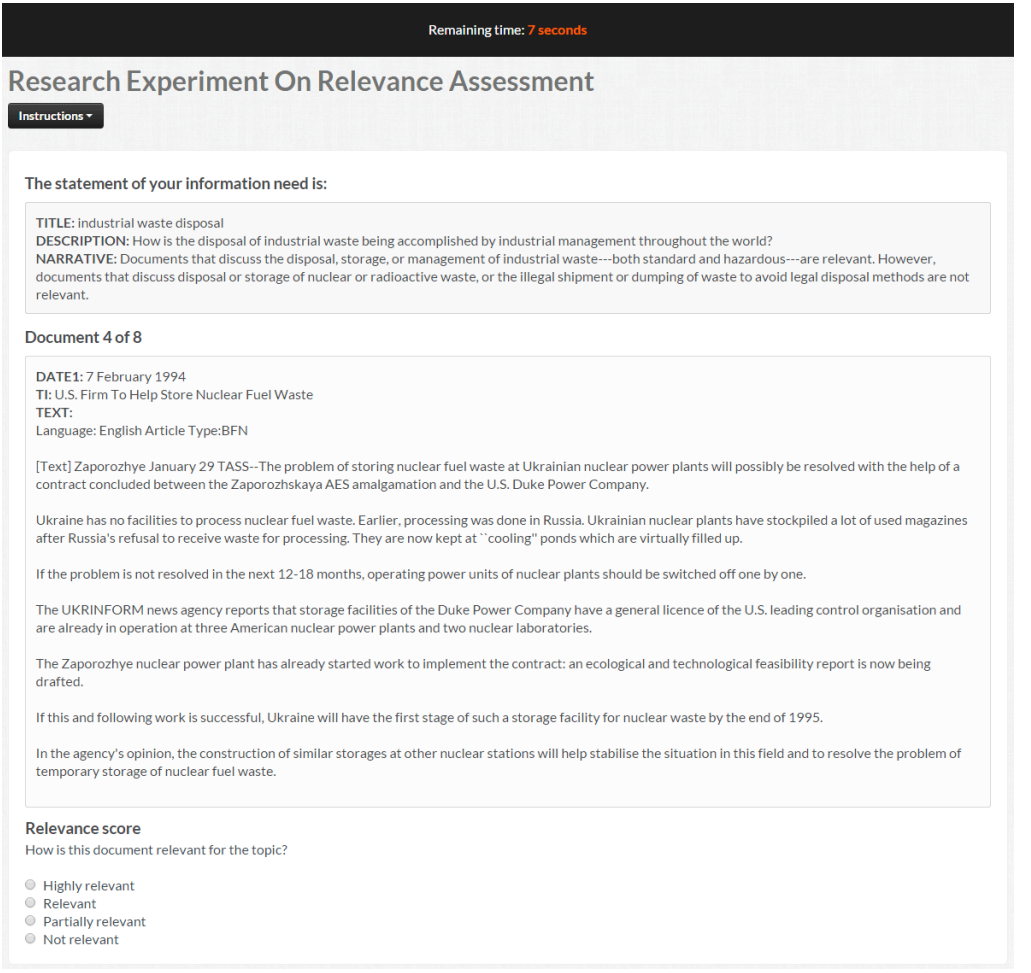


Figure 4.4: The task's interface seven seconds before the document disappears. After that, the worker can still express their judgment, but they are not allowed to see the document anymore.

Table 4.2: E1: Agreement, measured as Cohen's Kappa, between Workers and Sormunen (W-S) and Workers and TREC (W-T) over different document timeouts and positions, for both U.S. and India based workers, both individual (I) and aggregated (A). W-T Kappa is computed with three different weights: (i) default; (ii) `NM` into 0 and `RH` into 1; and (iii) `N` into 0 and `MRH` into 1. For comparison, S-T Kappa is 0.59, 0.55, and 0.77 with the three weights, respectively.

Position		p1	p2	p3	p4	p5	p6	p7	p8	AVG
U.S.										
W-S	I	.26	.34	.36	.46	.33	.49	.44	.43	.39
	A	.41	.36	.47	.65	.46	.63	.54	.47	.50
W-T	I	.20	.26	.27	.29	.24	.29	.32	.24	.26
	A	.30	.31	.35	.41	.34	.35	.42	.21	.34
W-T, `NM-RH`	I	.16	.22	.22	.21	.18	.25	.26	.22	.21
	A	.25	.27	.34	.34	.37	.34	.34	.15	.30
W-T, `N-MRH`	I	.28	.33	.41	.51	.34	.47	.48	.37	.40
	A	.52	.43	.52	.69	.38	.57	.78	.38	**.53**
IN										
W-S	I	.32	.39	.42	.22	.24	.30	.36	.31	.32
	A	.38	.57	.60	.44	.38	.34	.67	.38	.47
W-T	I	.21	.25	.29	.20	.14	.29	.31	.26	.24
	A	.26	.41	.36	.33	.25	.37	.56	.34	.36
W-T, `NM-RH`	I	.19	.27	.26	.17	.14	.28	.28	.26	.23
	A	.20	.51	.33	.30	.25	.43	.68	.27	.37
W-T, `N-MRH`	I	.30	.33	.39	.26	.24	.32	.44	.37	.33
	A	.36	.55	.59	.59	.43	.36	.71	.55	**.52**

time interval $5 - 30$ seconds.[1] In this experiment, workers are allowed to complete the judgment before the timeout (i.e., we set the *maximum time* to judge) and can proceed to the next document. Figure 4.4 shows the interface that the workers used for expressing their judgements. We use five TREC-8 topics (405, 408, 415, 416, 421). We selected 6 highly relevant, 6 relevant, 6 marginally relevant, and 6 not relevant documents per topic. We ask each worker to judge the relevance of 8 documents in total where the first two documents (a long and a short one) are displayed for 30 seconds, the following two documents for 15 seconds, other two for 7 seconds and, finally, two documents for 3 seconds. We set decreasing timeout values in order to let workers get prepared for shorter document visualization times and to learn how to grasp relevance signal in limited time. Any other timeout order would penalize short timeouts even further. Note also that even if some learning effect is present during the judgment of

[1]We can interpret judgments completed in less than 5 seconds as spam and tasks completed in more than 30 seconds as done by multi-tasker workers.

the first document (see Figure 4.2), having a first timeout at 30 seconds allows enough time to perform their judgment for at least 80% of workers (Figure 4.3). We also point out that the topics (and documents) used in E2 are different from E1, for two reasons: we needed topics having 6H, 6R, 6M, 6N documents for each topic, and we wanted to avoid possible biases from workers participating in both E1 and E2. Figure 4.5 summarizes the experimental design of E2.

4.4.3 Results

Figure 4.6 shows how judgment time varies given a set timeout. We can observe that both in the case of 3 and 7 second timeouts workers judge relevance *after* the document display time is expired (red horizontal line). We call the time between the timeout and the judgment *latency*. In the case of 15 and 30 second timeout, the judgment happens, in the median case, before the timeout.

Table 4.3: E2: Cohen's Kappa values between Workers and Sormunen (W-S) and Workers and TREC (W-T) over different document timeouts and positions, for both U.S. and India based workers (see also Table 4.2).

Timeout(sec)	30		15		7		3	
Position	p1	p2	p3	p4	p5	p6	p7	p8
U.S.								
W-S	.43	.44	.52	**.53**	.51	.51	.35	.43
	.43		**.52**		.51		.39	
W-T	**.37**	**.37**	.34	.31	.34	.32	.21	.32
	.37		.32		.33		.27	
W-T, NM-RH	.28	.34	**.35**	.17	.30	.32	.22	.32
	.31		.26		**.31**		.27	
W-T, N-MRH	.37	.37	.30	.42	**.44**	.30	.23	.36
	.37		.36		**.37**		.29	
IN								
W-S	.39	.42	**.44**	.42	.35	.38	.36	.34
	.40		**.43**		.36		.35	
W-T	**.31**	.25	.28	.28	.26	.28	.29	.19
	.28		**.28**		.27		.24	
W-T, NM-RH	**.31**	.27	.29	.27	.24	.24	**.31**	.19
	.29		.28		.24		.25	
W-T, N-MRH	.28	.28	.25	.34	.30	**.41**	.30	.25
	.28		.30		**.36**		.28	

Table 4.3 shows the Cohen's Kappa agreement values of aggregated judgments over 5 crowd workers with Sormunen judgments on a four level relevance scale and with

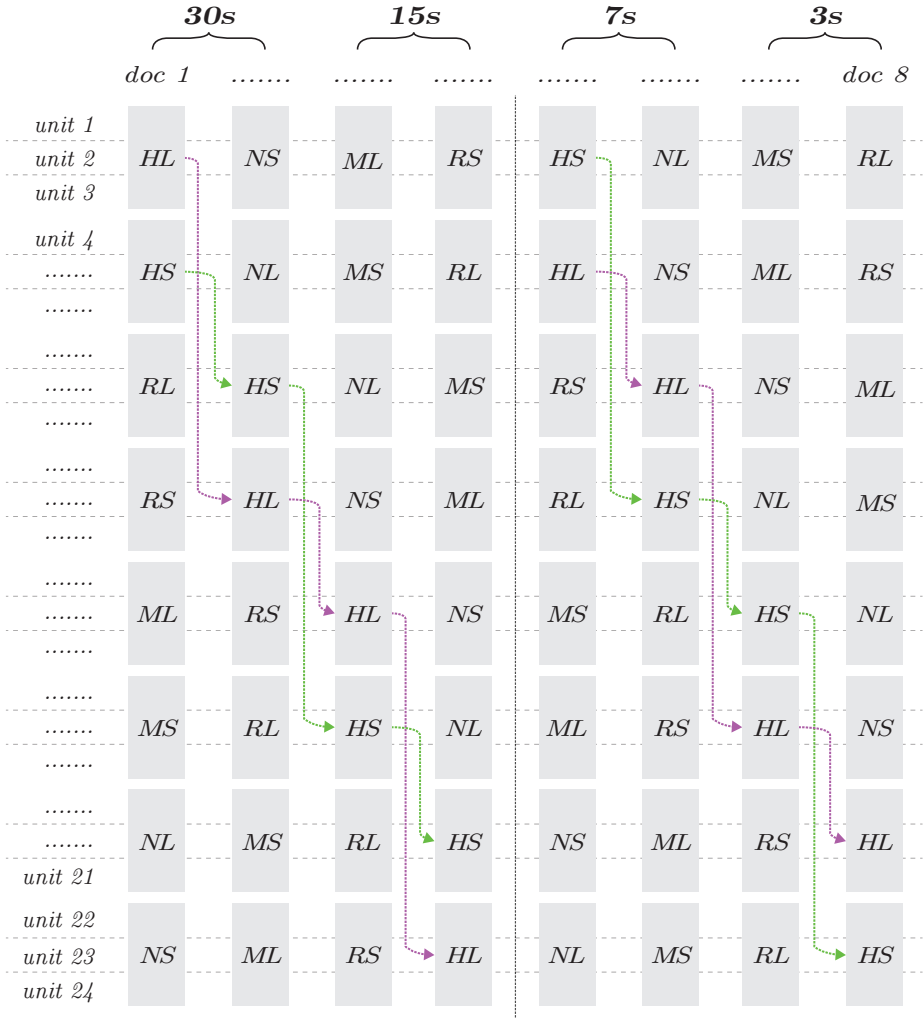


Figure 4.5: E2: Experimental design. Each worker is presented with 2 documents (one short and one long) over 4 different timeout values (30, 15, 7, and 3 seconds). Documents presented to the worker are different (i.e., 8 different documents in each row). Documents appearing in each of the 8 positions are different (i.e., 24 different documents in each column). The 8 presented documents are grouped in four pairs, each with a different relevance value (based on the NMRH scale of Sormunen). In each row, the choice of the documents that are presented to the worker is random inside each of 3 documents blocks (thus respecting the constraints).

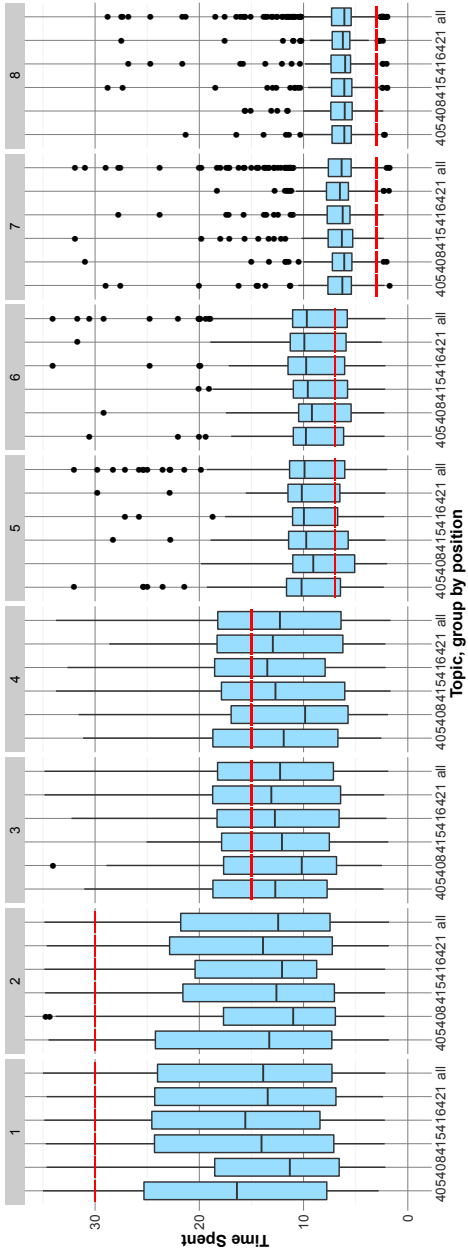


Figure 4.6: E2: Time spent by individual US based workers, breakdown over topics and document positions. The red lines show the timeouts.

TREC binary judgments. We can observe that in most cases 3 and 7 seconds are not enough to make a relevance judgment as agreement values are consistently lower than for other timeout values and the judgments are expressed well after the deadline (see the two rightmost panels in Figure 4.6). We can also see that US workers tend to have higher quality judgments than workers based in India.

When comparing agreement rates at 30 and 15 seconds we can see better results at 15 seconds. This can be explained by the learning bias in position 1-2 as confirmed by the E1 results (see Table 4.2) and of the following experiment presented in Section 4.6. It has also to be noted that the quality difference between 15 and 30 seconds is not statistically significant (t-test $p = 0.39$). Compared to TREC assessments, after transforming aggregated crowd judgments into binary ones by means of a threshold, we can see that the agreement of crowd workers is lower than that of Sormunen's assessments.

4.5 Experiment three

When comparing judgment quality obtained with crowdsourcing (measured by Cohen's Kappa agreement with the original relevance assessments) we can observe that using some sort of timeout leads to better quality judgments as compared to judgments obtained with unlimited amount of time (Section 4.3). E2 results in Table 4.3 show that best judgment quality is obtained with 30s for TREC judgments (Kappa = .37) and with 15s for Sormunen judgments (Kappa = .52). Average Kappa values with TREC and Sormunen was 0.34 and 0.50 respectively in E1 (Table 4.2).

4.5.1 Aims

E2 results are not directly comparable with E1 results for two reasons: the documents used in E1 are different from those used in E2, and there could be a learning effect in E2 which could make the comparison biased by document positions and different time slots used. To allow for a direct comparison with E1 and to understand which timeout value leads to better worker performance, we run two modified versions of E2 with 15 and 30 seconds available to workers.

4.5.2 Experimental design

The experimental design of E3 is very similar to that of E2 but with two important differences: the documents used are the same used in E1 and the time available to each worker for viewing the document is the same for all the 8 documents (fixed to 15 or 30 seconds). Like in E2, workers are free to judge a document before or after its disappearance thus not using all the time made available to them (*maximum-timeout*). We use the same quality checks as for previous experiments (i.e., topic understanding question and high-quality workers from the platform).

Table 4.4: E3: Average of Cohen's Kappa values measuring the agreement between Workers and Sormunen (W-S) and Workers and TREC (W-T) over 8 documents for both U.S. and India based workers.

Country	U.S.		India		Average	
Timeout(sec)	30	15	30	15	30	15
W-S	0.47	0.49	0.48	**0.50**	0.48	0.50
W-T	0.33	**0.38**	0.32	0.35	0.33	0.37
W-T, NM-RH	0.31	0.37	0.31	**0.38**	0.31	0.38
W-T, N-MRH	0.47	**0.64**	0.49	0.50	0.48	0.57

4.5.3 Results

Table 4.4 shows the agreement between workers and the assessment of TREC and Sormunen obtained for experiment E3. For both TREC and Sormunen comparison, the table shows that workers performances are always better for 15 than for 30 seconds and the difference is even statistically significant (t-test, $p < 0.01$) for TREC values.

Comparing E1 and E3 quality levels, we observe that, when looking at Sormunen judgments, in E1 the agreement between workers and gold standard is .47 for Indians and .5 for Americans while looking at TREC judgments, in E1 American and Indian workers agree with the gold standard respectively at .28 and .24 (see Table 4.2). In E3 (Table 4.4), the average values for 15 seconds (rightmost column) are always higher. We can thus conclude that the introduction of timeouts in crowdsourced relevance judgment tasks is also beneficial in terms of judgment quality. This does not yet completely answer RQ3; we will come back to this issue at the end of the next section.

4.6 Experiment four

4.6.1 Aims

Given the results so far, we now want to understand, given a fixed budget, what is the most effective way to collect crowdsourced relevance judgments within time-bound tasks. We study the trade-off between collecting many judgments with a very short timeout as compared to very few judgments with long timeout (RQ4). With the assumption that the cost is computed, according to our model in Equation (4.1), by the actual workforce time spent on tasks (i.e., we pay workers for the time they spend on our tasks), we aim at finding the best timeout t and number of assignments values j for a fixed monetary budget c.

4.6.2 Experimental design

In order to compare the judgment quality over different trade-offs, we crowdsourced a number of different timeout/assignment combinations with the same total monetary

budget for a number of topic-document pairs. Table 4.5 shows the ten time/judgments combinations (with a constant cost c according to Equation (4.1)).

Table 4.5: E4: Different timeouts (rounded) and number of judgments, with the same monetary cost of a total of 150 seconds.

Timeslot(sec)	6	7.9	10	13.7	16.7	21.5	25	30	37.5	50
Assignments	25	19	15	11	9	7	6	5	4	3

To also address RQ3, as compared to timeouts used in E2 and E3 (i.e., *maximum-time*), in E4 we instead use the *exact-time* alternative. That is, we set a time after which the document disappears and workers have to make a relevance judgment, but we do not allow workers to judge and proceed before the given time, if they wish to do so.

4.6.3 Results

Figure 4.7 shows the judgment quality of different timeouts using 3 assignments in each case. We can observe that the quality increases as more time is available to the judge (thus, at a higher cost), but after 30s there is a sort of "plateau" with no noticeable increment of quality (whereas the cost c increases noticeably according to our model in Equation (4.1)); indeed, the quality seems to decrease after a "peak" at 30s. In Figure 4.8 the cost is kept constant (the number of judgments for each timeout is the one in Table 4.5), and we use all the available judgments at each timeout. Here the peak at around 25-30s is even more clear. Thus, since the cost c is constant for each timeout level, the highest quality is obtained with $t \in [25, 30]$s. Comparing workers based in US and India, we can see from Figure 4.8 that a budget gives significantly (t-test $p < 0.01$) better quality judgments when spent giving workers based in India 25s timeouts and to workers based in US 30s timeouts. Figure 4.9 shows the effect of the topic. The 25-30s range results in the highest quality across (most of the) topics; note that our collection includes both easy and difficult topics, with clearly lower quality judgments for topics 427 and 420.

This lower quality may be a sign of ambiguity of the documents contained in that topic, which seems confirmed by the calculation of the Intraclass Correlation Coefficient (ICC) of workers' judgments (Table 4.6). Topic 427 is, in fact, the topic which displays the lower agreement between workers themselves. Manually looking at the two most difficult topics we notice that they both contain the most technical concepts (427: 'UV

Table 4.6: E4: Intraclass Correlation Coefficient of the workers judgments over the different topics, calculated for the assignments defined in Table 4.5.

Topic	403	418	420	427	448
Median ICC	.41	.39	.28	.21	.27

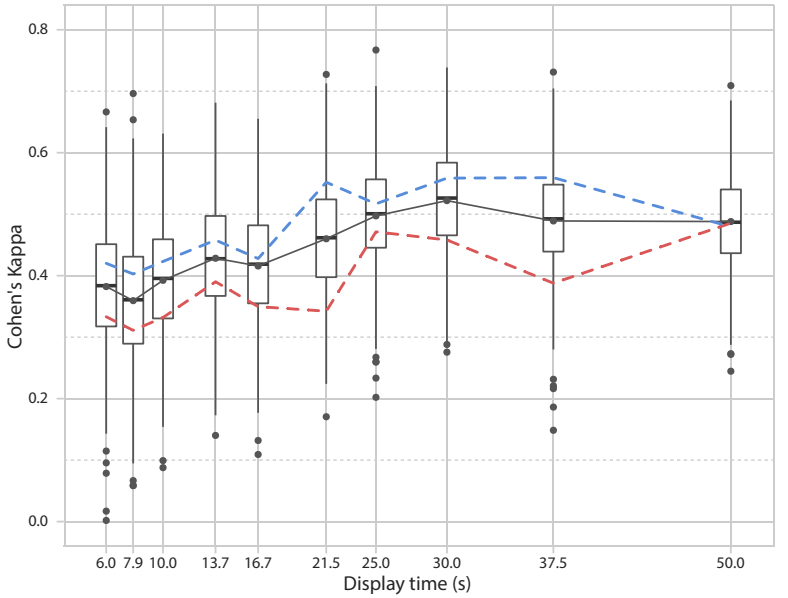


Figure 4.7: E4: Judgment quality (measured as Cohen's Kappa) over the 10 display time conditions with the same number of workers (three)

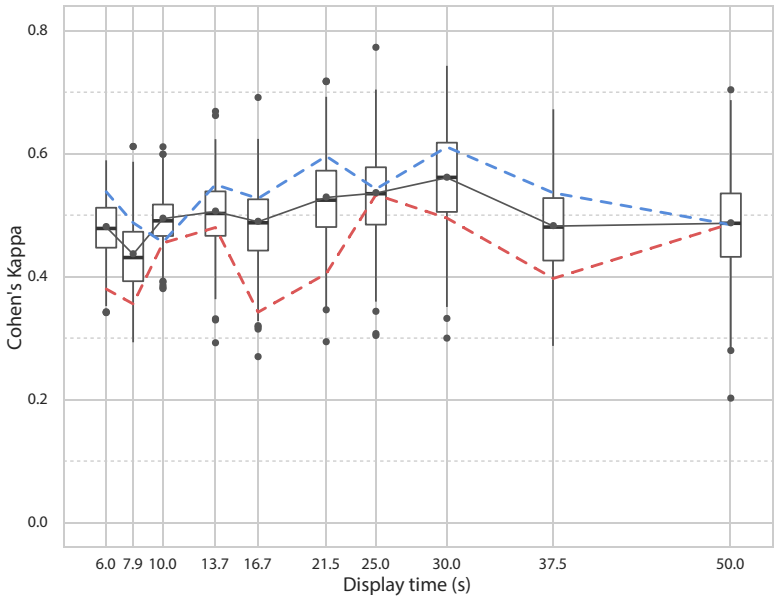


Figure 4.8: E4: Judgment quality (measured as Cohen's Kappa) over the 10 display time conditions with the same cost

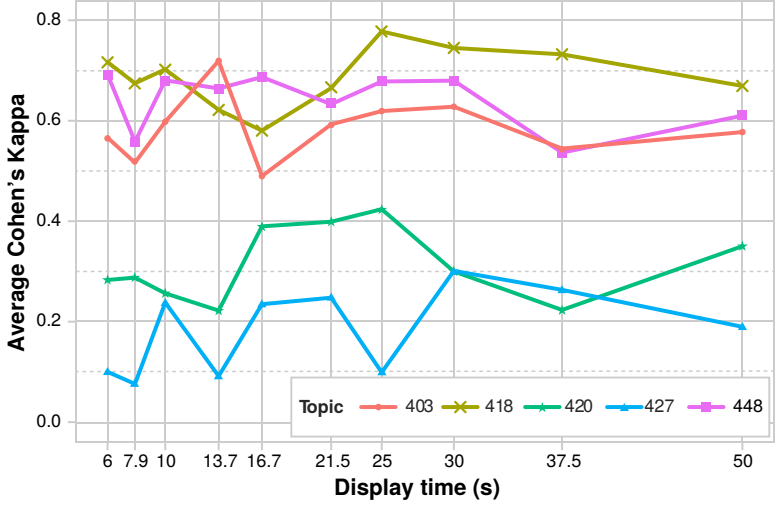


Figure 4.9: Judgment quality (measured as Cohen's Kappa) over the 10 display time conditions breaking down on the five topics

Table 4.7: E4: Intraclass Correlation Coefficient over different timeouts, calculated for the assignments defined in Table 4.5.

