Optimally Selecting and Combining Assessment and Assessor Types for Information Retrieval Evaluation

A dissertation presented
by

Maryam Bashir

to the Faculty of the Graduate School
of the College of Computer and Information Science
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy

Northeastern University
Boston, Massachusetts
July, 2014
THESIS TITLE: Optimally Selecting and Combining Assessment and Assessor Types for Information Retrieval Evaluation

AUTHOR: Maryam Bashir

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

Thesis Advisor:                      Date: 6/22/15

Thesis Reader:                      22 June 2015

Thesis Reader:                      Date

Thesis Reader:                      Date

GRADUATE SCHOOL APPROVAL:

Director, Graduate School

Date: July 21, 2015

COPY RECEIVED IN GRADUATE SCHOOL OFFICE:

Recipient's Signature

Date: 7/8/2015

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

First and foremost, all the praises be to Allah (God), the Lord of the Alamin (worlds i.e. mankind and all that exists).

A successful work is never a one person job. It is the result of the support of many people. Here, I am going to thank those people without whom I might not have been able to finish my PhD successfully.

I am honored to work under the supervision of my advisor Professor Javed A. Aslam and I would like to express my sincere gratitude to him for his academic guidance, motivation, encouragement, and support. I have been fortunate to have an advisor who gave me the freedom to explore on my own. Professor Javed A. Aslam dedicated his time and effort throughout my PhD studies. He has been actively interested in my work and has always been available to advise me.

I would like to thank my thesis committee Dr. Mark D. Smucker, Dr. David A. Smith, and Dr. Yizhou Sun for their valuable comments on my thesis proposal.

I express sincere thanks to research scientist Dr. Virgil Pavlu for his guidance, constructive critiques and valuable suggestions to shape the direction of this work. Many thanks go to my labmates Jesse Anderton, Peter Golbus and Pavel Metrikov for their help, support and research collaboration during last two years. I would also like to thank my labmate Matthew Ekstrand-Abueg for managing our lab’s Linux server.

Finally, I would like to express my deepest gratitude to my parents for their unconditional love and encouragement. Their continuous support in pursuing my study is invaluable to me. I would also like to thank my sister and brothers for their encouragement and friendship. Last but not the least, I would like to thank my husband Irfan for his love, support, care, companionship, and patience during the past few years.
Abstract

Current test collection construction methodologies for Information Retrieval evaluation generally rely on large numbers of document relevance assessments, obtained from experts at great cost. Recently, the use of inexpensive crowd workers has been proposed instead. However, while crowd workers are inexpensive, their assessments are also generally highly inaccurate, rendering their collective assessments far less useful than those obtained from experts in the traditional manner. Our thesis is that instead of using either experts or crowd workers, one can obtain the advantages of both—inexpensive and accurate assessments—by optimally combining them. Another related problem in Information Retrieval evaluation is asking right kind of question to the assessors for the collection relevance judgments. Traditional methods of collecting relevance judgments are based on collecting binary or graded nominal judgments, but such judgments are limited by factors such as inter-assessor disagreement and the arbitrariness of grades. Previous research has shown that it is easier for assessors to make pairwise preference judgments. However, unless the preferences collected are largely transitive, it is not clear how to combine them in order to obtain document relevance scores. Another difficulty is that the number of pairs that need to be assessed is quadratic in the number of documents. We show how to combine a linear number of pairwise preference judgments from multiple assessors to compute relevance scores for every document. We propose a general Bayesian framework for leveraging disparate categories of workers on the cost-accuracy scale, asking the right worker the right kind of question at the right time in order to obtain accurate assessments most cost-effectively for test collection construction. Experiments with Mechanical Turks and expert assessors show promising results for our framework.
# Contents

Abstract i  
List of Tables vii  
List of Figures ix  

## 1 Introduction 1  
1.1 Research Problems 4  
1.1.1 Combining Assessments from Experts and Crowd workers 4  
1.1.2 Preferences 6  
1.2 Research Questions 7  
1.3 Contribution and Results 8  
1.3.1 Comparison of Preferences and Nominal Assessments 8  
1.3.2 Learning Contexts for Predicting Crowd Worker’s Assessment Quality 8  
1.3.3 Combining Preferences to Get A Rank List 8  
1.3.4 Framework for Optimally Combining Assessments from Crowd Workers and Experts 9  
1.4 Structure of Thesis 9  