Timeslot(sec)	6	7.9	10	13.7	16.7	21.5	25	30	37.5	50
ICC	.28	.29	.31	.25	.30	.32	.35	.30	.35	.41

Table 4.8: E4: Cohen's Kappa values over the ten timeslots, for U.S. and India based workers.

Timeslot(sec)	6	7.9	10	13.7	16.7	21.5	25	30	37.5	50
U.S										
W-S	.54	.49	.46	.55	.53	.60	.54	**.61**	.54	.48
W-T	.37	.30	.26	.41	.37	.42	.39	.35	**.44**	.39
W-T, NM-RH	.34	.31	.34	.37	**.43**	.40	.40	.34	.40	.34
W-T, N-MRH	.61	.47	.30	**.74**	.43	.69	.55	.50	.67	.55
IN										
W-S	.38	.36	.46	.48	.34	.40	**.53**	.50	.40	.49
W-T	.32	.18	.33	.32	.23	.29	.36	**.47**	.35	.34
W-T, NM-RH	.29	.21	.37	.37	.21	.31	.44	**.52**	.37	.34
W-T, N-MRH	.53	.10	.43	.43	.39	.44	.47	**.65**	.54	.46

damage, eyes' and 420: 'carbon monoxide poisoning') and have strict criteria of relevance specified in the narrative field. This indicates that time-constrained judgments may be more appropriate for general topics as compared to more technical/scientific ones which are probably more difficult for the average crowd worker to assess.

Table 4.7 shows ICC values over different timeout levels confirming that workers agree most when having at least 25s available for the judgment task.

As a last result, we go back to RQ3 as promised at the end of Section 4.5.3. By comparing the W-S Kappa values in Table 4.8 (or in Figure 4.8) with the Kappa values in Tables 4.2 and 4.4 we observe that:

- We have already seen in Section 4.5.3 that maximum-time timeouts (used in E2 and E3) seem effective, since there is some increase in Kappa values from Tables 4.2 to Table 4.4.

- However, exact-time timeouts, used in E4, do cause a clearly higher increase: the Kappas obtained in E4 (Table 4.8) are clearly higher than those of E1 and E3, both at the 30s timeout, and at 13.7s and 16.7s (the closest values to 15s).

4.7 Discussion

In this section we summarize our main findings on how time constrains affect relevance judgments in a crowdsourcing setting across all the different experiments we performed.

4.7.1 Findings

We have observed in both E1 and E2 that there is a learning effect at the beginning of the judgment tasks which implies that the first couple of judgments a worker completes will be of lower quality, compared to the following ones. From E1 and E2 we have observed crowdsourced relevance judgments taking more than 30s tend to show lower quality. Indeed, the best quality in E4 is obtained making workers judge relevance within a [25,30] seconds interval. We have noted that topic difficulty (measured by judgment quality as well as by ICC) does not impact the best timeout to be used (Figure 4.7 and Table 4.6 for E4). We measured judgment quality using both Cohen's Kappa as well as the distance between crowd and editorial judgments (both by TREC and Sormunen). Results obtained with both metrics concur. We also generally observed that workers based in US provided, in our experiments, better quality judgments as compared to those based in India (see Tables 4.3 and 4.8).

Answering RQ3 we observed that, given a certain timeout value, the best option is to have the worker to view the document for all the allocated time (i.e., not allowing him/her to proceed to the next document earlier) and to limit that time (E3 vs. E4).

Comparing the average Cohen's Kappa values between crowdsourced relevance judgments and judgments collected by Sormunen [115] we can observe that the quality obtained in our first experiment E1 (Section 4.3) where workers could take any amount of time to complete the judgment task as it is traditionally done in crowdsourced relevance judgments, is on average 0.5 (Table 4.2) for US workers as compared to the average Kappa of 0.61 obtained with a timeout of 30 seconds for US workers in E4 (Table 4.8).

In all experiments, an interesting and recurrent aspect emerged with respect to Cohen's Kappa values between workers and TREC (W-T): very often we observed higher quality work judgments when thresholding them as `N-MRH` instead of `NM-RH`. This shows that workers are similar to TREC assessors in separating strictly not relevant documents to those they judged being somehow relevant.

4.7.2 Limitations of the study

Our work looks at optimizing relevance judgment time in crowdsourcing settings. While workers in the crowd are different [69] and show different behaviors [70], our approach does not take individual differences into account. We rather aim at finding a general strategy that works well over all workers. As future work, we will investigate the effectiveness of personalized approaches to time-bound relevance judgment by evaluating adaptive timeouts over different worker types and expertise levels.

In our work we used majority vote as a technique to aggregate crowd judgments. While more sophisticated techniques to aggregate crowd labels exist (e.g., [56, 125, 110])

in this work we do not focus on obtaining the highest label quality but rather on observing the effect on quality degradation due to given time constrains in completing the task. For the same reason, we only include basic quality checks (e.g., topic understanding questions, use of high quality workers provided by the platform, etc.) when collecting data from the crowd. Thus our quality measurements indicate lower bounds and can easily be improved by combining other techniques to increase quality still reducing the cost of creating IR evaluation collections. Previous research has shown that even if assessor agreement levels are lower in crowdsourcing as compared to editorial judgments, IR evaluation results are still reliable and experiments are repeatable if we consider IR system ranking correlation levels [14].

4.8 Conclusions

In this chapter we have addressed the problem of limiting the time available for crowdsourced relevance judgment tasks. This is an important problem as controlling the judgment time allows for faster data collection (i.e., avoids the common *starvation* effect when batches of tasks do not finish as some workers take very long time to complete) as well as limits the cost of evaluation collection creation if workers are paid for the time they spent executing judgments.

We performed extensive experiments using standard test collections to evaluate the quality of crowdsourced judgment; as compared to editorial judgments in a number of controlled experimental settings to understand the effect of limited time on quality. Results clearly show that limiting the time to perform a relevance judgment brings benefits both in terms of cost (and this was expected) as well as of quality (and this was unexpected). We observed that the best timeout value to be used lies in the interval of $25 - 30$ seconds and does not depend on topic, document, or crowd. Our findings are key for those researchers using crowdsourcing for the creation of large-scale IR evaluation collections as they can better control the creation cost still obtaining high quality annotations thanks to our proposed techniques.

Acknowledgments We would like to thank all the crowd contributors who participated to this study. This work was partially supported by the *UK EPSRC grant number EP/N011589/1.*

5

Using magnitude estimation for relevance assessment

The material presented in the chapter is based on a published work [88].

5.1 Introduction

Relevance is an important concept in information retrieval (IR), and relevance judgments form the backbone of test collections, the most widely-used approach for the evaluation of IR system effectiveness. Document relevance judgments are typically made using ordinal scales, historically at a binary level, and more recently with multiple levels [133]. However, despite its importance, operationalizing relevance for the evaluation of IR systems remains a complicated issue; for example, when using an ordinal scale it is unclear how many relevance categories should be chosen [121].

Magnitude estimation is a psychophysical technique for measuring the sensation of a stimulus. Observers assign numbers to a series of stimuli, such that the numbers reflect the perceived difference in intensity of each item. The outcome is a ratio scale of the subjective perception of the stimulus; if a magnitude of 50 is assigned to one stimulus, and 10 to another, then it can be concluded that the two items are perceived in a 5:1 ratio of sensation [96]. While initially developed for the measurement of the sensation of physical stimuli such as the intensity of a light source, magnitude estimation has been successfully applied to the measurement of non-physical stimuli including the usability of interfaces [92].

Being able to derive ratio scales of subjective perceptions of the intensity of stimuli, magnitude estimation may be a useful tool for measuring and better understanding document relevance judgments. The application of magnitude estimation in the IR field has been limited to the consideration of the relevance of carefully curated abstracts returned from bibliographic databases [41], and our own small-scale pilot study [108, 86].

While suggestive, these studies have been limited in terms of demonstrating the broader utility of the approach, or its direct application to IR evaluation. In this work, we investigate the larger-scale application of magnitude estimation to document relevance scaling, reporting on a user study over 18 TREC topics and obtaining judgments for 4,269 documents. This is also the first work to consider the direct application of magnitude estimation to the evaluation of IR systems, and to rely on crowdsourcing workers to gather a large amount of magnitude estimation relevance assessments. Specifically, we consider four research questions.

RQ1. Is the magnitude estimation technique suitable for gathering document-level relevance judgments, and are the resulting relevance scales reasonable with respect to our current knowledge of relevance judgments?

RQ2. Is crowdsourcing a viable method to gather robust magnitude estimation values? Crowdsourcing might exacerbate some of the potential complications of using magnitude estimation, such as the subjective scale used by each judge, or the difficulty in using a ratio scale without proper training. Do the workers agree with previous judgments, and are redundant workers required to obtain agreement with previous judging methods?

RQ3. How does IR system evaluation change when ratio scale magnitude estimation relevance judgments are used to calibrate the gain levels of the widely-used nDCG and ERR evaluation metrics, compared to using arbitrarily set gain values for a pre-chosen number of ordinal levels?

RQ4. Can magnitude estimation relevance scores provide additional insight into user perceptions of relevance, into actual gain values, and into individual gain profiles?

We also investigate the qualitative comments that users of the magnitude estimation relevance judging approach were required to make to justify their scores, and present an analysis of the relationship between comments, scores and source documents.

In the next section, background material and related work are presented. Details of our user study and experimental methodology are provided in Section 5.2, and the first descriptive results are presented in Section 5.3. The full analysis and discussion of our results, corresponding to the four research questions, are detailed in Sections 5.4 to 5.7.

5.2 Experimental Methodology

In this section we describe the details of the experiments that were carried out to crowdsource the required relevance judgments using magnitude estimation.

5.2.1 Topics and Documents

Document-level relevance assessments have traditionally been made using binary or multi-level ordinal scales. To compare magnitude estimation judgments to these approaches, we selected a set of search tasks and documents from the TREC-8 ad hoc

collection, which studies informational search over newswire documents. The TREC-8 collection includes binary relevance judgments made by NIST assessors [132]. Subsequently, Sormunen [115] carried out a re-judging exercise where a subset of topics and documents were judged by a group of six Master's students of information studies, fluent in English although not native speakers, on a 4-level ordinal relevance scale: `N`–not relevant (0); `M`–marginally relevant (1); `R`–relevant (2); `H`–highly relevant (3).

The 18 *search topics* used in our study were the subset of TREC-8 topics which also have Sormunen ordinal judgments available. They are listed in the first column of Table 5.1. The *documents* for which we collected magnitude estimation judgments were the set of the top-10 items returned by systems that participated in TREC-8. This gives a total of 4,269 topic-document pairs, of which 3881 have binary TREC relevance judgments available, and 805 of which have Sormunen ordinal judgments available. The number of documents for each topic is shown in Table 5.1 (2nd column).

5.2.2 User Study

We carried out a user study through the CrowdFlower crowdsourcing platform during December 2014 and January 2015. The experimental design was reviewed and approved by the RMIT University ethics review board. Participants were paid $0.2 for each task *unit*, defined as a group of magnitude estimation judgments for 8 documents in relation to one topic. The number of units for each topic is shown in Table 5.1 (3rd column).

5.2.2.1 Practice task

After agreeing to take part in the study, a participant was shown a first set of task instructions, including a brief introduction to the experiment and explanation of the magnitude estimation process. Since magnitude estimation may not be familiar to participants, they were first asked to complete a practice task, making magnitude estimations of three lines of different lengths, shown one at a time. If they successfully completed this practice task (success was defined as assigning magnitudes such that the numbers were in ascending order when the lines were sorted from shortest to longest), participants were able to move on to the main part of the experiment. The full instructions shown to participants for the practice task are included in Appendix A.

5.2.2.2 Main task

The main task of the experiment required making magnitude estimations of the relevance of documents. Participants were informed, by means of a second set of instructions, that they would be shown an information need statement, and then a sequence of eight documents that had been returned by a search system in response to the information need statement, presented in an arbitrary order, and that their task was to *indicate how relevant these documents appear* in relation to the information need.

For the main task, the title, description and narrative of a TREC topic were first displayed at the top of the screen. After reading the information need, participants had to respond to a 4-way multiple-choice question, to test their understanding of the

Topic	No. docs	No. units	H_k	N_k	Back 0	Back 1	Back ≥ 2	Fail 0	Fail 1	Fail ≥ 2
402	278	460	LA111689-0162	FBIS3-10954	452	8	0	426	18	16
403	111	182	LA092890-0067	LA071290-0133	178	4	0	120	19	43
405	214	354	LA061490-0072	FBIS3-13680	347	6	1	322	20	12
407	212	350	FT921-6003	FR940407-2-00084	346	3	1	324	9	17
408	188	310	FT923-6110	LA062290-0070	306	4	0	266	16	28
410	212	350	FBIS4-64577	FBIS4-44440	346	4	0	326	10	14
415	179	295	FBIS3-60025	FBIS4-10862	289	4	2	255	23	17
416	174	287	FBIS4-49091	LA112590-0107	286	1	0	267	12	8
418	243	402	LA102189-0167	FT924-6324	394	8	0	372	15	15
420	164	270	LA121590-0108	LA112690-0001	266	4	0	256	4	10
421	342	567	FT941-428	LA073189-0033	554	11	2	537	16	14
427	195	322	FT943-5736	LA080590-0077	315	7	0	214	52	56
428	253	419	FT943-9226	FBIS3-20994	412	6	1	396	13	10
431	203	335	FBIS3-46247	FT944-5962	333	2	0	296	27	12
440	264	437	FT942-3471	LA020589-0074	427	10	0	397	19	21
442	408	677	LA011390-0057	FT923-4524	672	5	0	629	25	23
445	210	347	FT924-8156	LA031989-0092	334	11	2	334	10	3
448	419	695	LA080190-0139	FBIS3-16837	687	8	0	652	21	22
Sum:	4269	7059			6944	106	9	6389	329	341
Percentage:					98	2	0	90	5	5

Figure 5.1: Topics, number of documents, number of units, H_k, N_k, number of back buttons usage, number of failures in the (c) and (d) checks, as detailed in the text.

topic. The test questions focused on the main information concepts presented in the topic statements, and were intended to check that participants were engaging with the task, and as a mechanism to remove spammers. Participants who were unable to answer the test question correctly were unable to continue with the task.

Next, participants were presented with eight documents, one at a time. For each document, the participant was required to enter a magnitude estimation number in a text box displayed directly below the document, and a brief justification of why they entered their number into a larger text field (Figure 5.2). They were then able to proceed to the next document. A back button was available in the interface, with participants being advised that this should only be used if they wish to correct a mistake; under 3% of submitted jobs included use of this feature (details on the occurrences of clicks on the back button are shown in Table 5.1, where the number of units with zero, one, or two or more back button clicks are shown).

After entering responses for eight documents, the task was complete. Participants were able to complete further tasks for other topics, up to a maximum of 18 tasks, as they were not able to re-assess the same topic.

5.2.2.3 Magnitude estimation assignments

Participants were instructed to assign ME scores as follows. *You may use any numbers that seem appropriate to you – whole numbers, fractions, or decimals. However, you may not use negative numbers, or zero. Don't worry about running out of numbers – there will always be a larger number than the largest you use, and a smaller number than the smallest you use. Try to judge each document in relation to the previous one. For example, if the current document seems half as relevant as the previous one, then assign a score that is half of your previously assigned score.* While some applications of magnitude estimation use a fixed modulus (specifying a particular number that is to be assigned to the first stimulus that is presented), this has been found to promote clustering of responses and the potential over-representation of some numbers [96]; we therefore allowed participants to freely choose their own values. The complete instructions that were shown to participants for both the practice and main tasks are included in Appendix A.

5.2.2.4 Document ordering

As explained above, each participant task *unit* required the judging of a set of eight documents for a particular search topic statement. The experiments used a randomized design, with documents presented in random order, to avoid potential ordering effects and first sample bias [96]. The document sets were constructed such that two of the documents in each set were a known ordinal `H` and `N` document for the topic; these documents (henceforth $\mathtt{H}_k$ and $\mathtt{N}_k$) were the same for all the participants working on the same topic (the TREC document identifiers for each topic are shown in Table 5.1). This was to ensure that each participant saw at least one document from the high and low ends of the relevance continuum. Ensuring that stimuli of different intensity levels are included in a task has been found to be important for the magnitude estimation

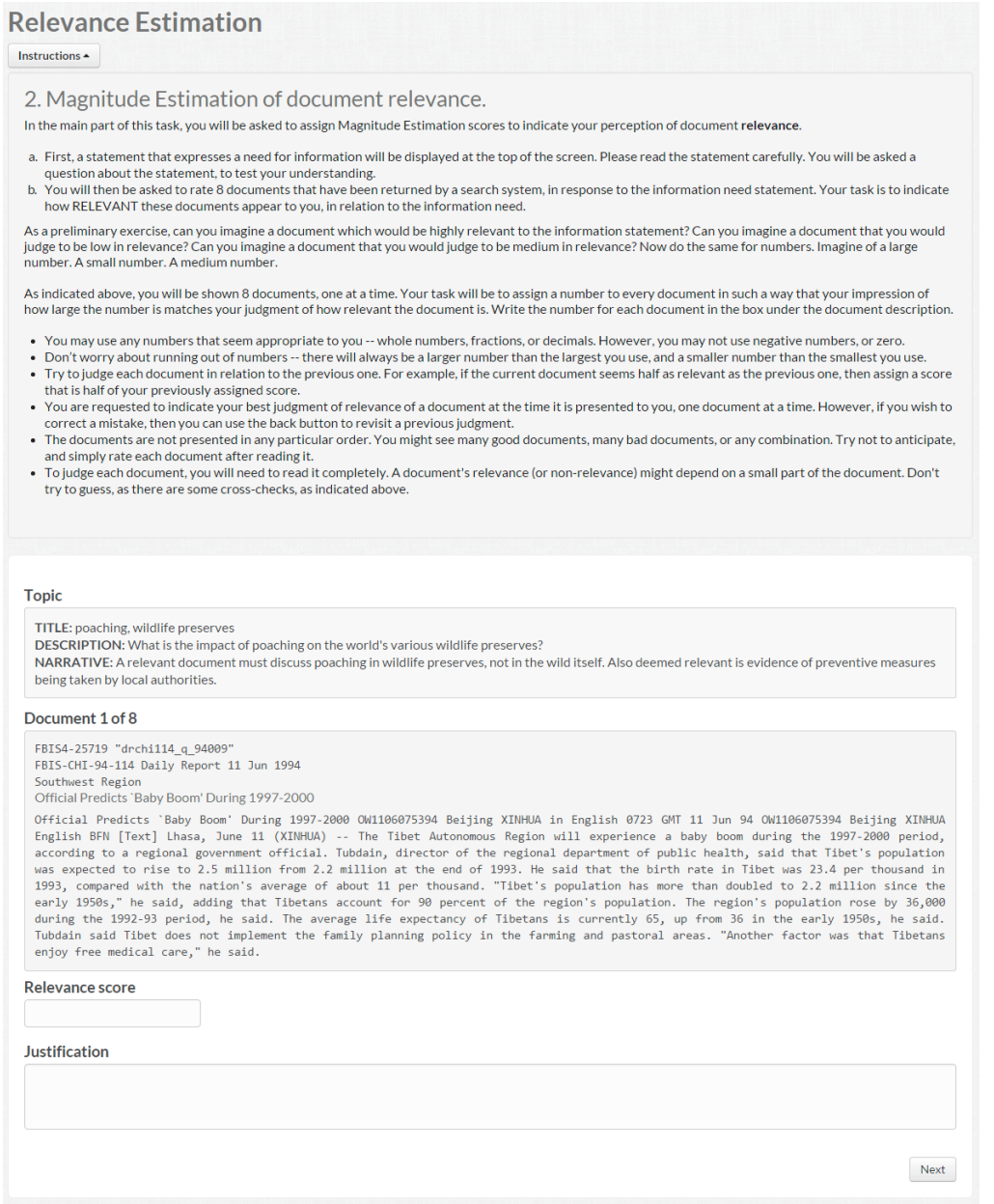


Figure 5.2: The interface used by workers to express the relevance of the documents.

process [48], and also has an impact on the score normalization process, described below. Moreover, including the two $\mathtt{N}_k$ and $\mathtt{H}_k$ documents with "known" ordinal relevance values enabled a further data collection quality control check: after judging all eight documents, participants who had assigned magnitude estimation scores for the $\mathtt{N}_k$ document that were larger than for the $\mathtt{H}_k$ document were not able to complete the task. In total, at least 10 ME scores were gathered for each topic-document pair.

5.2.2.5 Quality checks

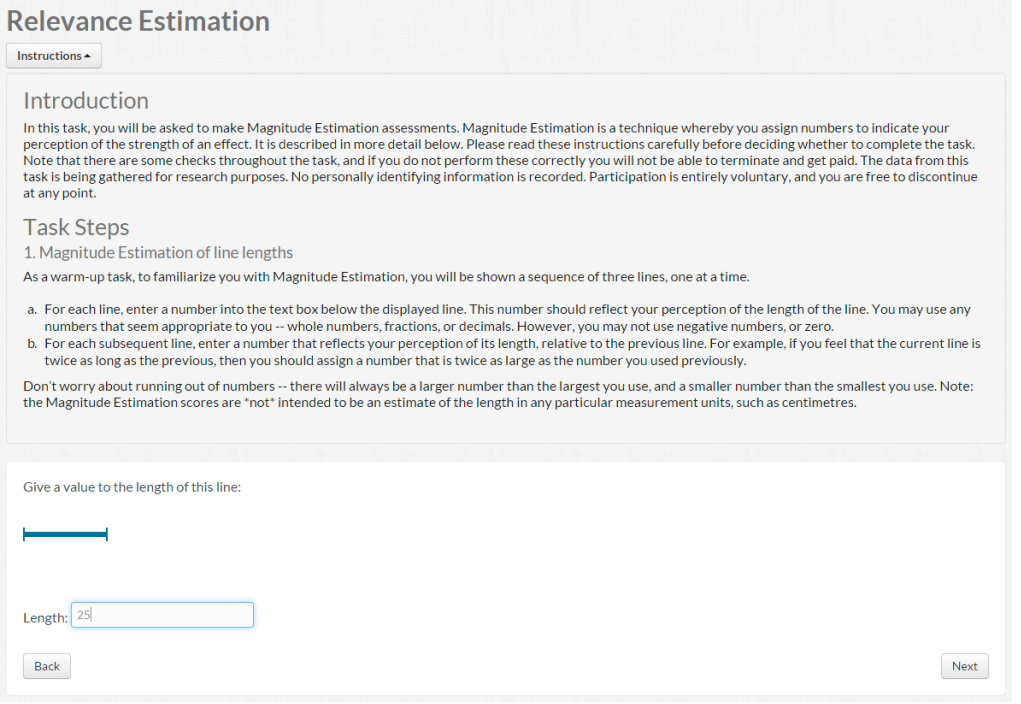


Figure 5.3: A screenshot of the practice task in which workers were asked to estimate sequentially the length of three line.

In total, four quality checks were included in the data gathering phase of our experiments:

(a) a practice task requiring magnitude estimation of line lengths (Figure 5.3);

(b) a multiple-choice test question to check a participant's understanding of the topic;

(c) a check that the magnitude estimations score for $\mathtt{H}_k$ was greater than that assigned to $\mathtt{N}_k$; and

(d) each participant had to spend at least 20 seconds on each of the 8 documents.

If any of (a) or (b) was unsuccessful, the participant could not continue with the task. They were allowed to restart from scratch, on a different unit and therefore on a randomly selected topic, if willing to do so; the same quality checks were applied again. If checks (c) or (d) were unsuccessful, the participant received the message "*Your job is not accurate enough. You can revise your work to finish the task*", and was allowed to use the back button and revise their previously assigned scores if they wished (the same two checks (c) and (d) were performed again in such a case); however, participants were not made aware which documents or scores were the "offending" responses. Less than 10% of units resulted in conditions (c) or (d) being triggered (see details in Table 5.1, where the number of eventually successful units with 0, 1, or two or more failed checks are shown), and as already mentioned the back button was used in less than 3% of units. Finally, there was a syntax check that the numeric scores input by the participants were in the $(0, +\infty)$ range. If any of the checks were not successfully completed, no data for that unit was retained. Based on these extensive quality checks, we did not carry out any further filtering once the data had been collected.

5.3 General Results and Descriptive Statistics

5.3.1 Crowd Judging

As detailed in Table 5.1, with 7,059 units comprised of 8 documents each, we collected more than 50,000 judgments in total, at a cost of around $\$1,700$ (CrowdFlower fees included). This is in the order of magnitude of \$0.4 for each document, which is broadly competitive when compared with the cost of TREC assessors or other similar crowdsourcing initiatives; for example, according to Alonso and Mizzaro [5, Footnote 2], gathering ordinal scale judgments cost around \$0.1 per document. However, the comparison is more favorable when considering that in our experiments 10 redundant judgments per document are collected, and the $\mathtt{N}_k$ and $\mathtt{H}_k$ documents receive multiple judgments, whereas in Alonso and Mizzaro's work only 5 redundant judgments were collected, and there was nothing similar to our $\mathtt{N}_k$ vs. $\mathtt{H}_k$ check. We return to the cost issues in more detail in Section 5.5.3, where the number of repeat judgments that are required for stable ME estimates is investigated.

In total, 1481 workers participated in our experiments. Since it was impossible for a worker to perform the task twice (or more) for each topic, each worker could complete between 1 (around 30% of the workers did so) and 18 (less than 1% of the workers) tasks. As is usual in similar crowdsourcing experiments, the distribution of the number of tasks for a worker resembles a power law, although with only 18 tasks at maximum it is difficult to be precise on that. Finally, workers tend to work in sessions: when they complete a topic they either leave after that, or sometimes they complete a subsequent topic straight away—they much more rarely leave and come back later to do another.

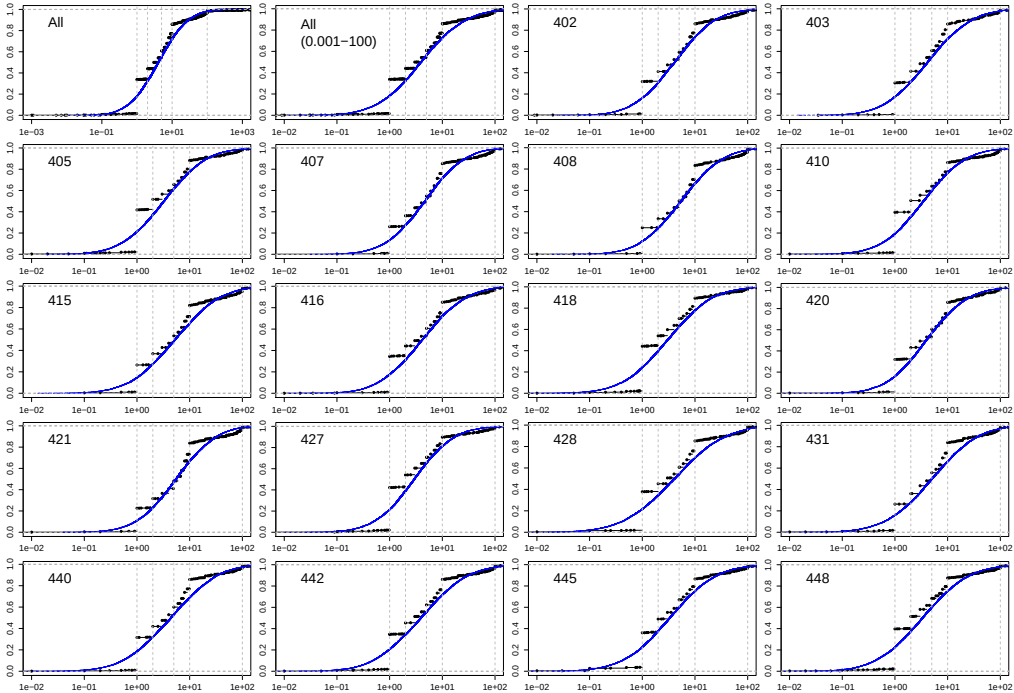


Figure 5.4: Cumulative distribution functions of raw ME scores for each topic (labelled in the top left of each panel) for all scores in the range 0.01 to 100. Black dots are gathered scores and the blue line in each panel is a fitted log-normal curve. The x axis is a log scale. Vertical dashed lines are at 1, 2, 5 and 10.

5.3.2 Score Distribution

Magnitude estimation scores should be approximately log-normal [90, 96]. Figure 5.4 shows the situation for our data, plotting the Cumulative Distribution Functions (CDFs) of: all the scores (top left), all the scores in the 0.001-100 range (which includes about 99.5% of the scores), and the scores for each topic. The continuous blue lines are fitted log-normal distributions with the same mean and standard deviation of the corresponding data. From the figure it is clear that the log-normal distribution is a reasonable approximation of the raw data. Differences are generally due to a *round number tendency* [96]: in general, people expressing magnitudes using a numeric scale tend to use some specific values more than others, for instance integer numbers and powers of ten. From the figure it can be seen that our workers tended to use especially 1, 2, 5, and 10 (the values where "steps" can be seen on the CDF curves — see the gray vertical dashed lines on the charts), and also quite a lot of 3, 4, 5, 6 7, 8, 9, 20, 30, 40, 50, 60, 70, 80, 90, and 100. The breakdowns on single topics show that topics do have an effect (for instance, in topics 418 and 421 the scores up to 1 included account for about 44% and

23% of the values respectively), although the general trend is very similar, as is again demonstrated by the fitted log-normal distributions. Topic means ranged from 7.9 to 12.9, and topic standard deviations ranged from 17.5 to 23.2.

5.3.3 Score Normalization and Aggregation

Magnitude estimation is a highly flexible process, with observers being free to assign any positive number, including fractions, as a rating of the intensity of a presented stimulus. The key requirement is that the ratio of the numbers should reflect the ratio of the differences in perception between stimulus intensities. As the stimuli are presented in a randomized order, and observers are free to assign a number of their choice to the first presented item, it is natural that different participants may make use of different parts of the positive number space. Magnitude estimation scores are therefore normalized, to adjust for these differences. Geometric averaging is the recommended approach for the normalization of magnitude estimation scores [48, 92, 96], and was applied in our data analysis. Recall that for a given search topic, participants made magnitude estimation judgments for groups of eight documents (a unit). To normalize the scores, first the log of each raw score, s_i, is taken. Next, the arithmetic mean of these log scores is calculated for each *unit* of 8 documents, and for the *topic* as a whole. Each individual log score is then adjusted by the unit and topic means, and the exponent is taken of the resulting quantity, giving the final normalized score

$$s'_i = e^{\log(s_i) - \mu(\log(unit)) + \mu(\log(topic))} \tag{5.1}$$

An alternative equivalent formalization is:

$$s'_i = s_i \cdot \frac{\gamma}{\gamma_u}, \tag{5.2}$$

where γ_u is the geometric mean for each unit and γ is the geometric mean for the topic as a whole. The log transformation is theoretically motivated by the fact that magnitude estimation scores of perceived stimulus intensities have been found to be approximately log-normal, which was the case in our data as shown in Section 5.3.2. Intuitively, normalization by geometric averaging means that the raw magnitude estimation scores are moved along the number line, both in terms of location and spread, and the individual differences in scale are nullified. For example, as shown by both Moskowitz [96] and McGee [92], if two judges assigned scores of {1, 2, 3, 4} and {10, 20, 30, 40} to the same set of four items, respectively, then the normalized values would be identical in the two cases, as can be easily verified by applying either Equation (5.1) or (5.2). Importantly, normalization through geometric averaging has the property of preserving the ratios of the original scores, the essential feature of the magnitude estimation process.

Other normalizations have been proposed, for example using the median or the arithmetic mean instead of the geometric mean (especially when zero is an allowed score, so the geometric mean cannot be used) Moskowitz [96]. Other approaches include carrying out an additional calibration task that is aimed at understanding the differences

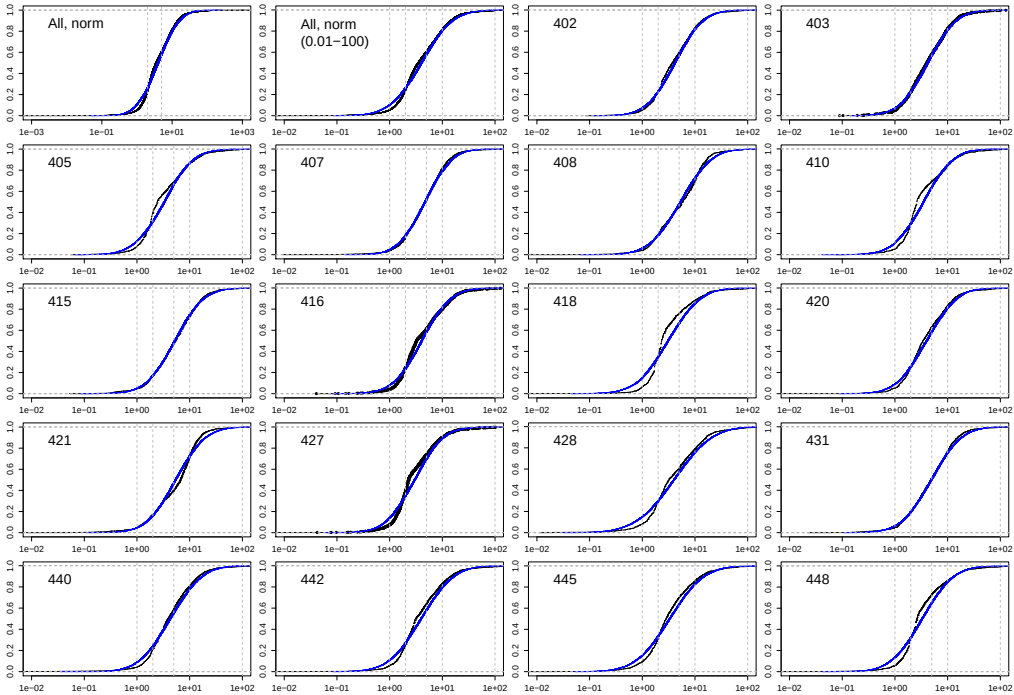


Figure 5.5: CDFs of normalized ME scores using the same style as Figure 5.4.

in individual scales; in our scenario, this might be implemented by exploiting the $\mathtt{N}_k$ and $\mathtt{H}_k$ documents, or in a more implicit way by relying on the maximal and minimal scores expressed by each worker.

Unless otherwise noted, all analysis reported in this paper uses magnitude estimation scores normalized by geometric averaging. Geometric averaging normalization is in principle adequate since, as we have discussed, our scores are approximately log-normal and they do not include zero. Figure 5.5 shows the distribution of scores after normalization which are similar to Figure 5.4. It can be noted how the normalization has the effect of smoothing the scores (the round number tendency disappears, since each score is moved by a quantity which is different for each worker) and of making them even more log-normal.

The problem of aggregating several ME scores for the same item also needs some attention. For each topic-document pair we collected at least 10 ME scores, by 10 different workers (many more for the $\mathtt{N}_k$ and $\mathtt{H}_k$ documents). In the following we will primarily use the *median* of the normalized ME scores for a topic-document pair, although again alternatives are available at this step, such as using the arithmetic mean or the geometric mean of obtained ME scores [96]. We briefly examine the effect of different normalization and aggregating schemes on our findings in Section 5.5.1.

5.4 Magnitude Estimation Relevance Judgments

Having obtained magnitude estimation scores for 4,269 topic-document pairs, we analyze the user-perceived relevance to answer the first research question RQ1, whether the magnitude estimation technique is suitable for gathering document level relevance judgments, and whether the resulting relevance scales are reasonable with respect to our current knowledge of relevance judgments.

The distribution of the median normalized magnitude estimation scores for each document are shown in Figure 5.6, aggregated across all 18 topics, and split by Sormunen ordinal relevance levels (left side of figure, levels are `N`, `M`, `R`, `H`, and `U`, the group of documents that were not judged by Sormunen), and by TREC binary levels (right side of figure, levels are 0, 1, and `U`, the documents that were not judged by TREC, as not all participating runs were included in the judging pool). Boxplots in this paper show the median as a solid black line; boxes show the 25th to 75th percentile; whiskers show the range, up to 1.5 times the inter-quartile range; and outliers beyond this range are shown as individual points.

There is a clear distinction between each of the four adjacent Sormunen levels (two-tailed t-test, $p < 0.002$), with the magnitude estimation scores on average following the ordinal scale rank ordering. The differences between the two TREC levels are also significant ($p < 0.001$), with the magnitude estimation scores on average again being aligned with the binary levels. This is strong evidence for the overall validity of the magnitude estimation approach.

Figure 5.7 shows the magnitude estimation score distributions for each of the 18 individual topics. Although there is some variability across topics, overall the figure confirms that the magnitude estimation scores are generally aligned with ordinal cate-

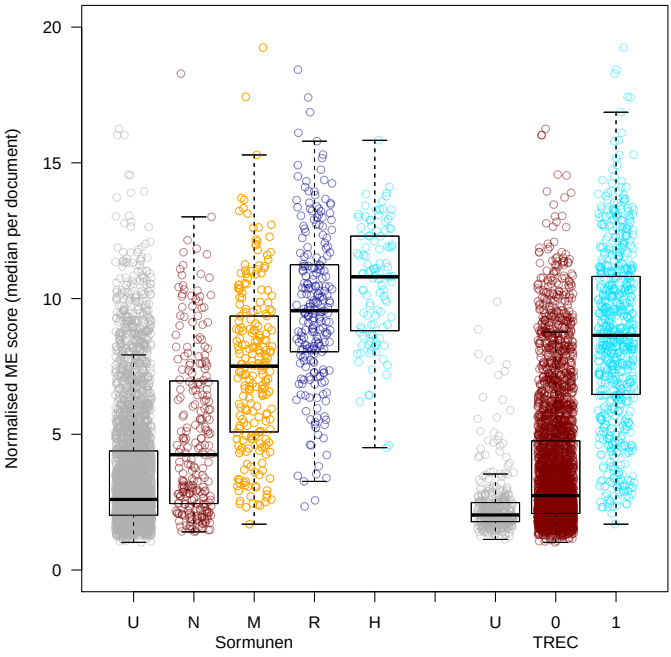


Figure 5.6: ME score distribution by Sormunen and TREC levels.

gories even when considering individual topics: the medians of the median magnitude estimation scores (the solid black lines) generally follow the ordinal categories, for all categories and for all topics (there are a small number of exceptions for some of the Sormunen adjacent categories in topics 403, 405, 410, 420, 421, and 440; there are no exceptions for non-adjacent categories, nor for TREC categories). Since for each topic there could potentially be 3 exceptions for adjacent categories, and 6 exceptions in general, plus one exception for the two TREC categories, the 6 exceptions found are out of $3 * 18 + 18 = 72$ possible cases when considering only adjacent categories, or out of $6 * 18 + 18 = 126$ cases when considering also non-adjacent categories. Such a limited fraction of exceptions (in the 5%-8% range) is further strong evidence for the validity of our approach, even at the single topic level.

Regarding the set of documents that were not judged by Sormunen (left-most boxes in Figure 5.6 and sub-plots in Figure 5.7), based on the magnitude estimation scores it can be inferred that the bulk of this class are likely to be non-relevant; however there are also instances that occur across the central parts of the marginal to highly relevant score distributions. There were also a handful of documents unjudged by TREC that seemed to be rated highly by our judges. While the overall distributions of magnitude estimation scores are strongly consistent with the ordinal and binary categories, there are also documents in each class where the ME scores fall into the central region of a different class. We therefore next investigate judge agreement.

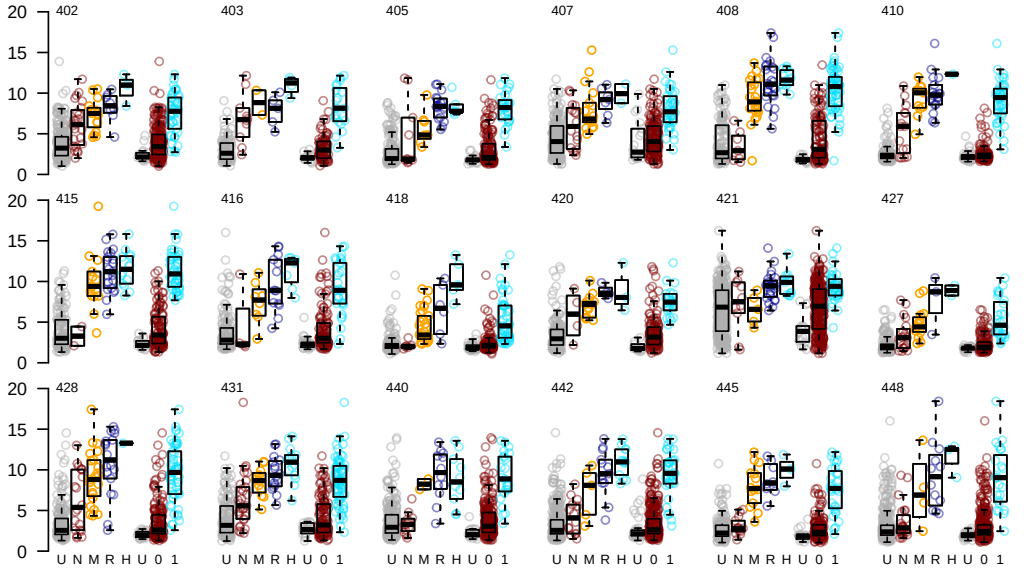


Figure 5.7: ME score distribution by Sormunen (U, N, M, R, H) and TREC (U, 0, 1) levels: breakdown for individual topics.

5.5 Magnitude Estimation and Crowdsourcing

We now turn to the second research question and investigate whether crowdsourcing is a valid methodology to gather magnitude estimation scores. We have already touched upon several related issues: the quality checks that we put in place in our experiment and that allowed us to gather good quality data (Section 5.2.2.5); the approximately log-normal distribution of our magnitude estimation scores (Section 5.3.2); the normalization and aggregation functions (Section 5.3.3); and the overall agreement of the magnitude estimation scores with official relevance judgments (Section 5.4). We now present more details, focusing on measuring judge agreement, on analyzing disagreement with official judges, and on discussing what happens when the number of collected judgments decreases.

5.5.1 Judge Agreement and Quality

It is well-known that relevance is subjective, even when focusing on "topical" relevance as is typically done in evaluation campaigns such as TREC, and that judges will therefore not perfectly agree. We define and discuss two kinds of agreement:

- *internal*: agreement within one group of workers judging the same topic-document pair; and
- *external*: agreement with expert judges (TREC, Sormunen).

5.5.1.1 Internal Agreement

In our data, there are at least ten workers judging the same topic-document pair, and the source of internal disagreement might be threefold: (i) the arbitrary, personal scale used by a judge when expressing an ME score; (ii) the subjective nature of relevance; and (iii) the often-feared low quality of work in crowdsourcing exercises. The first has already been discussed and is removed by the geometric average normalization process (Section 5.3.3). The second is a feature that one might not want to remove from the data: different judges can truly have different opinions on the relevance of a document, whatever scale of measurement is used. Usually IR evaluators assume that this variation is distilled into a single value (for example: mean, median or mode) that represents a population of users. Of course, the third source is in need of particular attention: low quality workers might express unreliable scores, and in general this would lead to low internal agreement.

To examine internal agreement, we can investigate the spread of the 10 normalized scores collected for each topic-document pair. Being on a ratio scale, an appropriate way of quantifying the upper limit of the spread is by the ratio between maximum and minimum $\max(s'_i)/\min(s'_i)$. Figure 5.8 shows the distribution of the ratio in our data. Another measure of dispersion that can be used is the geometric standard deviation (a version of standard deviation adapted for geometric mean and log-normal distributions). The chart on the right of the figure shows that the geometric standard deviation correlates quite well with the max/min ratio (Pearson's correlation is 0.999), and therefore is an almost equivalent measure.

While it is difficult to make conclusions about the absolute values of the ratio or geometric standard deviation without some reference point, it does highlight, for example, that there are 23 documents that seem to have an unusually high ratio of 10000 or more. Looking at each of these cases, there are two distinct causes. Firstly, 15 of the 23 looked like genuine attempts to assign ME scores, but the ranges of scores for each worker involved was very wide. One reason given in the comments by workers was that there was no zero available, so for documents that were truly off topic, they chose an arbitrary, very very small, number, making a wide scale. These wide scales were not markedly compressed by the normalization stage. In some cases the numbers assigned seemed arbitrary, but suitably large or small in correlation with the expert judgments. The second cause, in 8 of the 23 cases, seemed to be workers not following instructions or reading documents carefully. In these cases the comments provided by the workers bore no relationship to their numeric scores, and the numbers themselves seemed arbitrary. We will discuss the seemingly arbitrary nature of some scores later on (Section 5.8.2).

Another way of examining internal (intra-assessor) agreement is through measures of reliability. Krippendorff's α [53] is a widely-used reliability measure that is defined for different types of measurement scales. α is a chance-corrected measure of reliability, taking into account both observed and expected agreement. For ratio-level data, the α difference function is defined as square of the ratio of the difference between a pair of values assigned by two judges to the sum of the values assigned by the same pair of judges [76].

Across the full set of 4,269 topic-document pairs, $\alpha = 0.323$ (recall that each item

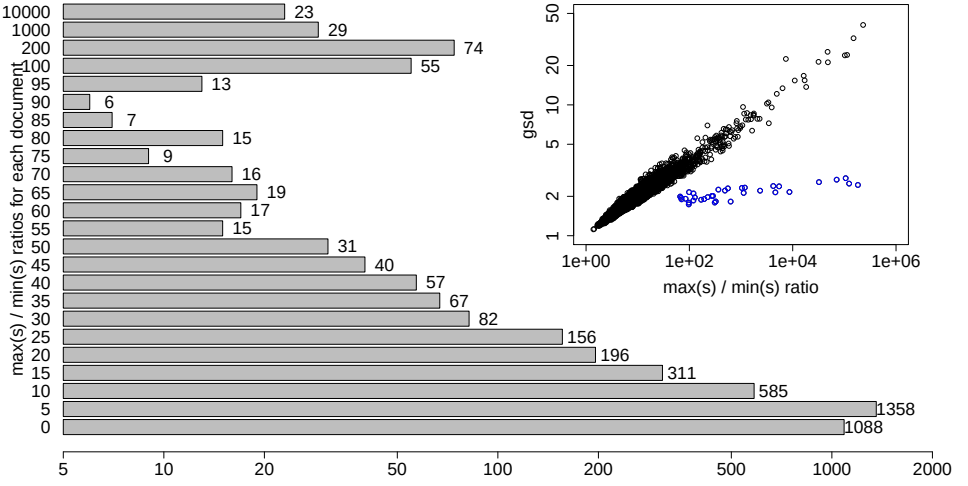


Figure 5.8: Number of documents with a $\max(s_i)/\min(s_i)$ ratio in a particular range (rounded to the nearest 5), out of 4269 documents. The inner panel shows a scatterplot of the correlation between $\max(s_i)/\min(s_i)$ ratio and the geometric standard deviation (log scale; the blue dots on the bottom are the 34 out of the 36 $\texttt{N}_k$ and $\texttt{H}_k$ documents, which received many more judgments and therefore exhibit a higher ratio; 7 points are left out, including the two missing $\texttt{N}_k$/ $\texttt{H}_k$ documents).

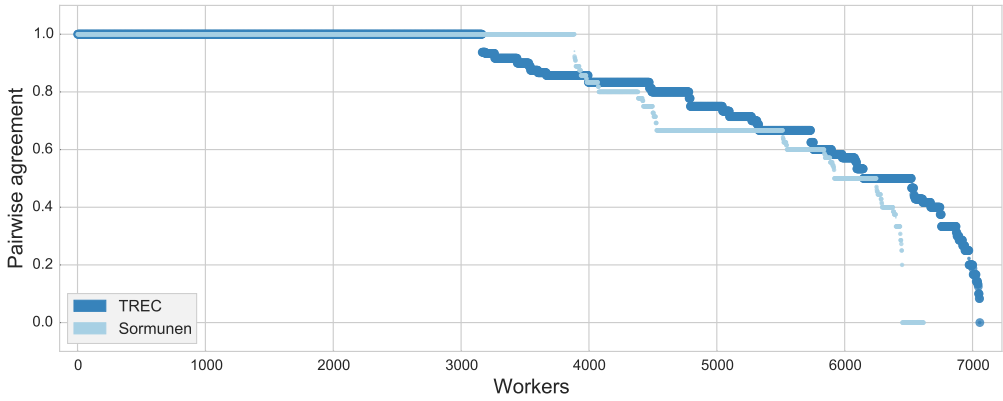


Figure 5.9: Agreement over the ranking of all pairs of documents for individual units using worker-assigned ME scores versus TREC or Sormunen categories. The number of pairs for each unit may vary, since some of the documents might be unjudged on the ordinal scales.

received at least 10 independent judgments; for the few items where a larger number of judgments were collected, the first 10 responses are used). Based on a bootstrap hypothesis test, this level of agreement is significantly different from zero agreement ($p < 0.01$). We note that the numeric value of α is difficult to interpret in this context, as to the best of our knowledge this is the first investigation of document-level relevance judgments using magnitude estimation, so no direct comparison is available.

Overall, the data show a not perfect but reasonably high internal agreement.

5.5.1.2 External Agreement and Quality of Workers

We now turn to analyze the external agreement, namely the agreement with the "official" TREC and Sormunen judges. In passing we also investigate the level of agreement when using different scales for judging relevance. We have already discussed the overall consistency with the official judges in Section 5.4. Here the results of a more specific analysis are shown: comparing the pairwise orderings between binary, ordinal and magnitude estimation relevance judgments. For example, if two documents are rated `N` and `M` on the ordinal scale, and the corresponding magnitude estimation score of the first is lower than or equal to the score for the second, this would be counted as an agreement. If the magnitude estimation score was higher for the second document, it would indicate disagreement. Intuitively, we can estimate that a worker has a high quality if the number of pairs of documents are ordered as in TREC or Sormunen. This measure of the quality of our crowdsourcing workers is reasonable even if the scales of the judgments differ, as long as they impose a ranking on the documents.