## 2 Background on Information Retrieval Evaluation 11  
2.1 Information Retrieval Evaluation 11  
2.1.1 Test Collections 11  
2.1.2 Evaluation of Rank Lists 12  
2.1.3 Evaluation of Relevance Assessments 14  

## 3 Related Work 17  
3.1 Crowd Sourcing 17  
3.1.1 Crowd Sourcing for Information Retrieval Evaluation 18
5.5 Analysis .................................................. 64
  5.5.1 Weaknesses in Design of Matches for Elo Algorithm .... 64
  5.5.2 Wrong Judgements by Crowd Workers ................. 65
  5.5.3 Assessor Agreement ................................. 67
5.6 Summary .................................................. 68

6 A Bayesian Framework for Optimally Combining Assessments from Crowd
Workers and Experts 69
  6.1 Motivation .............................................. 69
  6.2 Framework for Optimal Selection of Questions and Assessors . 71
    6.2.1 Modeling Crowd Workers .......................... 71
    6.2.2 Combining Crowd Worker Assessments .............. 72
    6.2.3 Selecting Crowd Worker Assessments ................. 73
  6.3 Experiments ............................................. 77
    6.3.1 Data ................................................ 78
    6.3.2 Contextual Confusion Matrices ....................... 79
    6.3.3 Prior of Relevance Grades .......................... 80
    6.3.4 Off-line Collection of Expert Assessments .......... 81
    6.3.5 Baselines ........................................... 82
  6.4 Results ................................................ 83
    6.4.1 Quality of Rank List ............................... 83
    6.4.2 System Evaluations ................................ 84
    6.4.3 Analysis ........................................... 85
  6.5 Framework for Preferences ............................... 86
    6.5.1 Combining Preferences and Nominal Assessments .... 88
    6.5.2 Experiments ........................................ 88
  6.6 Summary ................................................ 89

7 Conclusion and Future Work 103
  7.1 Results of the research ................................ 103
    7.1.1 Analysis of Crowd Worker’s Assessments ........... 103
    7.1.2 Combining Preferences to Get A Rank List .......... 104
    7.1.3 Framework for Combining Assessments from Crowd Work-
    ers and Experts ........................................... 105
  7.2 Evaluation .............................................. 105
    7.2.1 Analysis of Crowd Worker’s Assessments ........... 106
    7.2.2 Combining Preferences to Get A Rank List .......... 106
7.2.3 Framework for Combining Assessments from Crowd Workers and Experts ........................................... 106
7.3 Conclusions and Discussion .................................................. 107
7.4 Future Work ................................................................. 108

A Participation in TREC 2013 Crowdsourcing Track .......... 111
A.1 Introduction .............................................................. 111
A.2 Methodology ............................................................ 112
  A.2.1 Pivot Selection ...................................................... 113
  A.2.2 Linear Search for Each Document ............................... 113
  A.2.3 Grades from Preference ........................................... 114
A.3 Experiments .............................................................. 114
  A.3.1 Interface Design ................................................... 114
  A.3.2 Data ................................................................. 116
A.4 Results ................................................................. 116
A.5 Conclusion ............................................................... 116

Bibliography ................................................................. 119
## List of Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>Data Retrieval OR Information Retrieval?</td>
<td>2</td>
</tr>
<tr>
<td>2.1</td>
<td>Contingency Table</td>
<td>13</td>
</tr>
<tr>
<td>4.1</td>
<td>Distribution of assessments among crowd workers for preference pairs</td>
<td>36</td>
</tr>
<tr>
<td>4.2</td>
<td>Distribution of assessments among crowd workers for nominal judgments</td>
<td>36</td>
</tr>
<tr>
<td>4.3</td>
<td>Recall and precision of preference judgments</td>
<td>38</td>
</tr>
<tr>
<td>4.4</td>
<td>Sensitivity and specificity of assessments for nominal judgments</td>
<td>39</td>
</tr>
<tr>
<td>4.5</td>
<td>Averages of time spent per assessment for nominal judgments</td>
<td>41</td>
</tr>
<tr>
<td>4.6</td>
<td>Averages of time spent per assessment for preference judgments</td>
<td>41</td>
</tr>
<tr>
<td>5.1</td>
<td>Accuracy (AUC) results using Elo rating system with various enhancements</td>
<td>60</td>
</tr>
<tr>
<td>5.2</td>
<td>Evaluation Results using Mean Average Precision for preference based relevance judgements (EloEM) and graded relevance judgements (average grades, and mean grade using EM)</td>
<td>61</td>
</tr>
<tr>
<td>5.3</td>
<td>Average and standard deviation of number of matches played by a document per topic</td>
<td>65</td>
</tr>
<tr>
<td>5.4</td>
<td>Outcome of all matches played by some non-relevant documents that have high Elo score for topic ID 446.</td>
<td>66</td>
</tr>
<tr>
<td>5.5</td>
<td>Inter-assessor agreement for preference judgements</td>
<td>68</td>
</tr>
<tr>
<td>5.6</td>
<td>Inter-assessor agreement for graded judgements</td>
<td>68</td>
</tr>
<tr>
<td>6.1</td>
<td>Contexts for Confusion Matrices</td>
<td>81</td>
</tr>
<tr>
<td>6.2</td>
<td>Confusion Matrix of conditional probabilities (per row) for Topic 221 and Context 1 “low number of query terms and short document length”</td>
<td>81</td>
</tr>
</tbody>
</table>
6.3 Confusion Matrix of conditional probabilities (per row) for Topic 221 and Context 1 “low number of query terms and short document length” (Preferences) . . . . . . . . . . . . . . . . . . . . . . . . . . . . 82