Figure 5.9 shows, for each unit, the pairwise agreement with both TREC and Sormunen judgments. Workers are generally good using this metric, with around 50% of

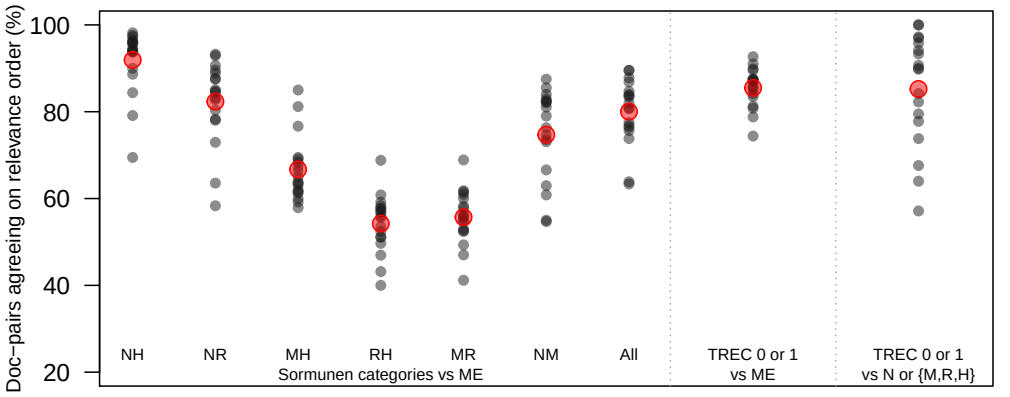


Figure 5.10: Raw agreement on the ordering of relevance of all document pairs (one small dot per topic) between judges as indicated on the x-axis (ME: magnitude estimation). The large (red) dot is the mean over all topics in each case. Unjudged documents are excluded.

the workers having perfect agreement; around 70% an agreement higher than 0.8; and around 86% of the workers have an agreement higher than 0.6. Note that as there are many documents unjudged on the Sormunen scale, thus 2376 units have only one pair of documents on which to compute agreement, contributing to the appearance of workers agreeing more with Sormunen than TREC using this metric.

Figure 5.10 shows the agreement data stratified by each of the possible six pairs of ordinal relevance levels of the Sormunen judgments (first six columns) and ungrouped (seventh column). The TREC vs. ME comparison, aggregated over all topics, is shown in the eighth column, and for comparison, the final column shows the pairwise agreement between TREC and Sormunen, assuming that `N` equates to TREC category "0", and the other three Sormunen categories equate to TREC category "1". Red circles indicate the mean score over the 18 topics for each group. It can be seen that the rates of agreement are highly consistent when comparing any of the three relevance scales. In particular, ME scores for `H` documents are greater than those for `N` documents 92% of the time (mean, first column), which is higher than the average agreement between TREC and Sormunen (85%—mean, last column). Similarly, the proportion of document pairs that have different TREC categories are ranked in the same order by ME scores 86% of the time (mean, second last column). ME and Sormunen ranking for documents in the `NR` category also agree 82% of the time (mean, second column), and over all pairs Sormunen and ME agree 80% of the time (mean, seventh column). This overall average is reduced by the lower agreements between pairs of documents that are deemed relevant and one or other are in the `M` or `R` categories (columns three to six). This is perhaps unsurprising, as distinguishing "marginal relevance" (`M`) from "irrelevant" (`N`) or "relevant" (`R`) can be difficult; and likewise distinguishing `R` from highly relevant (`H`) can also be challenging.

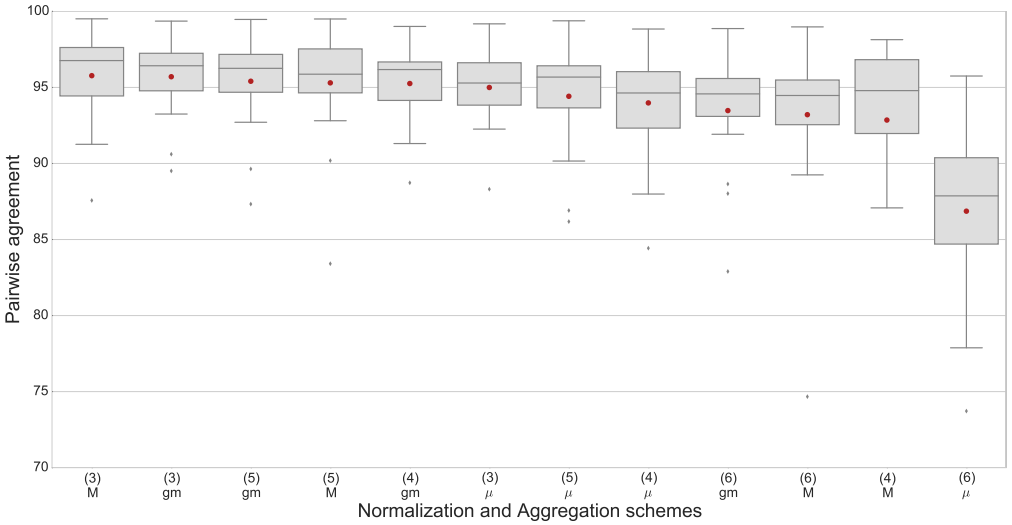


Figure 5.11: Effect of the four normalization functions (Equations (5.3), (5.4), (5.5), and (5.6) in the text) and aggregation functions (M = median, gm = geometric mean, and μ = arithmetic mean) on pairwise agreement.

Overall, the agreement between the ME approach and the existing categorical judgments seem reasonable, further supporting the validity of the magnitude estimation approach for gathering relevance judgments.

We now revisit the normalization and aggregation issues discussed in Section 5.3.3. We compare the following four normalization schemas

$$s'_i = e^{\log(s_i) - \mu(\log(unit)) + \mu(\log(topic))} \tag{5.3}$$

$$s'_i = e^{\log(s_i) - M(\log(unit)) + M(\log(topic))} \tag{5.4}$$

$$s'_i = e^{\log(s_i) - \mu(\log(\max(unit)), \log(\min(unit))) + \mu(\log(topic))} \tag{5.5}$$

$$s'_i = e^{\log(s_i) - \mu(\log(\mathtt{H}_k), \log(\mathtt{N}_k)) + \mu(\log(topic))} \tag{5.6}$$

(where the first one corresponds to that already discussed in Equations (5.1) and (5.2)), and the three aggregation functions mentioned at the end of Section 5.3.3, namely median, geometric mean, and arithmetic mean. Figure 5.11 shows the effect of the different normalization and aggregation functions on the quality of the workers measured by pairwise agreement. As we have discussed in Sections 5.3.2 and 5.3.3, we decided to use the somehow standard geometric averaging (Equation (5.3)) plus median normalized score. The chart in figure clearly shows that the selected two functions are among those avoiding a loss of quality. Both when considering the medians in the boxplots and when considering the means (the red dots in figure), on our data the highest pairwise agreement is obtained with:

- the normalization scheme in Equation (5.3), and
- the median or the geometric mean aggregators, rather than the arithmetic mean.

5.5.2 Failure Analysis

Despite the similar overall levels of agreement between the magnitude estimation method and ordinal relevance, Figures 5.6 and 5.7 show that some individual documents appear to be "misjudged". We therefore conducted a failure analysis, manually examining a subset of documents for which the Sormunen relevance level and the median magnitude estimation scores were substantially different (for example, where a particular document was assigned an ordinal Sormunen relevance level of `N`, but the median magnitude estimation score for the document was closer to the magnitude estimation scores assigned to `H` documents for the same topic, and substantially higher than the magnitude estimation scores assigned to other `N` documents for the same topic). Based on the manual examination of 34 documents where there appeared to be a significant flip between the ordinal and magnitude estimation scores, we found two broad classes of disagreements: those where one group of assessors appeared to be clearly wrong; and a class where the topic statement itself is so unclear as to be open to interpretation.

Of 34 documents that were examined, we found 14 cases (41.2%) where the Sormunen ordinal judgments appeared clearly wrong, and 9 (26.5%) cases where the crowd-based magnitude estimation assessments appeared clearly wrong. For this class of clear disagreements, where some assessors appear to be clearly wrong in the assignment of relevance (whether ordinal or magnitude estimation), the cause mostly appears to be that the assessors have missed or ignored a specific restriction included as part of the TREC topic. For example, the narrative of topic 410, *"Schengen agreement"*, includes the statement that: *"Relevant documents will contain any information about the actions of signatories of the Schengen agreement such as: measures to eliminate border controls..."*. Document `FT932-17156` makes clear reference to nine signatories of Schengen, and the process of removing passport checks. As such, it seems implausible that the document should be classed as `N`, or completely non-relevant. The original TREC binary judgment supports this view, having assigned a rating of 1 (indicating that the document is at least marginally relevant).

For the remaining 11 (32.4%) cases, it was not possible to determine that one assessment was clearly correct and the other wrong. Here, the original TREC topic statement itself was ambiguous, preventing a clear conclusion to be drawn based on the limited information that the topic statement provided. For example, a number of topics list several concepts in the narrative about what is or is not deemed relevant. However, they introduce ambiguity about whether the document must meet all of the listed criteria, or whether a subset is sufficient. For example, topic 407, *"poaching, wildlife preserves"*, states that *"A relevant document must discuss poaching in wildlife preserves, not in the wild itself. Also deemed relevant is evidence of preventive measures being taken by local authorities."* This raises the ambiguity of whether preventative measures by authorities against poaching, but not specifically in wildlife preserves, should be considered as being at least somewhat relevant, or completely non-relevant. We note that further ambigu-

ity is introduced due to the temporal mismatch between the time when the documents and topics were written (1990s), and when the magnitude estimation judgments are being made (2010s). This is particularly the case for topics that include terms such as "current".

The above failure analysis must also be interpreted in the context that it is known that assessors make mistakes when judging, perhaps due to fatigue or other lapses in attention, leading to self-inconsistencies [20, 107]; or they may display systematic errors due to a misunderstanding of the relevance criteria, or relevance drift [135]. Clearly, assessor errors will lower overall agreement rates when comparing assessments. Determining whether magnitude estimation relevance assessments lead to higher or lower error rates compared to using ordinal or binary scales is left for future work.

Overall, the examination of a set of clear disagreements demonstrates that there are cases where both groups of assessors (ordinal or magnitude estimation) are at odds with certain details of the TREC topic statements, and that these appear to occur at broadly similar rates. Moreover, the topic statements themselves are sometimes a cause of ambiguity, placing a practical upper-limit on the agreement that can be achieved. We conclude therefore that the magnitude estimation relevance judgments are sound and sensible, even when collected by means of crowdsourcing, having similar agreement rates with the ordinal Sormunen judgments as the Sormunen judgments have with TREC assessments.

5.5.3 How Many Workers are Required?

In Section 5.5.1.1 we observed that there is a spread of scores assigned by different workers to the same document for a topic, and that this is reasonable given that relevance is subjective quantity. For system evaluation it is typical to distill this variation into a single aggregate measure of relevance such as the mean or the median. This value is assumed to be an estimate of the true population relevance score for the document. If we have an estimate of the population variance, we can use standard sample size calculations to determine how many ME scores we should collect per document to be confident that we are estimating the true population mean score accurately. Specifically, to be 95% confident that our sample mean is within E of the true mean we need

$$n = 1.96^2\sigma^2/E^2$$

samples, where σ is the population standard deviation.

While we do not know the population standard deviation for ME scores of a particular document, we can look at the distribution of sample standard deviations that we have in our data as a guide. Figure 5.12 shows a histogram of the standard deviations of our normalized scores for each document. Using these as a guide, it would seem that assuming σ around 2 is reasonable. Table 5.1 shows the number of workers required per document for various values of E and σ. Assuming that we can tolerate ± 2 in our estimate of the true mean relevance, then we need about 7 workers, assuming $\sigma = 2$.

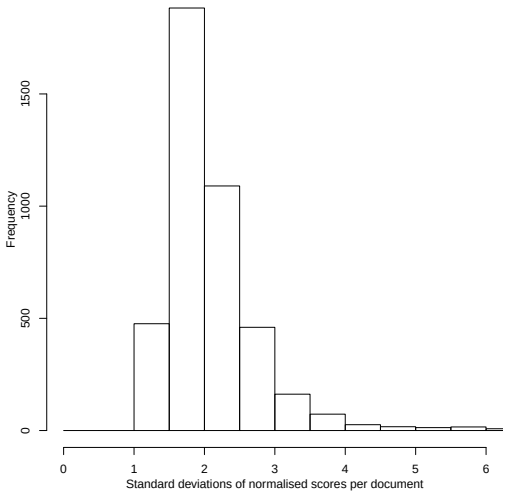


Figure 5.12: Histogram of standard deviations of the normalized scores for each topic-document in our data. There were 53 values greater than 6 which are not shown.

5.6 Magnitude Estimation For System-Level Evaluation

The third research question RQ3 concerns the direct application of magnitude estimation relevance judgments for the evaluation of IR systems, by considering their use for calibrating gain levels in two most widely-used gain-based IR evaluation metrics, nDCG and ERR.

5.6.1 Gain in the nDCG and ERR Metrics

Magnitude estimation provides scores that reflect ratios of human perceptions of the intensity of relevance of different documents in relation to a topic. This is directly related to the notion of gain in effectiveness metrics such as normalized discounted cumulative gain (nDCG). For example, Järvelin and Kekäläinen [61] describe "cumulative relevance gain" that a user receives by examining a search results list, and discuss setting "relevance weights at different relevance levels". That is, weights are applied to each level of an ordinal relevance scale, and can be chosen to reflect different assumptions about searcher relevance behavior. However, the "standard" approach that has been adopted when using nDCG is to simply assign ascending integer values to the ordinal levels, starting with 0 for the lowest (non-relevant) level; for a 3-level ordinal scale, the default gains would be 0, 1 and 2, as for example implemented in `trec_eval`.[1]

In addition to modeling different levels of gain, or relevance, the discounted cu-

[1] `http://trec.nist.gov/trec_eval`

Table 5.1: The number of workers required to judge each document to be 95% confident that mean normalized scores are within E of the true population value.

	σ					
E	1	2	3	4	5	6
0.5	27	107	239	425	664	956
1.0	7	27	60	107	166	239
1.5	3	12	27	48	74	107
2.0	2	7	15	27	42	60
2.5	2	5	10	17	27	39
3.0	1	3	7	12	19	27
3.5	1	3	5	9	14	20
4.0	1	2	4	7	11	15
4.5	1	2	3	6	9	12
5.0	1	2	3	5	7	10

mulative gain metric also includes a discounting function, so that documents that are retrieved further down in a ranked search results list contribute smaller amounts of gain, reflective of factors such as the effort or time that a user must invest while working their way through the list [61]. Using a logarithmic discount, discounted cumulative gain at cutoff N is calculated as

$$\text{DCG@N} = G_1 + \sum_{i=2}^{N} \frac{G_i}{\log_2(i)},$$

where G_i is the gain value for the document at position i in the ranked list. To enable fair comparisons across topics with different numbers of relevant documents, the DCG@N score is divided by an *ideal* gain vector, a ranking of documents in decreasing relevance order, to obtain normalized discounted cumulative gain, nDCG@N.

Expected reciprocal rank is a metric based on a cascade model of searcher behavior, where the probability of continuing on to the next position in the ranked results list is influenced by the relevance of previous items. The metric calculates the expected reciprocal rank at which the searcher will stop [24], and is defined as

$$\text{ERR@N} = \sum_{i=1}^{N} \frac{R(G_i)}{i} \prod_{j=1}^{i-1} (1 - R(G_i)),$$

where $R(G) = (2^{G_i} - 1)/2^{\max(G_i)}$. In the analysis that follows, we calculate both nDCG and ERR up to a depth of $N = 10$.

For both nDCG and ERR, gain is based on the relevance of a document, which has previously been measured on an ordinal scale, typically with 3 [65] or 4 [61] levels,

depending on the test collection being used. Since magnitude estimation relevance judgments are continuous rather than ordinal in nature, and reflect the perceived level of relevance for individual topic-document combinations, it is possible to assign more fine-grained gain values, potentially reflecting different gains for individual documents, or even for individual searchers.

5.6.2 Comparative System Rankings

Given that the agreement between our magnitude scores, TREC judgments and Sormunen's judgments is not perfect, it is reasonable to expect that relative system effectiveness orderings computed with each of these as a basis may differ. We first examine the correlation between system orderings using magnitude estimation scores and TREC relevance as gain values, as for both these judgment sets we have nearly complete coverage of the top 10 documents of all runs submitted to TREC-8, for our 18 judged topics. Rather than taking the median of all scores assigned to a document, we take the median of scores taken from the three units that have document orderings that have the highest agreement with the TREC orderings, or Sormunen orderings, respectively. In effect, the scores derived from units with high internal agreement as computed in Section 5.5.1.1, with ties broken in favor of units with the least unjudged documents. By aggregating scores over units that have a high agreement with the original TREC orderings, any differences in system rankings we see should be more attributable to the scale of the gain values, rather than perturbations in document orderings given by different gain values.

Figure 5.13 shows the concordance between system rankings using TREC categories and ME scores as gain values. It can be seen that there are definite changes in system rankings when using the different relevance scales, with Kendall's τ correlations of 0.677 and 0.222 for nDCG@10 and ERR@10, respectively. As the aim of system evaluation is to identify top-performing systems, we also consider changes in the *top set*, defined as the group of systems that are statistically indistinguishable from the system with the highest mean effectiveness, using a paired Wilcoxon signed-rank test with $p < 0.05$. These systems are shown as red dots. The overlap of the top set, based on the TREC and magnitude estimation relevance judgments is 82% for nDCG@10 and 20% for ERR@10, confirming that the perturbation in system ordering has an impact on which systems are identified as being the best performers.

Since the Sormunen judgments only cover around 19% of the documents that occur in the top 10 of all TREC-8 ad hoc runs, evaluating the original runs using these relevance judgments is problematic, due to the large number of unjudged documents. However, the judgment coverage of individual ranked lists (that is, for particular topics within a run) varies substantially. Therefore, to enable a comparison between the ordinal judgment and magnitude estimation judgments on system orderings, we simulate runs that only include ranked lists for topics that are fully judged by Sormunen.

For each topic, we identify all ranked lists within the full set of runs that have Sormunen judgments for all top 10 documents for that topic; we call each of these a *complete sub-run*. There are 12 topics where there are at least two complete sub-runs

out of all of the TREC-8 ad hoc runs for that topic. To form a simulated retrieval system, we then randomly choose one complete sub-run for each of the 12 topics to give a "full run" over all 12 topics. Using this method, we construct 100 simulated system runs that are random merges of complete sub-runs from real runs. While these simulated systems are not actual TREC submissions, they are plausible in that each individual topic ranking is from a real TREC run.

Figure 5.14 shows that there is also a large discordance between the system rankings using the Sormunen categories and magnitudes as gain values. However, note that the scale is different in this figure than in Figure 5.13, as the simulated systems all have relevant documents in the top 10, and so are high scoring. Correlation as measured by Kendall's τ is even lower here, at 0.147 and 0.06 for nDCG@10 and ERR@10, respectively. The overlaps in top sets are 32% for nDCG@10 and 68% for ERR@10. The relatively high performance of systems on this simulated collection is to be expected, as only complete sub-runs were selected, and around half of the Sormunen judgments are for documents that were originally rated as relevant by TREC, and so the runs in the simulated systems have a high number of relevant documents. It is interesting that there is still a sizable change in the top set using either metric with the different judgments, however.

5.6.3 Judgment Variability and System Rankings

The previous section showed that using magnitudes as gain led to changes in the ranking of systems using different metrics, even when only aggregating over magnitude judgments that agreed with individual pairwise document rankings of the original categorical scale. There is, however, substantial variation in the magnitudes assigned to documents by our judges. Perhaps using magnitudes from different judges would lead to different system rankings. As we have at least 10 judgments per topic-document pair, we can re-sample individual scores many times, recompute system orderings, and get a distribution of τ values. The results are shown in Figure 5.15: the correlations between system orderings seen in the previous section sit about in the middle of these distributions. The spread of the distributions reflect the wide range of individual variation – differences in perceptions of relevance – that in turn leads to a wide range of τ values.

An alternate explanation for the low τ values is that gains set as magnitudes are generally much higher than the gains set by Sormunen's categories. From Figure 5.6 it is apparent that magnitudes are generally in the range of 2 to 15 (although some are as large as 100 or 1000), whereas using categories as gains the values are 0, 1, 2 or 3. However, because both nDCG and ERR are normalized, this scale effect is nullified to a degree. For example, multiplying gains by a constant has no effect on nDCG, and even altering the category scores using a small exponential or additive constant has little effect. Table 5.2 shows this Kendall's τ for our 12 topics on the simulated systems using various mappings of Sormunen categories to gain values, and a mapping of those categories. The final two rows hint that using gain values from a broad scale, while preserving document orderings, may alter system rankings, particularly with nDCG@10, compared with simply using the category values as gains. The final row perhaps best

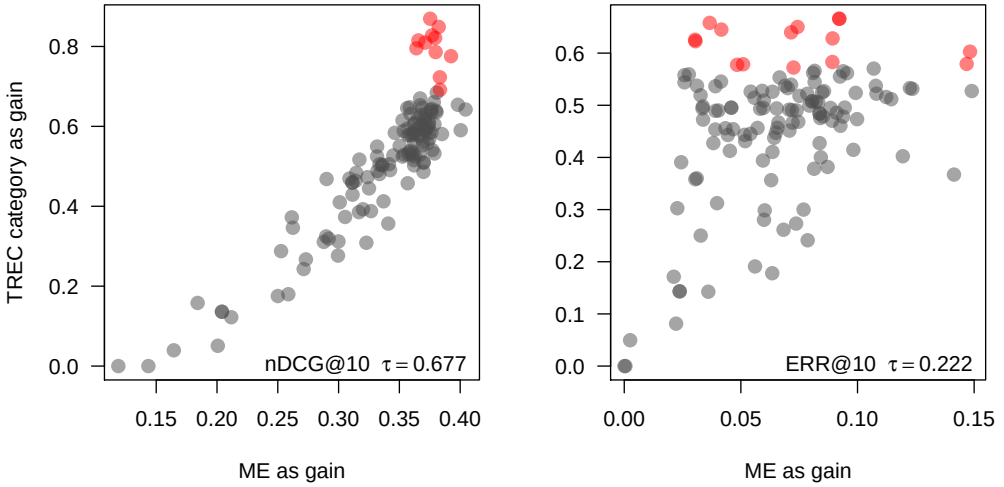


Figure 5.13: System scores using TREC categories (y-axis) and magnitudes (x-axis) as gains in the nDCG@10 and ERR@10 metrics. There is one dot for each of the 129 systems participating in the TREC-8 ad hoc track, with red dots showing the top-set (see text). Magnitudes used are the median of the three units per document that most agree with TREC document rankings. Kendall's τ is shown.

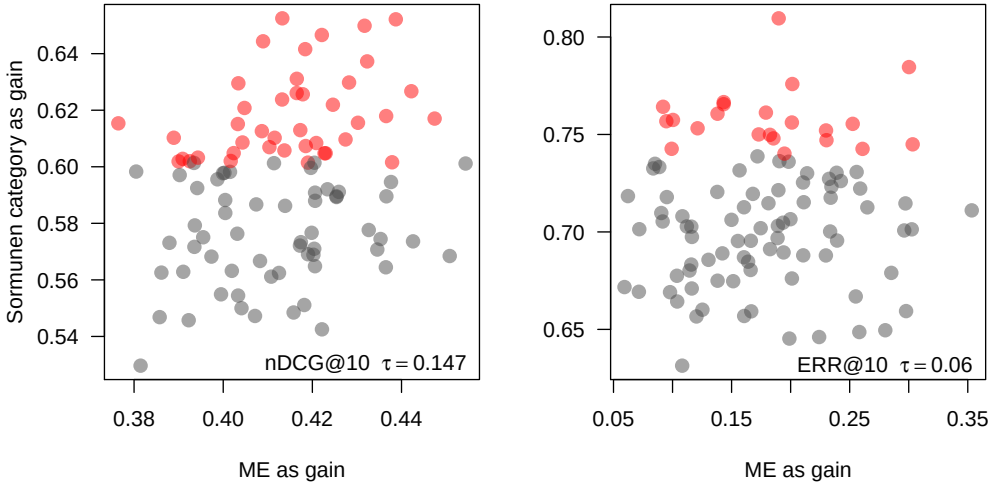


Figure 5.14: As for Figure 5.13 but using Sormunen categories (y-axis) and simulated runs as described in the text.

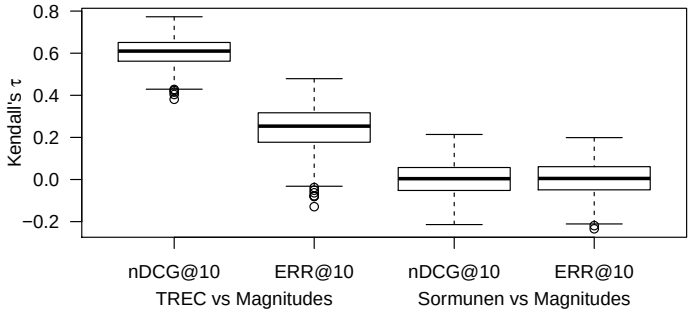


Figure 5.15: Kendall's τ between system rankings obtained using gains from the judgments and metrics indicated on the x-axis, where magnitudes are randomly sampled for each document 1000 times. Compare with τ in Figures 5.13 and 5.14.

represents our magnitude scores, and so goes partway to explaining the low τ values in the previous section.

One way to examine this further is to split our data into *narrow* units, those whose $\mathtt{H}_k/\mathtt{N}_k$ ratios are less than 5 (the median value), and *wide* units, where the ratios $\mathtt{H}_k/\mathtt{N}_k$ are 5 or more.

Table 5.3 shows the correlation (Kendall's τ) between system orderings, and overlap of runs in the top set, when using TREC and Sormunen judgments, and magnitude estimation scores when the median is obtained from units that used a narrow scale, a wide scale, or from all units (note that the τ values in the All column match those in Figures 5.13 and 5.14. In each case the score for a topic-document pair is taken as the median of the 3 units that most agree with TREC or Sormunen and that are either narrow, wide or any as appropriate. When there was no narrow or wide judgment for a topic-document pair (all units containing that topic-document pair had $\mathtt{H}_k/\mathtt{N}_k$ above or below the median), the median of the three best units was used. This occurred in 12% of TREC topic-document pairs, and 7% of Sormunen topic-document pairs. System orderings based on narrow or wide units show greater differences (lower τ values) than when considering the full data set. This is consistent with the results in Table 5.2. The overlap of the top set is generally less than 100% for the narrow/wide data sets, indicating that some perturbation of the set of "equivalent" top systems is occurring, but the pattern is not as clear as for τ values. This may be due to the relatively small number of topics that is being considered, especially for the simulated systems, and more topics are required before a definitive conclusion about the effect of narrow and wide units on the top set can be made.

Overall, there are clear differences in system effectiveness orderings as measured using Kendall's τ, when using narrow and wide units. In particular, it appears that using the median of narrow units, or of all units, leads to higher correlations with existing measurement techniques, compared to when using wide units. We can therefore infer that current evaluations using nDCG@10 and ERR@10 do not assume wide units as their underlying user model. This is important because it shows that nearly half of our

Table 5.2: Correlation (Kendall's τ) and overlap of top-set between system orderings when using Sormunen categories as gain ($G_i = C_i$), and other mappings between categories and gains on the simulated systems, when using the nDCG@10 and ERR@10 metrics.

	Kendall's τ		Top-set overlap	
Mapping	nDCG@10	ERR@10	nDCG@10	ERR@10
$G_i = 100 \times C_i$	1.000	0.698	100%	100%
$G_i = 1 + C_i$	0.989	0.969	98%	95%
$G_i = 10 + C_i$	0.961	0.939	95%	91%
$G_i = 100 + C_i$	0.957	0.939	95%	100%
$G_i = 10^{1+C_i}$	0.720	0.698	66%	100%
$G_i = 2^{1+C_i}$	0.870	0.702	42%	73%

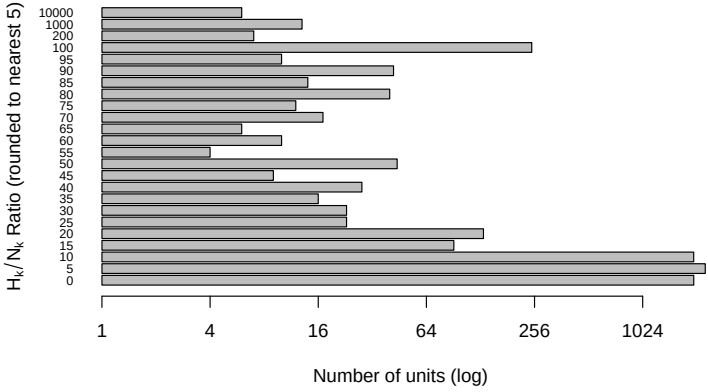


Figure 5.16: Number of units with a particular $\mathtt{H}_k/\mathtt{N}_k$ ratio over all topics. There were a total of 7,059 units. 37 units with frequency less than 4 are not shown.

Table 5.3: Correlation (Kendall's τ) and overlap of the top set between system orderings when median document magnitude scores are based on narrow, wide or all units.

Source of gains	Narrow	All	Wide
TREC, τ			
nDCG@10	0.679	0.677	0.744
ERR@10	0.361	0.222	0.247
TREC, top set			
nDCG@10	63.64%	81.82%	72.73%
ERR@10	20.00%	20.00%	60.00%
Sormunen-Simulated, τ			
nDCG@10	0.036	0.147	0.382
ERR@10	0.017	0.060	0.323
Sormunen-Simulated, top set			
nDCG@10	9.76%	31.71%	68.29%
ERR@10	68.18%	68.18%	100.00%

participants are not behaving according to the user model assumed by current evaluation techniques. Figure 5.16 shows the full distribution of the "width" ($\mathtt{H}_k/\mathtt{N}_k$ ratio) of units rounded to the nearest 5 (with bars containing less than 4 units omitted).

5.7 Investigating Gain Using Magnitude Estimation

The previous section used perturbations in system orderings to examine the difference between using magnitude estimate relevance scores and the usually employed ordinal relevance levels as gains. This section directly considers gain profiles, answering our fourth research question RQ4 by considering what additional insights magnitude estimation can provide into user perceptions of relevance.

The gain weights of the nDCG metric allow for the modeling of different user relevance preferences. However, in practice gains are usually set to one of two profiles: a *linear* setting (as defined in Section 5.6); or an *exponential* setting, where the ordinal relevance level forms a power of 2 [18], placing more emphasis on highly relevant documents. Figure 5.17 shows the distribution of magnitude estimation scores, normalized by the $\mathtt{N}_k$ score in each unit (intuitively, a measure of relevance standardized for each user). Black lines show the median magnitudes for each group. Superimposed are the two default profiles, linear (white circles), and exponential (white triangles). It can be seen

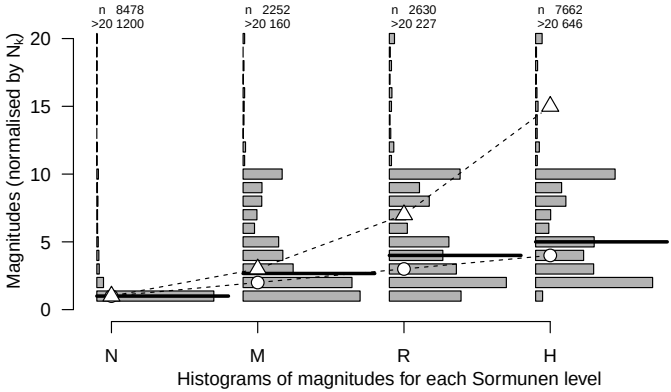


Figure 5.17: Distribution of magnitudes (normalized by $\mathtt{N}_k$) for each Sormunen level. Circles are linear gain values, $\{1, 2, 3, 4\}$, while triangles represent exponential gain values, $\{2^1, 2^2, 2^3, 2^4\}$. Black lines are medians; the text "n" gives the total number of magnitudes in the level, and "> 20" shows the count of scores above 20.

that, compared to actual user perceptions of the relevance space, the linear gain profile is fairly close to the gain that should be allocated at each level to satisfy the (mythical) median user. The exponential profile is further from the median, overestimating the gain for the `R` and `H` levels, but catering for some judgments.

It is readily apparent, however, that the large spread of the distributions of magnitudes cannot be captured in a single gain formulation. There is simply too much individual variation to warrant the simplification of relevance perceptions to a single profile.

Kanoulas and Aslam [65] investigated how gain values could be set in nDCG so as to maximize the stability of system rankings, based on the generalizability and dependability coefficients from Generalizability Theory. The settings were estimated empirically, based on system runs from the TREC 9 and 10 Web Tracks and the TREC 12 Robust Track. We compare this approach with ME scores of relevance perceptions gathered from users. Figure 5.18 shows the ratio of magnitudes for document pairs in the Sormunen categories highly relevant and partially relevant. The ratios are computed within each unit collected (8 documents for one topic), and if the unit contains more than one document in some category, all combinations of the documents are included in the graph. To allow comparison with gain values recommended by Kanoulas and Aslam [65], we either ignore marginally relevant documents (Sormunen category M), leftmost box; fold them in as partially relevant, middle box; or call them partially relevant and promote category R to highly relevant. When a unit did not contain a marginally relevant document, it is omitted from the middle and right boxes, hence there are less grey dots for those plots.

Kanoulas and Aslam [65] recommend a ratio of 1.1 for the gain values of relevant to highly relevant documents to maximize the effectiveness of a system comparison exper-

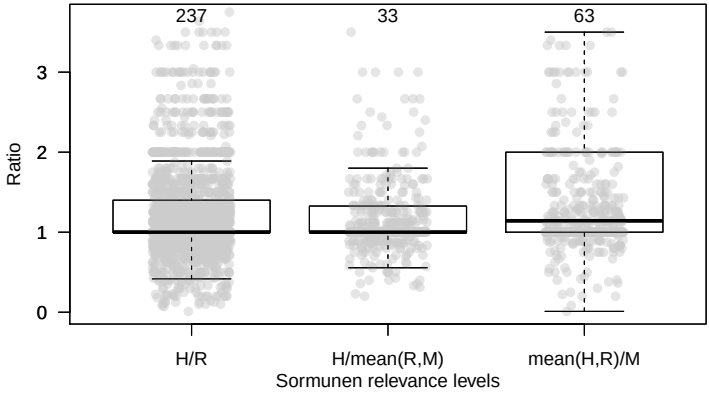


Figure 5.18: Ratio of the magnitude scores assigned to the Sormunen categories highly relevant (H) to relevant (R) and marginally relevant (M) over all pairs of documents. Numbers indicate the count of document pairs that are greater than 3.5 for each box. Each grey dot is a ratio of one document pair.

iment. Using magnitudes directly as gain values gives a median of 1.00, 1.00, and 1.14 in the three boxes in Figure 5.18, which happens to be close to their recommendation. However, again, we see a wide variation from this median, and mean values of the ratios are 48, 22 and 24 respectively, further suggesting that there may not be a single gain setting that is suitably representative across different user perceptions and tasks.

5.8 Conclusions and Future Work

Relevance is a fundamental concept in information retrieval, underpinning test collection-based system effectiveness evaluation. This is the first study to have collected large scale, real user data about relevance perceptions using magnitude estimation, obtaining through crowdsourcing over 50,000 ratings across 18 TREC topics. We close the paper by briefly summarizing our findings and sketching future work, as well as highlighting some limitations of this study.

5.8.1 Summary

The first research question that was considered asked whether magnitude estimation is a suitable technique for judging relevance at the document level. A comparison of the overall distribution of magnitude estimation judgments shows a close alignment with ordinal graded relevance categories, both at the individual topic level, and across the set of 18 topics as a whole. A high level of consistency was also established between the distribution of magnitude estimation judgments and classical binary relevance assessments. A failure analysis demonstrated that for those cases where the magnitude estimation and ordinal judgments disagree, this is typically not due to problems with

magnitude estimation per se, but rather arises due to different judging context, or from ambiguity in the original topic statements.

Magnitude estimation is a technique that may be unfamiliar to many people. The second research question therefore focused on the suitability of using crowdsourcing to obtain magnitude estimation judgments at scale. Aiming to avoid spam workers, our crowd task included four quality checks: a practice line length estimation task; a multiple choice topic understanding question; a check that magnitude estimates that were entered for two documents with known ordinal relevance were rank consistent; and, a minimum time of 20 seconds spent per document. The magnitude estimation judgments that were obtained after this process showed a high level of external agreement with existing TREC binary and Sormunen ordinal judgments: rank ordered magnitude estimation scores agree with binary judgments 86% of the time, and with ordinal judgments 80% of the time (the agreement with ordinal judgments is higher for ordinal levels that are further apart, and lower for ordinal levels that are closer together). The ranked agreement between binary and ordinal judgments was 85%, giving confidence that the using magnitude estimation is robust as a scaling technique for document-level relevance.

Considering the impact of magnitude estimation relevance judgments on IR system evaluation, the third research question, our analysis showed that results can vary substantially when different relevance scales are used, with a τ of 0.677 between system orderings when effectiveness is measured using nDCG@10 and comparing magnitude estimation scores with binary relevance across TREC runs. When comparing magnitude estimation with ordinal relevance on simulated runs, the correlation is even lower at 0.147. Moreover, when individual magnitude estimation scores are considered, rather than the mean, the correlations show a wide range of individual variation in terms of the perception of relevance. A key factor appears to be that different searchers have different perceptions of the extent of relevance variation, using wide or thin scales. Overall, the analysis suggests that it is important to incorporate different judging scales, rather that assuming that a single scale can closely reflect a larger population of users.

We also employed magnitude estimation to directly investigate gain profiles, the fourth research question. Typically, gains in nDCG are set as linear or exponential profiles. Our analysis of user-reported relevance perception showed that the linear profile is a closer fit to the relevance perceptions of the "average" user. However, the distribution of magnitudes again suggests that attempting to fit a single profile (or view of relevance) for system evaluation is unlikely to be sufficient, if the expectation is that the evaluation should be reflective of a broad user base.

While on average our magnitudes were broadly equivalent to previous ordinal scales, the outstanding feature of our data was the wide range of scores that participants chose to employ in the judging task. In particular, at least half of the participants chose gain values that are not consistent with currently used values. Section 5.6.2 shows that using judgments made on a wide scale leads to different system rankings than judgments collected on a narrow scale. Recall that these scales are not imposed on the judge, as they are in all previous relevance judgment tasks in the literature, but are chosen by the participants themselves.

This is another key contribution of this study: when a priori categorical scales are

used for relevance judgment tasks, there is no possible way to capture variance in human perception of the scale of relevance. In turn, this limits our understanding of how gain should be set in DCG-like metrics, and hence our ability to accurately evaluate systems.

5.8.2 Limitations and Future Work

This study is the first to investigate document level relevance judgments using magnitude estimation, at scale. A much-cited benefit of magnitude estimation is that it is a ratio scaling technique. Indeed, workers were instructed to assign relevance scores as ratios to those assigned to previous documents. However, some documents in the collected data lead to ratios that have a large number of significant figures. For example, in one unit, the first document was given a score of 54, and the second 76, implying that the worker thought that the second document was 1.407 times more relevant than the first. This seems like an unusually high amount of precision for such a task, and it may be that the worker was choosing numbers without closely following the ratio requirement. Intuitively, one might think that it would be possible to distinguish the relevance of two documents at a ratio of one significant figure (for example, "2 times as relevant", or "100 times as relevant"); or maybe even two significant figures ("21 times as relevant", or "140 times as relevant"); but it seems very unlikely that this could be done to three significant figures or more without extensive training and effort. We can safely assume that our crowdsourced workers were not trained in relevance assessment, nor spent more than a minute or two on each document (mean time to assess 78 seconds, standard deviation 86 seconds).

If we assess each unit of work using a criterion that each relevance score must have a ratio with at least one previously assigned score that has at most two significant figures, then 4806 of the 7059 units match. If we are more strict, and insist on only one significant figure, then only 3591 of 7059 units match. It seems, therefore, that about half of the units are being completed without a strict adherence to the idea of assigning numbers following ratio relationships. We note, however, that this might not be a problem in practice because of approximation and balancing effects: the 1.407 figure above is likely not exact, but may be a good approximation of a value between 1 and 2. Moreover, relevance is not a strictly defined concept, and the magnitude estimation is intended to reflect human perceptions. Overall, the obtained magnitude estimation scores were shown to have high external consistency with other judging scales.

We have assumed that topical relevance collected using magnitude estimation can be used directly as gain in DCG-like measures. While perhaps being more representative of user perceptions than other relevance scales, this still makes many assumptions about the user's search process which are probably untrue: for example, ignoring the interdependence of documents and other aspects of relevance. In future work, we plan to investigate whether magnitude estimation can be used to reliably scale other aspects of relevance such as novelty, as well as search outcome measures such as satisfaction, and to examine whether this can lead to more meaningful gain representations.

In this work we focused on quantitative analysis of the magnitude estimation process. As part of the judging interface, judges were also asked to enter short textual comments

to justify their choice of score. We plan to analyze this qualitative data in future work.

The bulk of the analysis in this paper considered magnitude estimation judgments as a reflection of relevance at the aggregate level. A further interesting line of analysis would be to consider the technique at the topic level, as different perceptions of relevance may be useful of indications of topic difficulty, or as potential calibration functions of how well a system could be expected to perform.

6

Preliminary results on agreement in relevance assessment

This chapter presents some preliminary results of studies concerning (dis)agreement among crowdsourcing workers. As introduced in Sections 2.4.2 and 2.4.3, the aggregation of data collected in crowdsourcig experiments is traditionally made without taking into account agreement among workers. In fact, disagreement is typically treated as noise, then opportunely filtered out by aggregation functions. By analysing workers agreement, we can discover valuable information, that can be used with profit, e.g., to detect problems of the task, to validate worker results, or to detect malicious workers. In this chapter we discusses some introductory analysis of agreement made on the TREC 2010 Relevance Feedback [17] test collection. The chapter is briefer than the previous ones, and presents results taken from ongoing works. Section 6.1 introduces the TREC 2010 Relevance Feedback collection. Section 6.2 details how the agreement by chance can be computed. Section 6.3 discuses agreement on TREC 2010 Relevance Feedback. Section 6.4 shows how actual agreement can be normalised. Section 6.6 presents the conclusions.

6.1 TREC 2010 Relevance Feedback

This section aims to study the agreement among workers, and for this purpose, we consider the TREC 2010 Relevance Feedback test collection. The ground truth data for the collection was created on English Web pages from the ClueWeb09 collection[1], for English search queries from the TREC 2009 Million Query Track [21]. Relevance levels are: not-relevant, relevant, and highly relevant (also referred with the values 0, 1 and 2). We focus on documents having at least five judgements and relevance score labels by NIST ("gold" standard values).

[1] `http://lemurproject.org/clueweb09/`

6.2 Agreement by chance

Before focusing on the details of the Relevance Feedback 10 test collection, we analyse agreement by chance, which consists in the expected agreement obtainable when all workers judge documents in a completely random way. First, we can easily notice that possible combinations of judgements are limited and easily computable. Each of the five documents receives exactly five relevance judgements, with values between 0 and 2. The possible combinations of scores are exactly 243 (3^5).

$$\begin{matrix} w_1 \\ \begin{bmatrix} 0 \\ 1 \\ 2 \end{bmatrix} \end{matrix} \times \begin{matrix} w_2 \\ \begin{bmatrix} 0 \\ 1 \\ 2 \end{bmatrix} \end{matrix} \times \begin{matrix} w_3 \\ \begin{bmatrix} 0 \\ 1 \\ 2 \end{bmatrix} \end{matrix} \times \begin{matrix} w_4 \\ \begin{bmatrix} 0 \\ 1 \\ 2 \end{bmatrix} \end{matrix} \times \begin{matrix} w_5 \\ \begin{bmatrix} 0 \\ 1 \\ 2 \end{bmatrix} \end{matrix} \rightarrow 243(3^5) \text{ combinations}$$

We are interested in the mean and the variance of the values of these combinations. Some of these combinations have equivalent values of both mean and variance. For example, the combinations $[0, 1, 0, 0, 1]$ and $[1, 0, 1, 0, 0]$ have exactly the same mean and variance. Obviously, the only combinations with variance equal to zero are three, precisely those having the same number repeated five times. All the other have mean included between 0 and 2, and variance strictly positive. Some combinations have in common only mean or only variance. For example, the combinations $[0, 0, 0, 0, 1]$ and $[3, 2, 3, 3, 3]$ have the same variance (0.16), but different mean (0.2 and 0.8 respectively); the combinations $[0, 0, 1, 2, 2]$ and $[0, 1, 1, 1, 2]$ have the same mean (exactly the value 1) but different variance (0.8 and 0.4 respectively). The 243 possible combinations generate 21 unique mean-variance pairs. Figure 6.1 shows the 21 mean-variance pairs; each of them is represented as a circle in the chart which specifies the number of combinations that generate such specific pair. Therefore, supposing to collect five random relevance judgements, each one (an integer number) between 0 and 2, the expected variance and mean are respectability 0.53 and 1. The figure also shows how mean-variance pairs in the central part of the chart are generated by more different combinations compared those closer to the borders of the chart.

6.3 Agreement on the documents of the TREC 2010 Relevance Feedback collection

We focus on the Relevance Feedback 10 collection. For each document of the collection, we compute mean and variance of the five relevance scores received. As shown in Section 6.2, there are 243 possible score combinations by which a document can be judged and 21 possible mean-variance pairs in which a document can be assigned. Figure 6.2 shows how documents are distributed on the 21 mean-variance pairs. Each circle of the chart represents the amount of documents having scores that generate the same mean-variance pair.

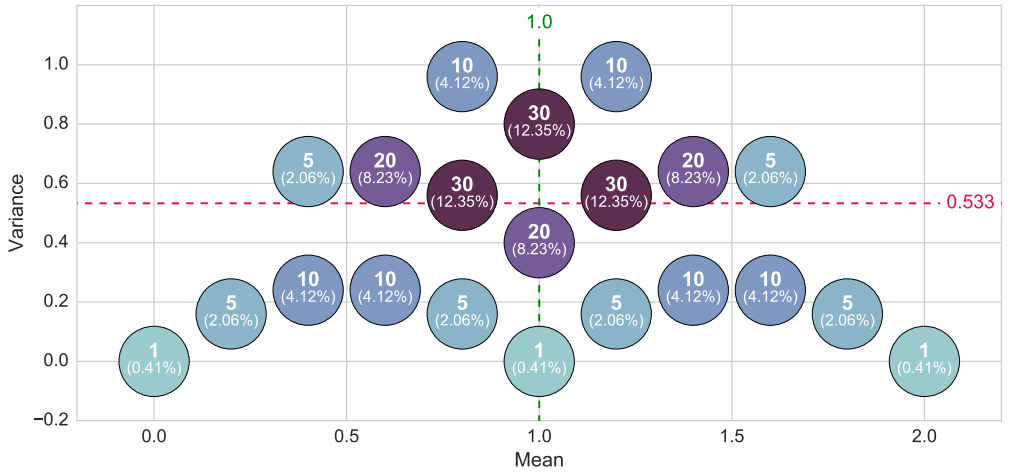


Figure 6.1: The 21 unique mean-variance pairs obtainable by the 243 combinations of five items, where each of them can assume value equal to 0, 1, or 2. The number inside each circle corresponds to the amount of different combinations which can generate that pair.

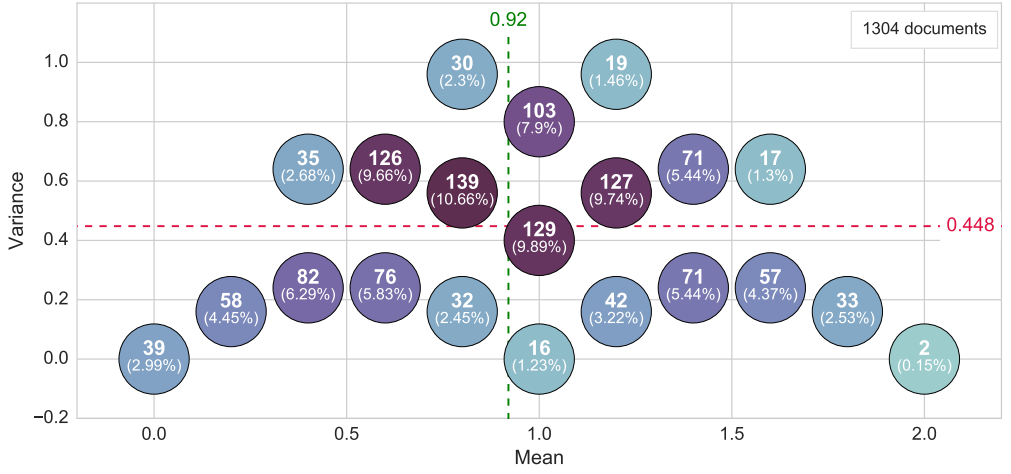


Figure 6.2: The 21 mean-variance pairs obtained analysing data from the Relevance Feedback 10 collection.

Figure 6.3: The 21 mean-variance pairs obtained analysing data of the Relevance Feedback 10 collection, breakdown over “NIST” labels.

The average of document scores variance (0.448) is smaller than that expected when judging documents by chance (0.533).