A.1 This table shows per-topic statistics and overall averages for the run NEUPivot1 and median score for 11 runs submitted to crowdsourcing track. The metrics GAP, ERR@20, AP-correlation and RMSE are listed for each topic. Note that for row all, (i) GAP is the mean gap over all 10 topics, (ii) APCorr and RMSE depend on the ranking of runs induced by the mean ERR20 for all the 10 topics. . . . . . . . . . . . . . . . 117
List of Figures

1.1 Process of Information Storage and Retrieval ........................................... 3
1.2 Average assessing cost VS quality, per document, separately for different assessors. ................................................................. 5

4.1 Screenshot of preference pair of documents with topic keywords and description ................................................................. 33
4.2 Interface for collection of graded judgements ........................................... 34
4.3 Quality of workers versus time spent for pair preferences. The y axis shows the distribution over average fraction of correct answers per worker. Workers are split into three groups based on their average durations. T-tests were run between the fastest and slowest groups; asterisks indicate low p-value ($p < 0.05$). .................................................. 39
4.4 Quality of workers versus time spent for relevance grades. The y axis shows the distribution over average fraction of correct answers per worker. Workers are split into three groups based on their average durations. T-tests were run between the fastest and slowest groups; asterisks indicate low p-value ($p < 0.05$). .................................................. 40
4.5 Document Length vs. Error rate for relevance grades. Longer relevant documents were assessed more accurately. ......................... 42
4.6 Document Length vs. Error rate for pair preferences. Error increased when on pairs of a non-relevant document VS a shorter relevant document. ................................................................. 43
4.7 TF-IDF vs. Error for relevance grades. Only half of the topics show a statistically significant result ($p < 0.05$) for TF-IDF. ......................... 43
4.8 TF-IDF vs. Error for pair preferences. Workers preferred non-relevant documents over relevant documents with lower TF-IDF scores. ....... 44
4.9 Title Relevance vs. Error. Non-relevant documents are frequently mislabeled when their titles are rich in query terms. ......................... 45
4.10 Title Relevance vs. Error. Non-relevant documents are frequently preferred over relevant documents when their titles are rich in query terms.

5.1 Relationship of Number of Elo rating iterations to percent of pairs inverted, separately for each query.

5.2 Comparison of AUC scores using EloEm method, mean grades using EM and Median of TREC participant runs.

5.3 Comparison of LAM scores using EloEm method, mean grades using EM and Median of TREC participant runs.

6.1 System flow of online bayesian framework.

6.2 Interface for Amazon MT workers: instructions(top); document link and grade form (bottom).

6.3 Graded Average Precision: Comparison of framework with baselines using only one assessor (expert or crowd worker).

6.4 Graded Average Precision: Comparison of framework with baseline method for combining assessments of experts and crowd workers.

6.5 System Evaluations (Tau AP): Comparison of framework with baselines using only one assessor (expert or crowd worker).

6.6 System Evaluations (Tau AP): Comparison of framework with baseline method for combining assessments of experts and crowd workers.

6.7 Root Mean Squared Error (RMSE): Comparison of framework with baselines using only one assessor (expert or crowd worker).

6.8 Root Mean Squared Error (RMSE): Comparison of framework with baseline method for combining assessments of experts and crowd workers.

6.9 Comparison of difference between true grade and expected grade between documents selected for sending to experts and documents not selected for sending to experts. Lower average indicates better quality from crowdsourcing alone, thus less benefit from expert assessment.