The average of the document scores mean is 0.92. This value is strongly influenced by the amount of not-relevant, relevant and highly-relevant documents. Then, we recompute the chart, by considering separately the 3 gold standard levels.

The result is shown in Figure 6.3. The three charts allow to appreciate how the average of mean document scores gradually increases when considering not-relevant documents (0.73), relevant documents (0.95), and highly-relevant documents (1.13). It indicates that the workers judge in a consistent way with respect to "NIST" experts. The average variance is stable over the 3 charts; this means that workers judge documents in similar ways independently of their relevance score.

6.4 Normalization of agreement

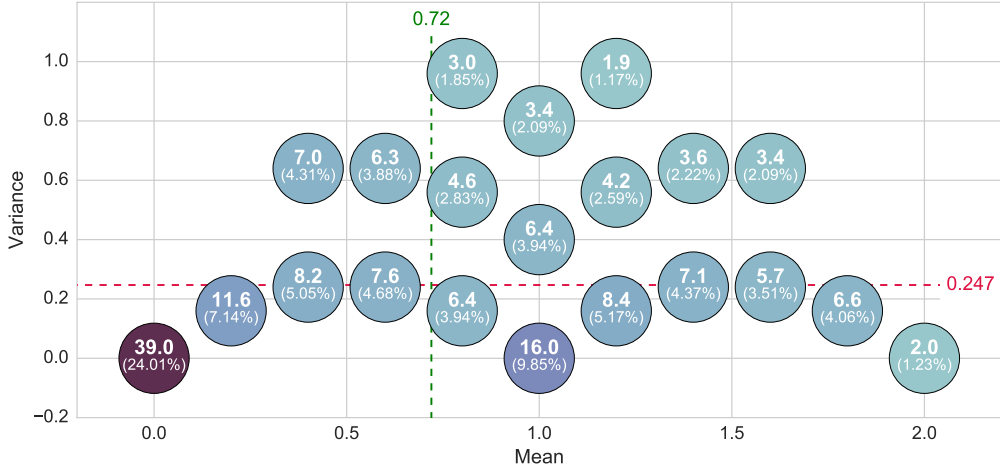


Figure 6.4: The 21 mean-variance pairs obtained normalising the data of the Relevance Feedback 10 collection.

As mentioned in the last part of the Section, some mean-variance pairs can be generated by more score combinations than others, therefore they occur more frequently when workers assess in a random way. For this reason, it is interesting to normalise the gathered scores; this is done by considering a mean-variance pair, and dividing the amount of documents by the number of combinations that can generate that specific mean-variance pair. Essentially, we divide the amounts of the circles in Figure 6.2 by those in Figure 6.1.

The result is shown in Figure 6.4. The variance decreases to 0.247. As before, we recompute the chart by considering separately not-relevant, relevant and highly-relevant documents. This split produces the charts in Figure 6.5. The normalization increased

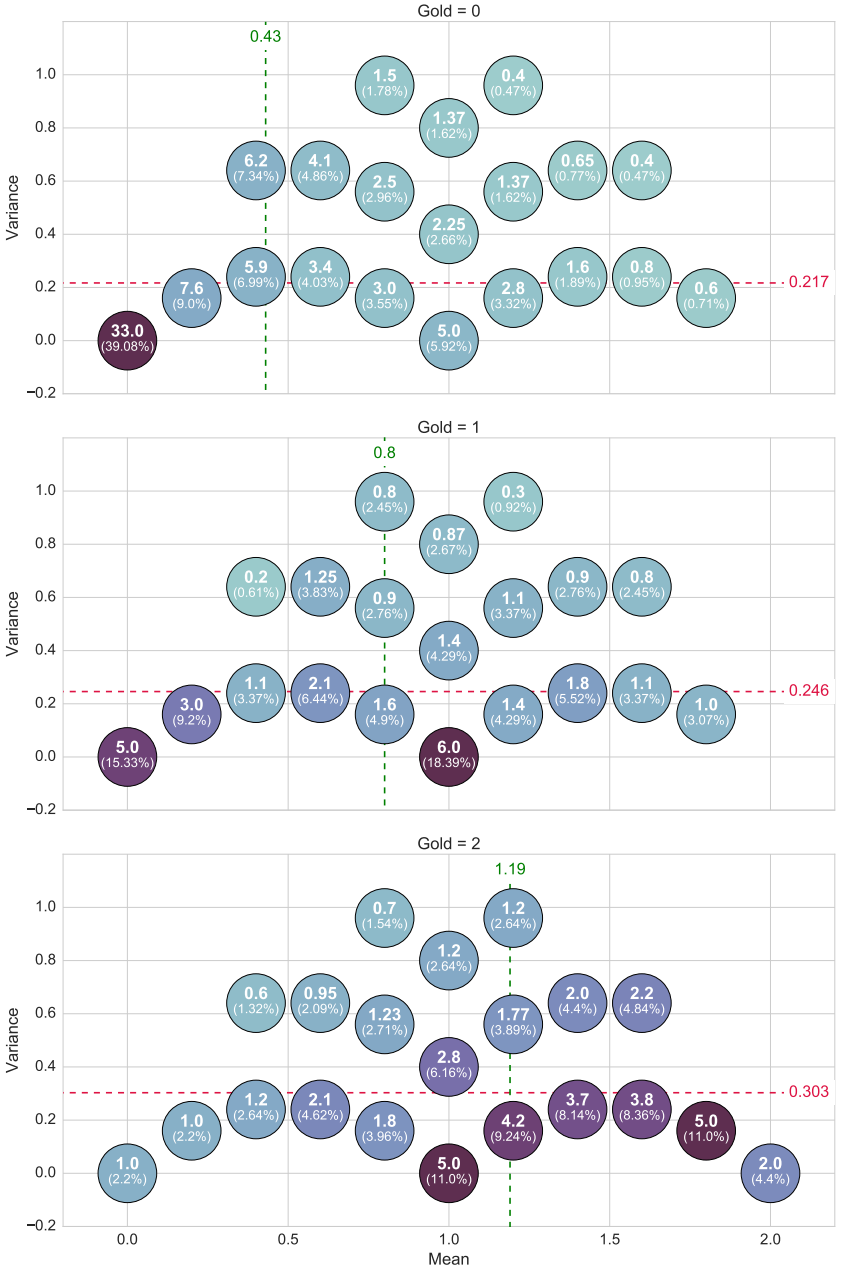


Figure 6.5: The 21 mean-variance pairs obtained normalising the data of the Relevance Feedback 10 collection, breakdown over "NIST" labels.

the difference between the mean of the scores received by not-relevant documents (0.43) to those of highly-relevant (1.19). Before the normalization this difference was 0.4, and after the normalization it has increased to 0.76. The normalisation considerably decreased the variance scores considering documents of three relevance levels. However, this increment is not uniform for all three relevance levels. After normalisation, the variance of not-relevant documents decreased more than that of highly-relevant documents (from 0.445 to 0.217 and from 0.449 to 0.303, respectively). This means that workers had a higher agreement when judging not-relevant documents compared to when judging highly-relevant documents.

6.5 Agreement on the topics of the TREC 2010 Relevance Feedback collection

After observing overall agreement in TREC 2010 Relevance Feedback collection (not equally spread on the different relevance levels), we analyse the agreement focusing on the topics.

6.5.1 Measures: Effectiveness, Ease, Agreement

For each topic we define *topic ease* as the average effectiveness on that topic of the systems participating in the evaluation exercise (we use the Million Query 2009 runs).On terminology, we notice that we prefer to use "topic ease" instead of the probably most common "topic difficulty" for symmetry with system effectiveness: higher e would mean both higher topic ease and higher system effectiveness.

Several agreement measures have been defined in the past: variance, standard deviation, entropy, ICC, Cohen's kappa, Fleiss's kappa, etc. Throughout this work we use Krippendorff's α [76], a standardized measure of agreement that adapts to items having different numbers of evaluators, different scales, missing values. α assumes values ranging from -1 (complete disagreement) through 0 random (agreement obtained by random evaluations) to 1 (complete agreement). We used α interval for the data, and a graded scale $0, 1, 2$. Using the interval version means that we weighted more the difference between 1 and 2 than the difference between 0 and 1. Results are in practice unchanged when using α ordinal.

α works on a set of assessments and computes the overall agreement for that set. Using it to compute the agreement on a single ⟨topic, document⟩ pair would be absolutely not standard and open to criticisms. Therefore our experiments are topic-based.

Table 6.1 summarizes the data. For each topic we have its ease e, its agreement computed using all the assessed documents (α^A, A stands for "All"), and its agreement computed using a subset of the documents having a specific relevance value in the Gold. We have Gold on a three level scale (0, 1, 2), so we can compute α^0, α^1, and α^2. In the following we will also refer to α^{01}, α^{02}, and α^{12} as the α values computed on the basis of documents having to Gold relevance values.

Table 6.1: Datasets. The α superscripts indicate the set of documents used to compute α, i.e., All, or having Gold = 0, Gold = 1, or Gold = 2.

Topic	Ease	Agreement			
	(e)	(α^A)	(α^0)	(α^1)	(α^2)
t_1	e_1	α_1^A	α_1^0	α_1^1	α_1^2
t_2	e_2	α_2^A	α_2^0	α_2^1	α_2^2
...					
t_n	e_n	α_n^A	α_n^0	α_n^1	α_n^2

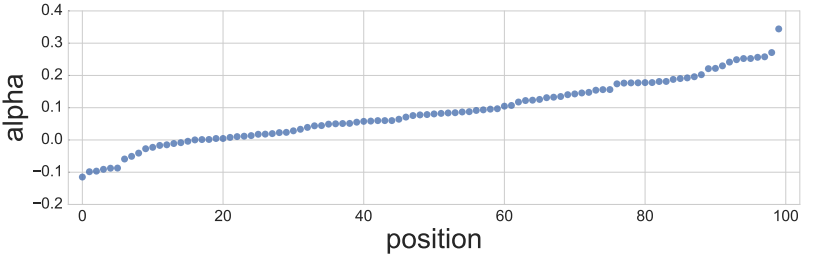


Figure 6.6: Agreement over topics

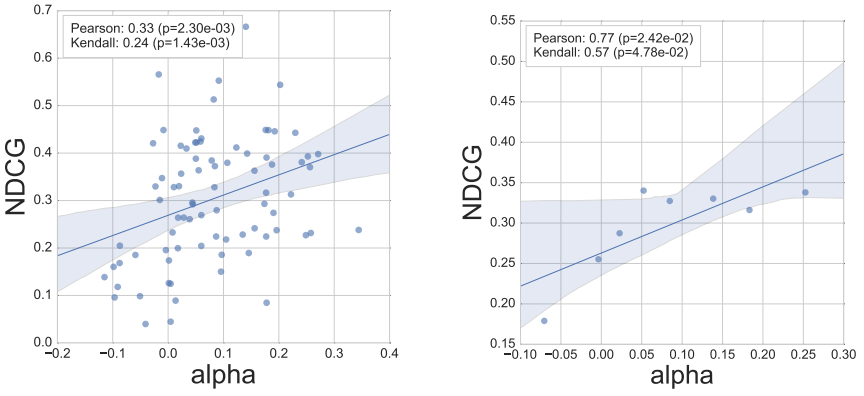
6.5.2 Agreement Over Topics

We analyze the variation of Krippendorff's α over topics. Figure 6.6 shows that there is quite a variation of α values over the 100 topics of the RF dataset: the range is approximately $[-0.1, 0.3]$. After this first analysis, there seems to be a significant effect of topic on agreement.

6.5.3 Relation Between Agreement and Topic Ease/Difficulty

As anticipated, we speak of topic ease. To understand if agreement is related to topic ease we plot for each topic its agreement value (α) against its ease value (e, NDCG in our case). Figure 6.7(a) shows the scatter plot (using 81 topics): there is a mild, though significant, correlation between agreement and topic ease on the dataset. In other terms, topics with a higher agreement among the five assessors also tend to also be easier topics, i.e., systems tend to obtain a higher NDCG on them. The correlation is mild but it becomes stronger (and still statistically significant) when binning the topics, as shown in Figure 6.7(b). In this figure each dot is a bin of topics of similar agreement; the values on the axes are the averages of α and NDCG for the topics in that bin.

We can state that there seems to be some relation between topic ease and agreement, although a positive and significant correlation manifests itself for the dataset.



(a) (b)

Figure 6.7: Agreement and topic ease: individual topics (a) and topics binned according to their agreement (b).

6.5.4 Effect on Effectiveness Measures

We begin by analizing how NDCG values vary when varying the topic subset on the basis of agreement, and also taking into account the relevance level in the Gold.

Figure 6.8 shows scatter plots of and correlations between α and NDCG values. For example, in the bottom-right scatter plot each dot is a topic, its X value is its α^A (agreement computed using all its documents) and its Y value is its NDCG. This is a similar figure to Figure 6.7(a), with only fewer dots because we selected the 71 topics having at least one document for each relevance level and for which we found an NDCG value from the summary files. In the last row individual topics are used; in the previous rows topics are binned, as above after having sorted them by agreement. So the top-right plot is similar to that in Figure 6.7(a), with again the same difference on the number of topics. The previous columns compute α values not on all the documents but on subsets on the basis of relevance values in the Gold.

The figure shows that, although no correlation is found between α^i, α^j, and α^k, neither between α^{ij} and α^k for $k \in \{0, 1, 2\}$, the correlation between agreement (α^A) and ease shown by Figure 6.7 does persist when agreement is computed on a subset of the topics only.

Figure 6.9 provides some further insight. In the top chart, the X axis represent the number of topics used to compute NDCG: at the extreme left all 71 topics are used, and topics are removed while going to the right of the chart. We use three strategies to remove the topics: randomly (with 300 random repetitions), removing the high agreement topics first, and removing the low agreement topic first. The figure shows that when computing NDCG after removing high agreement topics (the blue series) NDCG decreases, as expected since those are the easier topics. Conversely, when removing low agreement topics, NDCG increases. As a baseline, when removing

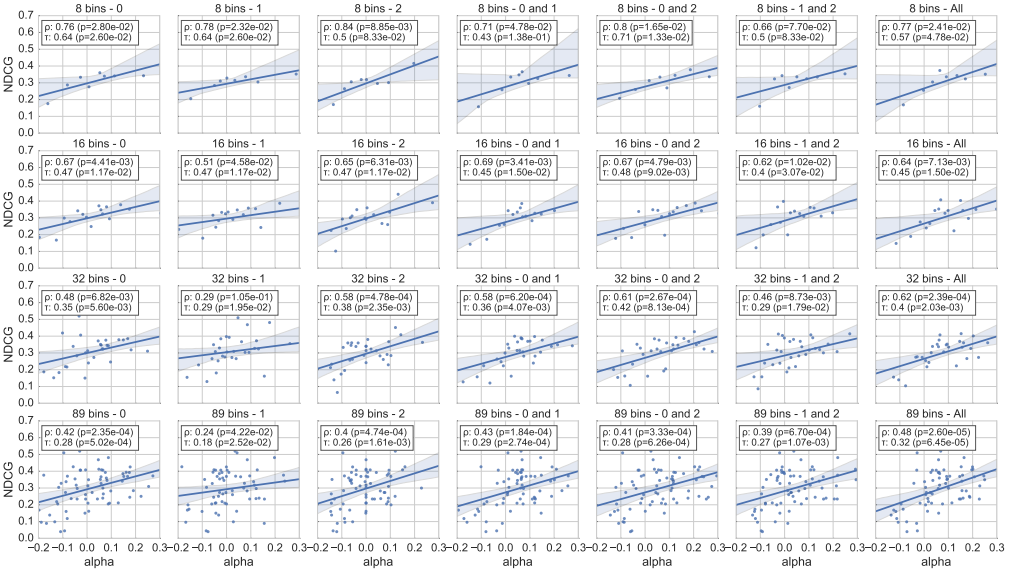


Figure 6.8: α - NDCG scatter plots over relevance levels combinations, for RF data.

randomly chosen topics, NDCG remains constant (with some statistical fluctuation). The plot on the bottom of Figure 6.9, where again topics are binned by agreement, shows the same effect with less fluctuations.

6.5.5 Effect on System Ranks

Having understood that high and low agreement topics impact differently on NDCG values, it is natural to ask how system ranks are likewise affected. We address this issue by trying to predict system ranks by using a subset of topics. In other terms, we compute the Kendall's τ correlation between (i) NDCG computed on the whole topic set and (ii) NDCG computed on a subset formed by selecting the high agreement topics, or the low agreement ones, or random ones (with 300 repertitions). This is a similar methodology to that used by Guiver et al. [50]. Figure 6.10 shows the result. The X axis represents the number of topics in the topic subset (differently from previous figures here cardinality increases while moving right; we can use 71 topics). The three series show the τ values for the three topic subset selection strategies. The figure shows that when using the topics with high agreement one can predict better the system ranks than when using low agreement topics. In other terms, system ranks are more affected by high agreement topics.

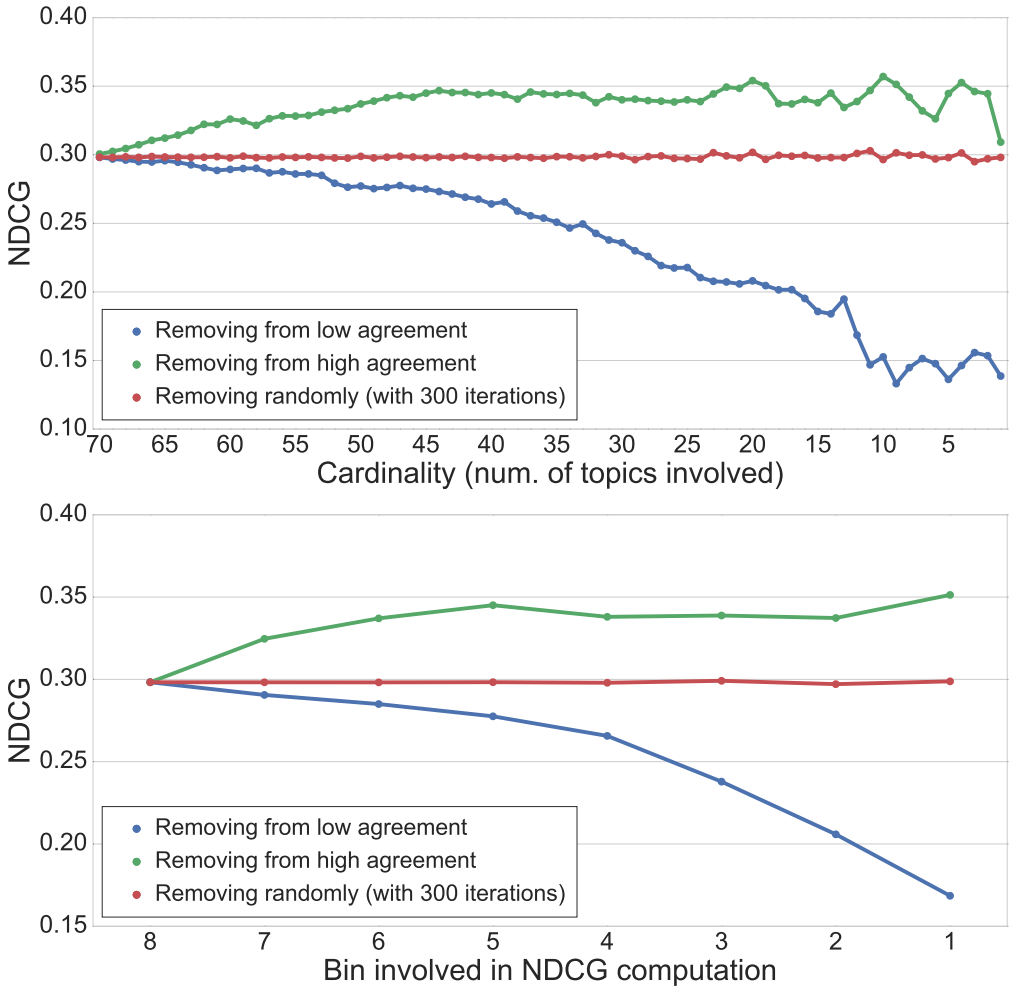


Figure 6.9: NDCG variation when removing topics (with high agreement, low agreement and randomly). Individual topics on the top, whole bins on the bottom.

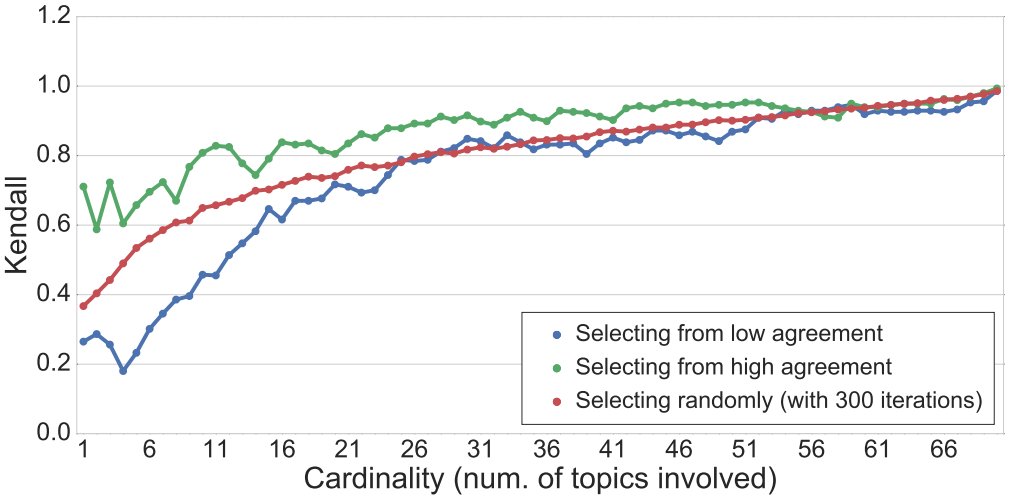


Figure 6.10: Effect of removing topics on system ranks. The Y axis shows the Kendall's τ correlation between the ranks of the systems obtained computing NDCG using the full 81 topics or a subset of the cardinality shown on the X axis. Subsets are formed selecting from low agreement, high agreement, and randomly.

6.6 Conclusion

This chapter discussed results of a preliminary study on agreement.

We analysed the agreement of the TREC 2010 Relevance Feedback collection (introduced in Section 6.1). We discussed the agreement by chance (expected agreement), and the normalisation of the observed agreement. The analysis involves both the agreement for single documents and the agreement of the topics. We also studied the effect of the agreement on the system evaluations. We demonstrate that system evaluations are affected when we exclude from the evaluation topics according to their relevance degrees. Systems scores increase when we exclude topics with a low agreement, vice-versa, the scores decrease when we exclude topics with a high agreement.

7

Conclusions

This chapter is composed by two sections: Section 7.1 which contains the conclusion of the thesis and Section 7.2 that present the future directions.

7.1 Summary of contributions

This thesis focuses on crowdsourcing relevance assessments. We introduced the background of crowdsourcing in Chapter 2 by discussing its definitions and main forms, some different points of view regarding it, and some open issues. Then, we presented the relevance assessment in Chapter 3; starting from the original definition until the most recent experiments that involve workers for the relevance assessment. Then we focused on three specific aspects of the relevance assessment: the *time factor*, the use of *magnitude estimation technique* for expressing relevance judgements, and the *agreement* among workers. Results obtained studying each aspect are summarized in following three sections.

7.1.1 Time factor

The studies concerning the impact of the time factor aspect on relevance assessment process aimed to understand how much time workers require to judge the relevance of a document. This information can result very useful in order to assign to workers the right amount of time for judging the relevance of the documents. This optimization guarantees both workers have enough time to carry out the tasks, and there will be not waste of time (this second aspect allows the requester to money saving). We performed four extensive experiments, with different controlled experimental settings, aimed to understand the effect of time limitations in relevance assessment process. We evaluated crowdsourced judgement quality by comparing results obtained by experiments with standard test collections. Results clearly show that limiting the time to perform a relevance judgment brings benefits both in terms of cost reduction as well as of increased quality. This last

point was unexpected. We observed that the best timeout value to be used lies in the interval of $25 - 30$ seconds and does not depend on topic, document, or crowd.

7.1.2 Magnitude estimation technique

We investigated the suitability of magnitude estimation for gathering relevance judgments. By mean of a large-scale crowdsourcing experiment, we collected over 50,000 magnitude estimation relevance judgments across 18 TREC topics. The scores we gathered are consistent with classical ordinal scores, both on single topics and overall on the 18 aggregated topics. The agreement among judges using magnitude estimation is comparable, if not higher, than judges agreement using classical ordinal and binary scales. Finally, a failure analysis demonstrated that disagreement is often due not to problems with magnitude estimation per se but rather to different judging contexts.

Our analysis suggests that it is important to incorporate different judging scales since users have subjective perceptions of the differences in relevance. Our analysis of user-reported relevance perception showed that the linear profile is close to the "average" user, but the distribution of magnitudes suggests that attempting to fit a single profile, or view of relevance, for system evaluation is unlikely to be sufficient.

When a priori categorical scales are used in relevance judgment tasks, there is no possible way to capture variance in the perception people have of the scale of relevance. In turn, this limits our understanding of how gain should be set in DCG-like metrics, and hence our ability to accurately evaluate systems.

7.1.3 Preliminary results on agreement

The study of agreement between workers, presented in Chapter 6 aims to study the agreement among workers. We analysed the agreement of the TREC 2010 Relevance Feedback collection [17]. We discussed the agreement by chance (expected agreement) and the normalisation of the observed agreement. The analysis involves both the agreement for single documents and the agreement of the topics. We also studied the effect of the agreement on the system evaluations. We demonstrate that system evaluations are affected when we exclude from the evaluation topics according to their relevance degrees. Systems scores increase when we exclude topics with a low agreement, vice-versa, the scores decrease when we exclude topics with a high agreement.

7.2 Future work

The research presented in this thesis can be extended in many directions.

Concerning the time factor studies, new time limitations can be studied. In our work, we only considered upper time limitation. Nevertheless, lower time limitations or hybrid solutions would be an interesting option to study. Also, reading and comprehension speeds are personal and context-dependent issues. Studies focused on the impact of these aspects can lead to customise the assignment of the available amount of time for judging on the basis of workers' characteristics and contexts. For example, each

worker can have a different amount of time for judging depending on his/her reading and comprehension speeds, but also the available judging time could be differently assigned to a worker who has to judge documents in different languages. The magnitude estimation work demonstrates that this technique is particularly suitable for collecting relevant judgements. Our study showed that specific aspects such as the choice of the normalisation technique directly impacts the overall results. The studies on the agreement can be expanded by studying the relations between agreement among workers and quality of the results. Ideally, workers who strongly agree should lead to stable results, but we still do not know if and how it is related to the quality of their work. Also, future explorations of crowdsourcing agreement could lead to the study of new ad hoc techniques for measuring the agreement in crowdsourcing contexts.

A

Logic programming for supporting the generation of data units

This appendix details ASP paradigm to support crowdsourcing data unit generation. The chapter is structured as follows: Section A.1 and Section A.2 introduce unit data and ASP paradigm, respectively. Section A.3 describes a case study which involves a particularly complicated unit generation, and Section A.4 discusses an ASP based solution. Section A.5 considers results, and Section A.6 presents the conclusions.

A.1 Data units

Unit generation is an important issue in crowdsourcing. Sometimes this activity can be complicated to do. Indeed, crowdsourcing experiments can involve wide amounts of data, but workers are required to perform only a slight amount of work. The activity division among workers may be subject to strict limitations and constraints. A data unit file contains specific data that the task assigns to each worker.

A task is run over a set of units. Each unit contains actual parameters used by a specific task instance. When a worker begins a task, the crowdsourcing platform assigns him/her a unit. Workers may wish to repeat a task, and if it is expected by the task's design and there are still available units, the platform assigns a new unit to the worker, who can then restart the task with new data. In mTurk a unit is also called HIT.

The file which contains the list of units is called *data unit*, and typically consists of a table (in *CSV* or *XLSX* format), having units on rows and relative parameters on columns (Table A.1 shows the typical structure of a data unit file).

Unit data design is a crucial process since a little flaw in its creation may compromise

Table A.1: The structure of a data unit file

	Param_1	Param_2	...	Param_N
Unit 1	...	...	...	...
Unit 2	...	...	...	...
...	...	...	...	...
Unit_N	...	...	...	...

the entire experiment. Several strategic factors have to be necessarily considered when designing a crowdsourcing task, such as the activity subdivision modality and their subsequent assignment to the workers. These have to be done avoiding introduction of experimental biases. In simple tasks, this step is easy, and can be carried out by adopting elementary combinatorial techniques, such the "Latin square" or some its variant. On the contrary, more complex tasks which require the satisfaction of multiple constraints may need complicated unit generations that can reveal challenging for the task designers.

A.2 Answer set programming

ASP is a form of declarative programming that offers a simple and powerful modelling language to solve combinatorial problems. These are reduced to computing stable models, then answer set solvers-programs perform searchs of solutions [80].

In this thesis, we use *Clingo*, an effective tool of POTASSCO[1] (Potsdam Answer Set Solving Collection), a set of tools for ASP developed at the University of Potsdam[2]. Clingo is a monolithic system which combines together two important components:

- *Gringo*, a grounder that given an input program with first-order variables, computes an equivalent ground (variable-free) program;
- *Clasp*, which is an answer set solver for (extended) normal and disjunctive logic programs. It combines high-level modelling capacities of ASP with state-of-the-art techniques from the area of Boolean constraint solving.

All POTASSCO tools are written in *C++* and are published under *GNU General Public License(s)*.

A.3 A case study

After a brief introduction of ASP, let us see how to use it in a real scenario; in particular, we consider those of magnitude estimation experiment described in Chapter 5. The task involves 18 topics, each containing a variable amount of documents, and for each of those

[1] http://potassco.sourceforge.net/
[2] http://www.uni-potsdam.de/

we have to collect ten relevance score. Each worker sequentially judges the relevance of seven different documents, five useful for the task, and two, called gold documents, used only for quality check purposes. Also, each worker can perform the task only once per topic; this constraint guarantees that a worker never judges a document more than once. Documents are presented to each worker sequentially, and he/she can never go backwards the sequence. Therefore, the temporal order in which documents are proposed to workers is an important aspect that has to be taken into account (that has to be modelled in the unit data). The order has to be as more variable as possible to avoid creation of biases that may affect the measurements. This variability can be guaranteed by balancing the number of time that document is presented in each position. Balancing has to regards also the two gold standard documents even though these has to occur once every unit (row).

An important optimisation requires solutions that involve the minimum number of workers to avoid wasting of valuable resources like time or money.

The constraints described above makes complicate the unit data generation; in particular, the finding of acceptable solutions since the limitations considerably reduce the search space. It would represent a problem for solutions based on imperative programming.

A.4 The data unit generator

In order to solve the problem of the data unit generation for the given case study (and also for similar problems), we created a data unit generator. The software, written in *Python*, is a wrapper for Clingo which is the component that actually searches acceptable solutions. Given an input consisting in:

- `num_of_docs`: amount of documents that has to be judge;
- `num_judgments_per_doc`: amount of judgments required for each document;
- `pos`: number of documents to show to workers;
- `num_of_gold`: number of gold documents per row.

The generator has to produce valid data units which can be visualised in a *HTML* table for a preliminary graphical check or exported in a *Microsoft Excel*[3] file (in *XSLS* format), ready to be used in the *Crowdflower* platform. Some input combinations do not generate valid solutions, e.g., there are not solution if `num_of_gold` value is higher or equal than the value of `pos`. Figure A.1 shows how looks a valid generated output.

Figure A.2 shows the operation diagram of the generator, which can be summarised in four steps, detailed in the following four subsections.

Pos 1	Pos 2	Pos 3	Pos 4	Pos 5	Pos 6	Pos 7
1	12	17	2	13	20	14
10	1	13	14	2	16	19
13	2	1	17	15	14	8
17	14	11	1	2	15	16
2	18	16	15	1	19	11
20	17	18	2	14	1	3
17	15	14	16	2	8	1
1	20	2	10	16	8	13
19	1	20	12	9	2	18
14	2	1	9	6	16	12
2	12	9	1	11	10	8
11	10	8	13	1	2	9
15	13	2	18	12	1	20
9	13	8	11	2	20	1

(a)

	A	B	C	D	E	F	G
1	pos_1	pos_2	pos_3	pos_4	pos_5	pos_6	pos_7
2	1	12	17	2	13	20	14
3	10	1	13	14	2	16	19
4	13	2	1	17	15	14	8
5	17	14	11	1	2	15	16
6	2	18	16	15	1	19	11
7	20	17	18	2	14	1	3
8	17	15	14	16	2	8	1
9	1	20	2	10	16	8	13
10	19	1	20	12	9	2	18
11	14	2	1	9	6	16	12
12	2	12	9	1	11	10	8
13	11	10	8	13	1	2	9
14	15	13	2	18	12	1	20
15	9	13	8	11	2	20	1
16	1	19	2	12	7	9	13
17	2	1	12	19	18	17	10
18	18	8	1	2	14	12	17
19	2	16	15	1	10	13	20
20	12	10	9	8	1	11	2
21	11	2	10	8	9	1	15
22	12	9	10	11	8	2	1
23	1	11	6	2	7	3	5
24	6	1	2	3	19	18	5
25	8	3	1	4	20	2	7
26	2	6	3	1	5	7	4
27	3	5	4	2	1	7	6
28	6	5	3	7	2	1	4
29	4	6	7	5	3	2	1
30	1	7	5	3	6	4	2
31	4	1	2	5	8	6	3
32	16	3	1	6	4	2	7
33	3	2	7	1	4	5	6
34	5	7	6	4	1	3	2
35	7	4	5	6	3	1	2

(b)

Figure A.1: A portion of *data-unit* generated by using 20 documents, each one has to be judged 10 times by 10 different workers. Rows represent the documents that have to be judged by a single worker, and columns represent the position of a document on temporal succession. Colored cells represent the gold documents (used as *quality checks*); these have to be judged by all the workers. The Figure A.1a offers a HTML visualization of the generator's output useful for a quick graphic check of the generated data, while the Figure A.1b shows how the output appears in *Excel* format, ready for being loaded on Crowdflower platform.

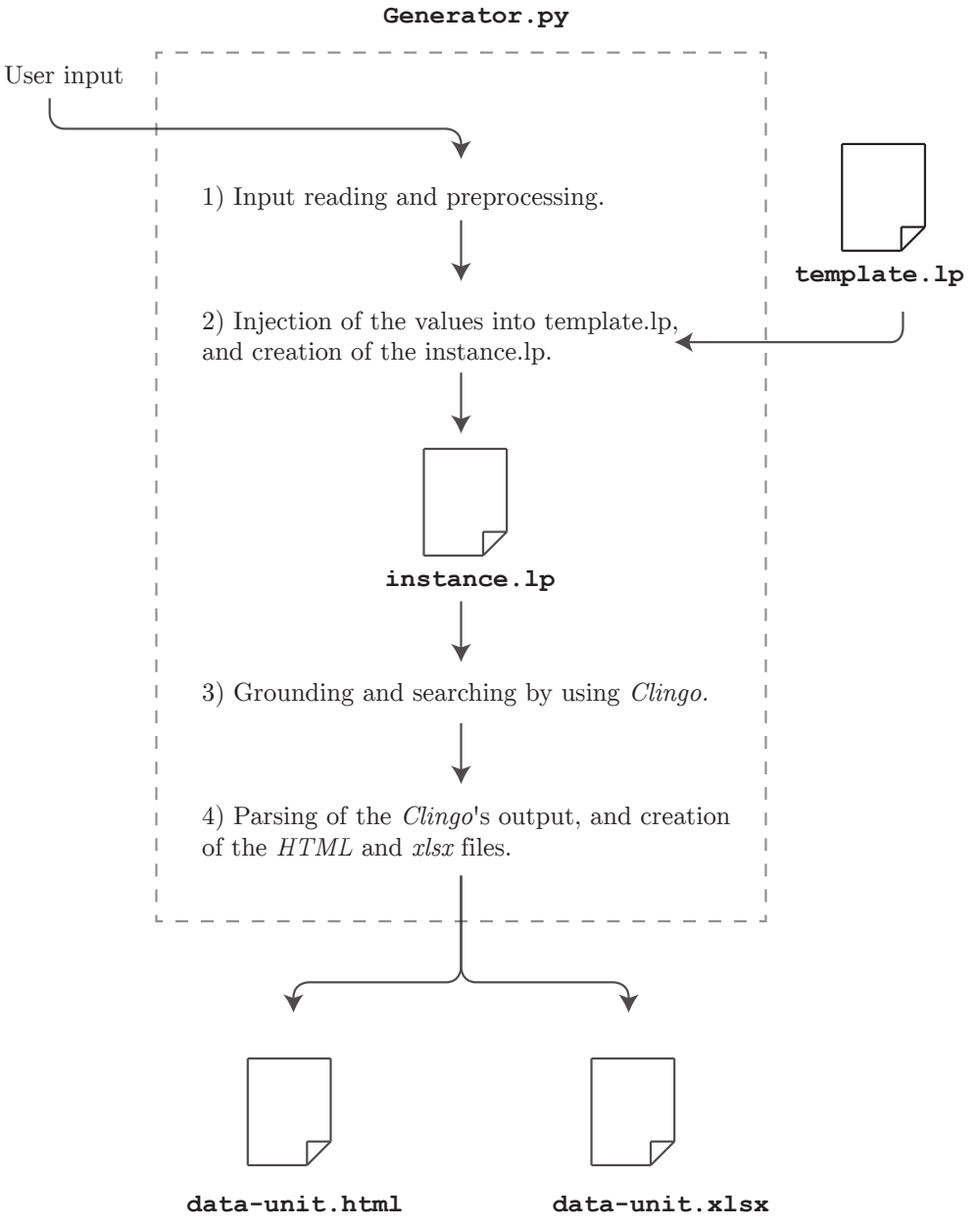


Figure A.2: The operating diagram of the unit-data generator.

A.4.1 Input reading and preprocessing

First phase consists of the input data reading and in their preprocessing, that computes the following parameters:

- minimum amount of rows for obtaining all required judgments;
  `num_of_rows = math.ceil((num_of_docs - num_of_gold ) * judgments_per_doc / (pos - num_of_gold))`
- total amount of available cells in the matrix:
  `num_of_available_cells = num_of_rows * pos`
- amount of cells necessary to contain gold documents:
  `num_of_cells_for_gold_docs = num_of_gold * num_of_rows;`
- amount of cells necessary to contain not-gold documents:
  `num_of_cells_for_not_gold_docs = num_of_not_gold * judgments_per_doc;`
- if in the last row there are any unallocated cells, these are filled with few documents randomly chosen:
  `num_of_not_allocated_cells = num_of_available_cells - num_of_cells_for_gold_docs - num_of_cells_for_not_gold_docs`

Furthermore, the generator computes the following parameters to favour the balancing of documents in matrix columns.

- minimum and maximum number of times that a gold document is placed in a determinate position:
  `min_occs_per_pos_gold = math.floor(num_of_rows / pos)`
  `max_occs_per_pos_gold = math.ceil(num_of_rows / pos)`
- minimum and maximum number of times that a not gold document is placed in a determinate position:
  `max_occs_per_pos_not_gold = math.ceil(judgments_per_doc / pos)`
  `min_occs_per_pos_not_gold = math.floor(judgments_per_doc / pos)`

A.4.2 Set up of the program

In second phase, the generator injects values computed in phase 1 into a file called `template.lp` (in Figure A.3), which consists in a template of program that is going to be launched on Clingo. This operation is carried out by replacing specific placeholders of the template with the computed values. The phase terminates with the creation of the `instace.lp` file, which contains the actual program that will be solved by Clingo. Let us suppose that we input on the generator the values `num_of_docs = 20` and `num_judgments_per_doc = 10`; after preprocessing, the generator produces the

[3] `https://products.office.com/it-it/excel`

```
doc(1..[docs]).
row(1..[rows]).
col(1..[col]).

% Every cell can contain only a value:

1 { cell(X,Y,V) : doc(V) } 1 :- col(X), row(Y).

% A row has to contain all different values:

:- cell(X,Y,V), cell(XX,Y,V), X != XX.

% The value 2 is a gold document and occur on the matrix exactly [rows] times:

1 { cell(X,Y,V) :  col(X) } 1 :- V == 2, row(Y).

% Optimization code:

% Minimum and maximum number of times that a NOT gold document can be placed in a certain column:

[min_occs_per_pos_not_gold] { cell(X,Y,V) : row(Y) } [max_occs_per_pos_not_gold] :- doc(V), V != 1, V != 2, col(X).

% Minimum and maximum number of times that a gold document can be placed in a certain column:

[min_occs_per_pos_gold] { cell(X,Y,V) : row(Y) } [max_occs_per_pos_gold] :-  V == 2, col(X).

% Distribution of a gold doc in a diagonal to reduce the size of the grounding:

cell(Z, Y,1)  :- row(Y), Z = ( (Y-1) \ [col]) + 1.

#show cell/3.
```

Figure A.3: The content of the doc `template.lp`

```
doc(1..20).
row(1..36).
col(1..7).

% Every cell can contain only a value:

1 { cell(X,Y,V) : doc(V) } 1 :- col(X), row(Y).

% A row has to contain all different values:

:- cell(X,Y,V), cell(XX,Y,V), X != XX.

% The value 2 is a gold document and occur on the
    matrix exactly 36 times:

1 { cell(X,Y,V) :  col(X) } 1 :- V == 2, row(Y).

% Optimization code:

% Minimum and maximum number of times that a NOT
    gold document can be placed in a certain column:

1 { cell(X,Y,V) : row(Y) } 2 :- doc(V), V != 1, V !=
    2, col(X).

% Minimum and maximum number of times that a gold
    document can be placed in a certain column:

5 { cell(X,Y,V) : row(Y) } 6 :-  V == 2, col(X).

% Distribution of a gold doc in a diagonal to reduce
    the size of the grounding:

cell(Z,Y,1)  :- row(Y), Z = ( (Y-1) \ 7) + 1.

#show cell/3.
```

Figure A.4: The content of the doc `instance.lp`

instance file shown in Figure A.4. The file contains the ASP program which consists of three main parts:

1. initialization of documents, rows, and columns (Figure A.4, rows 1-3);
2. enunciation of required constraints (Figure A.4, rows 5-15);
3. description of constraints useful for balancing of the documents in the columns, and for reducing the space of search; it allows to speed up the searching process as well as the execution time of the generator (Figure A.4, rows 17-31);

A.4.3 Grounding and solving

The generator invokes Clingo on the file `instance.lp` through a subprocess. After that, Clingo performs grounding and solving, and when both the processes are terminated, it gives the output of computation to the Python generator.

A.4.4 Post processing and plotting

Data unit generation ends with postprocessing and plotting of results. The generator properly parses the output received from Clingo and generates a *HTML* table which is useful for a quick graphical check and an *Excel* file which is the actual unit data ad hoc created ready to be inputted in Crowdflower platform (Figure A.1b).

A.5 Results

The constraints guarantee that all the produced solutions are correct, however, considering the vastness of the search space, the combination of all these limitations may reveal a critic aspect in terms of required time for solution finding. For measuring the constraints impact on the solution finding, we tested the generator by running a simulation that involved data units with up to 200 documents, and groups of 5, 10, and 15 workers. Figure A.5 shows the results of the tests.

The chart shows that the generator can create a data unit which includes 200 documents and 10 workers in less than 20 seconds. Also, the figure illustrates how the generator is much more sensible to the increasing of the number of the workers rather than those of the documents. The time needed for the generation of the data units with 200 documents and 15 workers grows up to a little more than a minute, which is a reasonable and satisfying time.

A.6 Conclusions

In this Appendix, we introduced Answer Set Programming, and we showed how this form of declarative programming can be used to support crowdsourcing data unit generation. We discussed a case study based on the task detailed on Section 5. We presented a

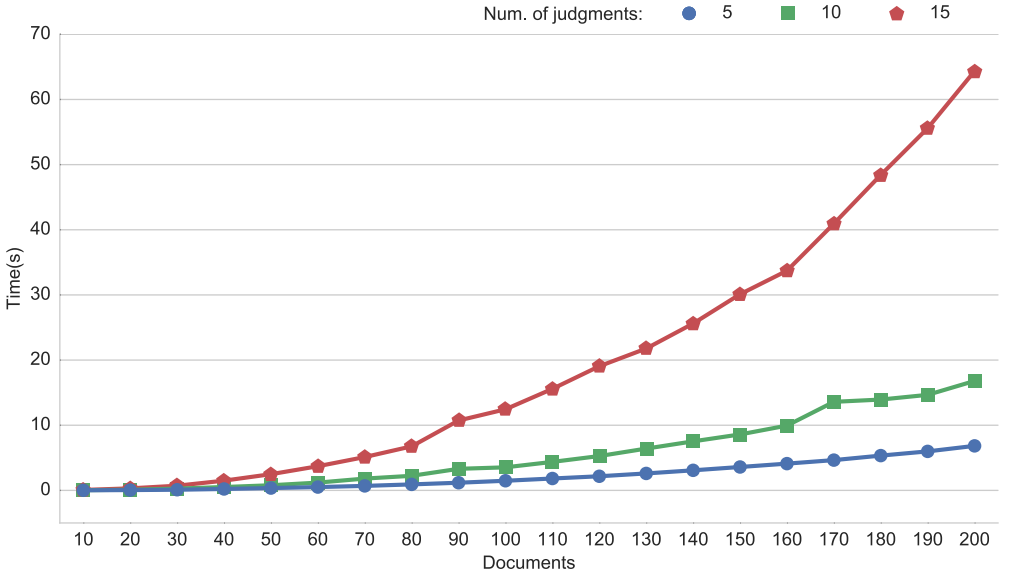


Figure A.5: Chart of execution time for the algorithm, to the growing of the total number of required documents, breakdown over number of judgements for every document.

data unit generator that creates valid user data for the case study. Finally, we discussed results obtained performing a test of the generator.

Bibliography

[1] Ittai Abraham, Omar Alonso, Vasilis Kandylas, Rajesh Patel, Steven Shelford, and Aleksandrs Slivkins. How many workers to ask?: Adaptive exploration for collecting high quality labels. In *Proceedings of the 39th international ACM SIGIR conference on Research and development in information retrieval (SIGIR 2016)*, pages 473–482, New York, NY, USA, 2016. ACM. ISBN 978-1-4503-4069-4.

[2] Niv Ahituv, Magid Igbaria, and Aviem Sella. The effects of time pressure and completeness of information on decision making. *J. Manage. Inf. Syst.*, 15(2): 153–172, September 1998. ISSN 0742-1222.

[3] Mohammad Allahbakhsh, Boualem Benatallah, Aleksandar Ignjatovic, Hamid Reza Motahari-Nezhad, Elisa Bertino, and Schahram Dustdar. Quality control in crowdsourcing systems: Issues and directions. *IEEE Internet Computing*, 17(2):76–81, 2013.

[4] Omar Alonso and Stefano Mizzaro. Can we get rid of trec assessors? using mechanical turk for relevance assessment. In *Proceedings of the 32nd International ACM SIGIR conference on Research and Development in Information Retrieval (SIGIR 2009)*, volume 15, page 16, 2009.

[5] Omar Alonso and Stefano Mizzaro. Using crowdsourcing for TREC relevance assessment. *Information Processing and Management*, 48(6):1053–1066, 2012.

[6] Omar Alonso, Catherine C. Marshall, and Marc Najork. Debugging a crowdsourced task with low inter-rater agreement. In *Proceedings of the 15th ACM/IEEE-CS Joint Conference on Digital Libraries*, JCDL '15, pages 101–110, New York, NY, USA, 2015. ACM. ISBN 978-1-4503-3594-2.

[7] Enrique Amigó, Julio Gonzalo, Javier Artiles, and Felisa Verdejo. A comparison of extrinsic clustering evaluation metrics based on formal constraints. *Information retrieval*, 12(4):461–486, 2009.

[8] Enrique Amigó, Julio Gonzalo, and Stefano Mizzaro. A general account of effectiveness metrics for information tasks: retrieval, filtering, and clustering. In *Proceedings of the 37th International ACM SIGIR conference on Research and Development in Information Retrieval (SIGIR 2014)*, pages 1289–1289. ACM, 2014.

[9] Jesse Anderton, Maryam Bashir, Virgil Pavlu, and Javed A. Aslam. An analysis of crowd workers mistakes for specific and complex relevance assessment task. In *Proceedings of the 22nd ACM Conference on Information and Knowledge (CIKM 2013)*, pages 1873–1876, 2013.

[10] Lora Aroyo and Chris Welty. Crowd truth: Harnessing disagreement in crowdsourcing a relation extraction gold standard. *ACM Web Science 2013*, 2013, 2013.

[11] Solomon E Asch. Studies of independence and conformity: I. a minority of one against a unanimous majority. *Psychological monographs: General and applied*, 70(9):1, 1956.

[12] Yoram Bachrach, Thore Graepel, Gjergji Kasneci, Michal Kosinski, and Jurgen Van Gael. Crowd iq: aggregating opinions to boost performance. In *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems-Volume 1*, pages 535–542. International Foundation for Autonomous Agents and Multiagent Systems, 2012.

[13] Ellen Gurman Bard, Dan Robertson, and Antonella Sorace. Magnitude estimation of linguistic acceptability. *Language*, 72(1):32–68, 1996.

[14] Roi Blanco, Harry Halpin, Daniel M. Herzig, Peter Mika, Jeffrey Pound, Henry S. Thompson, and Thanh Tran Duc. Repeatable and reliable search system evaluation using crowdsourcing. In *Proceedings of the 34th International ACM SIGIR conference on Research and Development in Information Retrieval (SIGIR 2011)*, pages 923–932, 2011.

[15] Gustave Le Bon. *The crowd: A study of the popular mind*. Classic Books Library, 1896.

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

[17] Chris Buckley, Matthew Lease, and Mark D Smucker. Overview of the trec 2010 relevance feedback track. In *Proc. of the 19th Text Retrieval Conference*, 2010.

[18] Chris Burges, Tal Shaked, Erin Renshaw, Ari Lazier, Matt Deeds, Nicole Hamilton, and Greg Hullender. Learning to rank using gradient descent. In *Proceedings of the 22nd international conference on Machine learning*, pages 89–96, Bonn, Germany, 2005.

[19] Vannevar Bush and As We May Think. The atlantic monthly. *As we may think*, 176(1):101–108, 1945.

[20] Ben Carterette and Ian Soboroff. The effect of assessor error on IR system evaluation. In *Proceedings of the 33rd International ACM SIGIR conference on Research*

and Development in Information Retrieval (SIGIR 2010), pages 539–546, Geneva, Switzerland, 2010.

[21] Ben Carterette, Virgiliu Pavlu, Hui Fang, and Evangelos Kanoulas. Million query track 2009 overview. In *TREC*, 2009.

[22] Ben Carterette, Evangelos Kanoulas, Mark Hall, and Paul Clough. Overview of the trec 2014 session track. Technical report, DTIC Document, 2014.

[23] Vitor R Carvalho, Matthew Lease, and Emine Yilmaz. Crowdsourcing for search evaluation. In *Proceedings of the 34th International ACM SIGIR conference on Research and Development in Information Retrieval (SIGIR 2011)*, volume 44, pages 17–22. ACM, 2011.

[24] Olivier Chapelle, Donald Metlzer, Ya Zhang, and Pierre Grinspan. Expected reciprocal rank for graded relevance. In *Proceedings of the 18th ACM Conference on Information and Knowledge (CIKM 2009)*, pages 621–630, Hong Kong, 2009.

[25] Justin Cheng, Jaime Teevan, Shamsi T Iqbal, and Michael S Bernstein. Break it down: A comparison of macro-and microtasks. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI2015)*, pages 4061–4064. ACM, 2015.

[26] Timothy Anatolievich Chklovski and Rada Mihalcea. Exploiting agreement and disagreement of human annotators for word sense disambiguation. In *Recent Advances in Natural Language Processing*, 2003.

[27] Cyril W Cleverdon and Michael Keen. Aslib cranfield research project-factors determining the performance of indexing systems; volume 2, test results. Technical report, Cranfield University, 1966.

[28] Jacob Cohen. Weighted kappa: Nominal scale agreement provision for scaled disagreement or partial credit. *Psychological bulletin*, 70(4):213, 1968.

[29] Kevyn Collins-Thompson, Paul Bennett, Fernando Diaz, Charles LA Clarke, and Ellen M Voorhees. TREC 2013 Web Track Overview. In *22nd Text REtrieval Conference (TREC 2013)*, Gaithersburg, MD, 2014.

[30] Adriel Dean-Hall, Charles L. A. Clarke, Jaap Kamps, Paul Thomas, and Ellen M. Voorhees. Overview of the TREC 2014 contextual suggestion track. In *Proceedings of TREC 2014*, 2014.

[31] Adriel Dean-Hall, Charles L. A. Clarke, Jaap Kamps, Paul Thomas, and Ellen M. Voorhees. Overview of the TREC 2014 contextual suggestion track. In *Proceedings of The Twenty-Third Text REtrieval Conference, TREC 2014, Gaithersburg, Maryland, USA, November 19-21, 2014*, 2014.

[32] Vincenzo Della Mea, Eddy Maddalena, and Stefano Mizzaro. Crowdsourcing to mobile users: A study of the role of platforms and tasks. In *DBCrowd*, pages 14–19, 2013.

[33] Vincenzo Della Mea, Eddy Maddalena, Stefano Mizzaro, Piernicola Machin, and Carlo A Beltrami. Preliminary results from a crowdsourcing experiment in immunohistochemistry. *Diagnostic pathology*, 9(1):1, 2014.

[34] Vincenzo Della Mea, Eddy Maddalena, and Stefano Mizzaro. Mobile crowdsourcing: four experiments on platforms and tasks. *Distributed and Parallel Databases*, 33(1):123–141, 2015.

[35] Gianluca Demartini and Stefano Mizzaro. A Classification of IR Effectiveness Metrics. In *Proceedings of the 28th European Conference on Information Retrieval (ECIR 2016)*, volume 3936 of *LNCS*, pages 488–491, 2006.

[36] Thomas Demeester, Dolf Trieschnigg, Dong Nguyen, Ke Zhou, and Djoerd Hiemstra. Overview of the trec 2014 federated web search track. Technical report, DTIC Document, 2014.

[37] Martin Dillon. Introduction to modern information retrieval: G. salton and m. mcgill. mcgraw-hill, new york (1983). 0-07-054484-0, 1983.

[38] Anca Dumitrache. Crowdsourcing disagreement for collecting semantic annotation. In *Proceedings of the 12th European Semantic Web Conference (ESWC2015)*, pages 701–710. Springer, 2015.

[39] Walter H. Ehrenstein and Addie Ehrenstein. Psychophysical methods. In Uwe Windhorst and Hakan Johansson, editors, *Modern techniques in neuroscience research*, pages 1211–1241. Springer, 1999.

[40] Carsten Eickhoff, Christopher G Harris, Arjen P de Vries, and Padmini Srinivasan. Quality through flow and immersion: gamifying crowdsourced relevance assessments. In *Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval (SIGIR 2012)*, pages 871–880. ACM, 2012.

[41] Michael Eisenberg. Measuring relevance judgements. *Information Processing and Management*, 24:373–389, 1988.

[42] Enrique Estellés-Arolas and Fernando González-Ladrón-De-Guevara. Towards an integrated crowdsourcing definition. *Journal of Information science*, 38(2):189–200, 2012.

[43] Enrique Estellés-Arolas, Raúl Navarro-Giner, and Fernando González-Ladrón-de Guevara. *Crowdsourcing Fundamentals: Definition and Typology*, pages 33–48. Springer International Publishing, 2015. ISBN 978-3-319-18341-1.

[44] Martin Ester, Hans-Peter Kriegel, Jörg Sander, Xiaowei Xu, et al. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Kdd*, volume 96, pages 226–231, 1996.

[45] Urs Fischbacher and Franziska Föllmi-Heusi. Lies in disguisean experimental study on cheating. *Journal of the European Economic Association*, 11(3):525–547, 2013.

[46] Joseph L Fleiss. Measuring nominal scale agreement among many raters. *Psychological bulletin*, 76(5):378, 1971.

[47] Frank Fuchs-Kittowski and Daniel Faust. Architecture of mobile crowdsourcing systems. In *CYTED-RITOS International Workshop on Groupware*, pages 121–136. Springer, 2014.

[48] George Gescheider. *Psychophysics: The Fundamentals*. Lawrence Erlbaum Associates, 3rd edition, 1997.

[49] Catherine Grady and Matthew Lease. Crowdsourcing document relevance assessment with mechanical turk. In *Proceedings of the NAACL HLT 2010 workshop on creating speech and language data with Amazon's mechanical turk*, pages 172–179. Association for Computational Linguistics, 2010.

[50] J. Guiver, S. Mizzaro, and S. Robertson. A few good topics: Experiments in topic set reduction for retrieval evaluation. *ACM Trans. Inform. Systems*, 27(4), 2009.

[51] Martin Halvey and Robert Villa. Evaluating the effort involved in relevance assessments for images. In *Proceedings of the 37th International ACM SIGIR conference on Research and Development in Information Retrieval (SIGIR 2014)*, pages 887–890, 2014.

[52] Li Hao and Daniel Houser. Honest lies. Technical Report 1021, George Mason University, Interdisciplinary Center for Economic Science, 2011.

[53] Andrew F. Hayes and Klaus Krippendorff. Answering the call for a standard reliability measure for coding data. *Communication methods and measures*, 1(1): 77–89, 2007.

[54] Dirk Helbing, Anders Johansson, and Habib Z Al-Abideen. Crowd turbulence: the physics of crowd disasters. *arXiv preprint arXiv:0708.3339*, 2007.

[55] Muhammad Helmy, Marco Basaldella, Eddy Maddalena, Stefano Mizzaro, and Gianluca Demartini. Towards building a standard dataset for arabic keyphrase extraction evaluation. In *Proceedings of the 20th International Conference on Asian Language Processing (IALP 2016)*, Tainan (Taiwan), 2016.

[56] Mehdi Hosseini, Ingemar J. Cox, Natasa Milic-Frayling, Gabriella Kazai, and Vishwa Vinay. On aggregating labels from multiple crowd workers to infer relevance of documents. In *Proceedings of the 34th European Conference on IR Research (ECIR 2012)*, pages 182–194, 2012.

[57] Jeff Howe. The rise of crowdsourcing. *Wired magazine*, 14(6):1–4, 2006.

[58] Nguyen Quoc Viet Hung, Nguyen Thanh Tam, Lam Ngoc Tran, and Karl Aberer. An evaluation of aggregation techniques in crowdsourcing. In *Proceedings of the 14th International Conference on Web Information Systems Engineering (WISE 2013)*, pages 1–15. Springer, 2013.

[59] Oana Inel, Khalid Khamkham, Tatiana Cristea, Anca Dumitrache, Arne Rutjes, Jelle van der Ploeg, Lukasz Romaszko, Lora Aroyo, and Robert-Jan Sips. Crowdtruth: Machine-human computation framework for harnessing disagreement in gathering annotated data. In *International Semantic Web Conference*, pages 486–504. Springer, 2014.

[60] Peter Emil Rerup Ingwersen. *Information Retrieval Interaction*. Taylor Graham, 1992.

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

[62] Korey Johnson and Aga Bojko. How much is too much? Using magnitude estimation for user experience research. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 54(12):942–946, 2010.

[63] Karen Sparck Jones. *Information retrieval experiment*. Butterworth-Heinemann, 1981.

[64] Sanjay Kairam and Jeffrey Heer. Parting crowds: Characterizing divergent interpretations in crowdsourced annotation tasks. In *Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing*, pages 1637–1648. ACM, 2016.

[65] Evangelos Kanoulas and Javed A. Aslam. Empirical justification of the gain and discount function for nDCG. In *Proceedings of the 18th ACM Conference on Information and Knowledge (CIKM 2009)*, pages 611–620, Hong Kong, 2009.

[66] Gabriella Kazai. In search of quality in crowdsourcing for search engine evaluation. In *Proceedings of the 33th European Conference on IR Research (ECIR 2011)*, pages 165–176, 2011.

[67] Gabriella Kazai, Natasa Milic-Frayling, and Jamie Costello. Towards methods for the collective gathering and quality control of relevance assessments. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval (SIGIR 2009)*, pages 452–459. ACM, 2009.

[68] Gabriella Kazai, Jaap Kamps, Marijn Koolen, and Natasa Milic-Frayling. Crowdsourcing for book search evaluation: Impact of hit design on comparative system ranking. In *Proceedings of the 34th International ACM SIGIR conference on Research and Development in Information Retrieval (SIGIR 2011)*, pages 205–214, 2011.

[69] Gabriella Kazai, Jaap Kamps, and Natasa Milic-Frayling. Worker types and personality traits in crowdsourcing relevance labels. In *Proceedings of the 20th ACM Conference on Information and Knowledge (CIKM 2011)*, pages 1941–1944, 2011.

[70] Gabriella Kazai, Jaap Kamps, and Natasa Milic-Frayling. An analysis of human factors and label accuracy in crowdsourcing relevance judgments. *Inf. Retr.*, 16 (2):138–178, April 2013. ISSN 1386-4564.

[71] Diane Kelly et al. Methods for evaluating interactive information retrieval systems with users. *Foundations and Trends in Information Retrieval*, 3(1–2):1–224, 2009.

[72] Jan H. Kietzmann. Crowdsourcing: A revised definition and introduction to new research. *Business Horizons*, 60(2):151 – 153, 2017. ISSN 0007-6813. Crowdsourcing.

[73] Gary G. Koch. Intraclass correlation coefficient. *Encyclopedia of statistical sciences*, 1982.

[74] Jacob Kohen. A coefficient of agreement for nominal scale. *Educ Psychol Meas*, 20:37–46, 1960.

[75] Helena Chmura Kraemer. Ramifications of a population model forκ as a coefficient of reliability. *Psychometrika*, 44(4):461–472, 1979.

[76] Klaus Krippendorff. Computing krippendorff's alpha reliability. *Departmental Papers (ASC)*, 1(1-25-2011), 2011.

[77] Pavel Kucherbaev, Azad Abad, Stefano Tranquillini, Florian Daniel, Maurizio Marchese, and Fabio Casati. Crowdcafe-mobile crowdsourcing platform. *arXiv preprint arXiv:1607.01752*, 2016.

[78] Abhishek Kumar, Kuldeep Yadav, Suhas Dev, Shailesh Vaya, and G. Michael Youngblood. Wallah: Design and evaluation of a task-centric mobile-based crowdsourcing platform. In *Proceedings of the 11th International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services*, MOBIQUITOUS '14, pages 188–197, ICST, Brussels, Belgium, Belgium, 2014. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering).

[79] Matthew Lease. On quality control and machine learning in crowdsourcing. *Human Computation*, 11(11), 2011.

[80] Vladimir Lifschitz. What is answer set programming?. In *AAAI*, volume 8, pages 1594–1597, 2008.

[81] Rensis Likert. A technique for the measurement of attitudes. *Archives of psychology*, 1932.

[82] Jimmy Lin, Miles Efron, Yulu Wang, and Garrick Sherman. Overview of the trec 2014 microblog track. Technical report, DTIC Document, 2014.

[83] Ying-jie LU, Peng-zhu ZHANG, and Jing-fang LIU. Task-oriented talent selection in crowdsourcing [j]. *Journal of Systems & Management*, 1:010, 2013.

[84] Eddy Maddalena and Stefano Mizzaro. The axiometrics project. In *Proceedings of the 5th Italian Information Retrieval Workshop (IIR2014)*, pages 11–15, 2014.

[85] Eddy Maddalena and Stefano Mizzaro. Axiometrics: Axioms of information retrieval effectiveness metrics. In *Proceedings of the 6th EVIA 2014 Workshop , Tokyo, Japan*, pages 17–24, 2014.

[86] Eddy Maddalena, Stefano Mizzaro, Falk Scholer, and Andrew Turpin. Judging relevance using magnitude estimation. In *Proceedings of the 37th European Conference on IR Research (ECIR 2015)*, volume 9022 of *LNCS*, pages 215–220, Vienna, Austria, 2015.

[87] Eddy Maddalena, Marco Basaldella, Dario De Nart, Dante Degl'Innocenti, Stefano Mizzaro, and Gianluca Demartini. Crowdsourcing relevance assessments: The unexpected benefits of limiting the time to judge. In *Proceedings of the 4th AAAI Conference on Human Computation and Crowdsourcing (HCOMP 2016)*, 2016.

[88] Eddy Maddalena, Stefano Mizzaro, Falk Scholer, and Andrew Turpin. On crowdsourcing relevance magnitudes for information retrieval evaluation. *ACM Trans. Inf. Syst.*, 35(3):19:1–19:32, January 2017. ISSN 1046-8188.

[89] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schutze. *Introduction to Information Retrieval*. Cambridge University Press, New York, NY, USA, 2008.

[90] Lawrence Marks. *Sensory Processes: The new Psychophysics*. Academic Press, 1974.

[91] Richard McCreadie, Craig Macdonald, and Iadh Ounis. Crowdsourcing blog track top news judgments at TREC. In *wsdm*, pages 23–26, 2011.

[92] Mick McGee. Usability magnitude estimation. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 47(4):691–695, 2003.

[93] Stefano Mizzaro. Relevance: The whole history. *JASIS*, 48(9):810–832, 1997.

[94] Stefano Mizzaro. How many relevances in information retrieval? *Interacting with computers*, 10(3):303–320, 1998.

[95] Calvin N. Mooers. Zatocoding applied to mechanical organization of knowledge. *American documentation*, 2(1):20–32, 1951.

[96] Howard R. Moskowitz. Magnitude estimation: notes on what, how, when, and why to use it. *Journal of Food Quality*, 1(3):195–227, 1977.

[97] Quoc Viet Hung Nguyen, Thanh Tam Nguyen, Ngoc Tran Lam, and Karl Aberer. Batc: A benchmark for aggregation techniques in crowdsourcing. In *Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2013)*, SIGIR 2013, pages 1079–1080, New York, NY, USA, 2013. ACM.

[98] Stefanie Nowak and Stefan Rüger. How reliable are annotations via crowdsourcing: A study about inter-annotator agreement for multi-label image annotation. In *Proceedings of the International Conference on Multimedia Information Retrieval*, MIR '10, pages 557–566, 2010.

[99] Sri Devi Ravana and Alistair Moffat. Exploring evaluation metrics: Gmap versus map. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval (SIGIR 2008)*, pages 687–688. ACM, 2008.

[100] Stephen Robertson. On GMAP: and other transformations. In *Proceedings of the 15th ACM Conference on Information and Knowledge (CIKM 2006)*, pages 78–83, New York, USA, 2006.

[101] Marta Sabou, Kalina Bontcheva, Leon Derczynski, and Arno Scharl. Corpus annotation through crowdsourcing: Towards best practice guidelines. In *LREC*, pages 859–866, 2014.

[102] Gerard Salton. Automatic text processing: The transformation, analysis, and retrieval of. *Reading: Addison-Wesley*, 1989.

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

[104] Tefko Saracevic. Relevance: A review of and a framework for the thinking on the notion in information science. *Journal of the American Society for Information Science*, 26(6):321–343, 1975.

[105] Linda Schamber. Relevance and information behavior. *Annual review of information science and technology (ARIST)*, 29:3–48, 1994.

[106] F. A. Schmidt. The good, the bad and the ugly: Why crowdsourcing needs ethics. In *2013 International Conference on Cloud and Green Computing*, pages 531–535, 2013.

[107] Falk Scholer, Andrew Turpin, and Mark Sanderson. Quantifying test collection quality based on the consistency of relevance judgements. In *Proceedings of the 34th International ACM SIGIR conference on Research and Development in Information Retrieval (SIGIR 2011)*, pages 1063–1072, Beijing, China, 2011.

[108] Falk Scholer, Eddy Maddalena, Stefano Mizzaro, and Andrew Turpin. Magnitudes of relevance: Relevance judgements, magnitude estimation, and crowdsourcing.

In *The Sixth International Workshop on Evaluating Information Access (EVIA 2014)*, Tokyo, Japan, 2014.

[109] R.J. Senter and E.A. Smith. Automated readability index. Technical report, DTIC Document, 1967.

[110] Aashish Sheshadri and Matthew Lease. SQUARE: A benchmark for research on computing crowd consensus. In *Proceedings of the First Conference on Human Computation and Crowdsourcing, (HCOMP 2013*, 2013.

[111] David Sheskin. *Handbook of parametric and nonparametric statistical procedures, 4th ed.* CRC Press, 2007.

[112] Mark Smucker, Gabriella Kazai, and Matthew Lease. Overview of the TREC 2013 Crowdsourcing Track. In *Proceedings of the 22nd NIST Text Retrieval Conference (TREC)*, 2014.

[113] Mark D. Smucker, Gabriella Kazai, and Matthew Lease. Overview of the TREC 2013 crowdsourcing track. In *Proceedings of TREC 2013*, 2013.

[114] Guillermo Soberón, Lora Aroyo, Chris Welty, Oana Inel, Hui Lin, and Manfred Overmeen. Measuring crowd truth: disagreement metrics combined with worker behavior filters. In *Proceedings of the 1st International Conference on Crowdsourcing the Semantic Web (CrowdSem 2013)*, pages 45–58. CEUR-WS. org, 2013.

[115] Eero Sormunen. Liberal relevance criteria of TREC: Counting on negligible documents? In *Proceedings of the 25th International ACM SIGIR conference on Research and Development in Information Retrieval (SIGIR 2002)*, pages 324–330, Tampere, Finland, 2002.

[116] Amanda Spink and Howard Greisdorf. Regions and levels: Measuring and mapping users' relevance judgments. *JASIST*, 52(2):161–173, 2001.

[117] Stanley Smith Stevens. On the theory of scales of measurement, 1946.

[118] Stanley Smith Stevens. A metric for the social consensus. *Science (New York, NY)*, 151(3710):530–541, 1966.

[119] Siddharth Suri, Daniel G. Goldstein, and Winter A. Mason. Honesty in an online labor market. *Human Computation*, 11:11, 2011.

[120] James Surowiecki. *The wisdom of crowds*. Anchor, 2005.

[121] Rong Tang, William M. Shaw, Jr., and Jack L. Vevea. Towards the identification of the optimal number of relevance categories. *J. Am. Soc. Inf. Sci.*, 50(3):254–264, March 1999. ISSN 0002-8231.

[122] Alberto Tonon, Gianluca Demartini, and Philippe Cudré-Mauroux. Pooling-based continuous evaluation of information retrieval systems. *Inf. Retr. Journal*, 18(5): 445–472, 2015.

[123] Read D. Tuddenham. The influence of a distorted group norm upon individual judgment. *The Journal of Psychology*, 46(2):227–241, 1958.

[124] Cornelis J. van Rijsbergen. Information retrieval, 1979.

[125] Matteo Venanzi, John Guiver, Gabriella Kazai, Pushmeet Kohli, and Milad Shokouhi. Community-based bayesian aggregation models for crowdsourcing. In *Proceedings of WWW '14*, pages 155–164. ACM, 2014.

[126] Manisha Verma and Emine Yilmaz. Characterizing relevance on mobile and desktop. In *European Conference on Information Retrieval*, pages 212–223. Springer, 2016.

[127] Manisha Verma, Emine Yilmaz, and Nick Craswell. On obtaining effort based judgements for information retrieval. In *Proceedings of WSDM '16*, New York, NY, USA, 2016. ACM.

[128] Brian Campbell Vickery. The structure of information retrieval systems. In *Proceedings of the International Conference on Scientific Information*, volume 2, pages 1275–1290, 1959.

[129] Brian Campbell Vickery. Subject analysis for information retrieval. In *Proceedings of the International Conference on Scientific Information*, volume 2, pages 855–865, 1959.

[130] Robert Villa and Martin Halvey. Is relevance hard work?: Evaluating the effort of making relevant assessments. In *Proceedings of the 36th International ACM SIGIR conference on Research and Development in Information Retrieval (SIGIR 2013)*, pages 765–768, 2013.

[131] Della Mea Vincenzo, Maddalena Eddy, and Mizzaro Stefano. Crowdsourcing mitosis count: an experiment on the mitos dataset. In *Proc. Of 12th European Congress on Digital Pathology*, 2014.

[132] Ellen M. Voorhees and Donna K. Harman. Overview of the Eighth Text REtrieval Conference (TREC-8). In *The Eighth Text REtrieval Conference (TREC-8)*, pages 1–24, Gaithersburg, MD, 1999.

[133] Ellen M. Voorhees and Donna K. Harman. *TREC: experiment and evaluation in information retrieval*. MIT Press, 2005.

[134] Jianqiang Wang. Accuracy, agreement, speed, and perceived difficulty of users relevance judgments for e-discovery. In *Proceedings of SIGIR Information Retrieval for E-Discovery Workshop*, 2011.

[135] William Webber and Jeremy Pickens. Assessor disagreement and text classifier accuracy. In *Proceedings of the 36th International ACM SIGIR conference on Research and Development in Information Retrieval (SIGIR 2013)*, pages 929–932, Dublin, Ireland, 2013.

[136] Emine Yilmaz, Manisha Verma, Nick Craswell, Filip Radlinski, and Peter Bailey. Relevance and effort: An analysis of document utility. In *Proceedings of the 23rd ACM Conference on Information and Knowledge (CIKM 2014)*, pages 91–100, 2014.

[137] Omar F. Zaidan and Chris Callison-Burch. Crowdsourcing translation: Professional quality from non-professionals. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, HLT '11, pages 1220–1229, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics. ISBN 978-1-932432-87-9.

[138] Xiang Zhang, Guoliang Xue, Ruozhou Yu, Dejun Yang, and Jian Tang. You better be honest: Discouraging free-riding and false-reporting in mobile crowdsourcing. In *2014 IEEE Global Communications Conference*, pages 4971–4976. IEEE, 2014.

[139] Yinglong Zhang, Jin Zhang, Matthew Lease, and Jacek Gwizdka. Multidimensional relevance modeling via psychometrics and crowdsourcing. In *Proceedings of the 37th International ACM SIGIR conference on Research and Development in Information Retrieval (SIGIR 2014)*, pages 435–444. ACM, 2014.

[140] Guido Zuccon, Teerapong Leelanupab, Stewart Whiting, Emine Yilmaz, Joemon M Jose, and Leif Azzopardi. Crowdsourcing interactions: using crowdsourcing for evaluating interactive information retrieval systems. *Information retrieval*, 16 (2):267–305, 2013.