6.10 Comparison of average variance in crowd worker assessments between documents selected for sending to experts and documents not selected for sending to experts. Lower average indicates better quality from crowdsourcing alone, thus less benefit from expert assessment.
6.11 Graded Average Precision: Comparison of framework (using preferences) with baselines using only one assessor type (expert or crowd worker). .......................... 100
6.12 Graded Average Precision: Comparison of framework (using preferences and nominal judgments) with baselines using only one assessor type (expert or crowd worker). .......................... 101
A.1 Preference pair selection interface with topic keywords and description115
A.2 NEUPivot1-basic-ERR@20 vs qrels.basic-ERR@20. qrels.basic is the TREC 2013 web track qrels reduced to topics 202, 214, 216, 221, 227, 230, 234, 243, 246, and 250 .................................................. 117
Chapter 1

Introduction

In today’s world information is becoming increasingly accessible through internet and search engines play a huge role in making it possible. Hundreds of millions of people engage in information retrieval every day when they use a web search engine or search their email. Manning [2] has formally defined information retrieval (IR) as:

“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).”

Information retrieval is a widely used term and its meaning can be very broad. Information retrieval is conventionally used as a description of the kind of work done by Cleverdon [3], Salton [4], Sparck Jones [5], Lancaster [6] and others. In general it refers to the kind of retrieval that can be described by simply substituting “document” for “information”. A straightforward definition of IR is given by Lancaster [6]:

“An information retrieval system does not inform (i.e. change the knowledge of) the user on the subject of his inquiry. It merely informs on the existence (or non-existence) and whereabouts of documents relating to his request.”

Rijsbergen [1] has given some distinguishing properties of data and information retrieval to clarify the difference between Data Retrieval (DR) and IR. These properties are given in Table 1.1. In DR, artificial kind of query language is used and we are looking for an exact match, whereas in IR, natural language is used as query and small errors in matching generally do not affect performance of the system significantly. In DR, deductive type of inference is used, i.e. $a < b$ and $b < c$ then $a < c$. In IR, inductive inference is used more commonly and relations are only specified with a degree of certainty or uncertainty. Thus DR can be described
as deterministic and IR as probabilistic. Baye’s Theorem is frequently used in IR for inferences, but probabilities are not used in DR.

In today’s world availability of vast amount of information has made storage and retrieval of information a difficult task. Information storage and retrieval is a simple process in principle. Suppose a person has a collection of documents. The person formulates a query and the documents in collection have an answer to his/her query. He can obtain the answer by reading all documents in collection one by one, retaining the relevant documents and discarding all the others. This will solve retrieval problem perfectly but in reality the person might not have enough time to read all the documents apart from the fact that it may be physically impossible for him to do so. With the availability of high speed computers tasks of storage and retrieval of vast amount of information can be designated to computers. But storing natural language of text has caused problems for document representation and storage. Computers are not able to understand natural language so we need mechanism for different representations of document text which computers are able to process. Information storage and retrieval process can be divided into four major steps. Figure 1.1 shows different components of information retrieval and storage system. I will briefly discuss each of these steps in the following:

**Storage and Representation**  Computerized retrieval systems only store representations of documents and the text of documents is lost once it is processed for the purpose of retrieval. Most common representation is “bag of words” representation, disregarding grammar and even word order but keeping multiplicity.

**Indexing**  The vast amount of information from documents must be compressed and stored in an efficient manner so that the retrieval system can find relevant documents for a search query at fast speed. In the absence of an index, the retrieval system would scan every document in the collection, which would require considerable time and computing power. For example, while an index of 10,000

<table>
<thead>
<tr>
<th>Property</th>
<th>Data Retrieval (DR)</th>
<th>Information Retrieval (IR)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Matching</td>
<td>Exact match</td>
<td>Partial match or best match</td>
</tr>
<tr>
<td>Inference</td>
<td>Deduction</td>
<td>Induction</td>
</tr>
<tr>
<td>Model</td>
<td>Deterministic</td>
<td>Probabilistic</td>
</tr>
<tr>
<td>Query Language</td>
<td>Artificial</td>
<td>Natural</td>
</tr>
<tr>
<td>Query specification</td>
<td>Complete</td>
<td>Incomplete</td>
</tr>
<tr>
<td>Items wanted</td>
<td>Matching</td>
<td>Relevant</td>
</tr>
<tr>
<td>Error response</td>
<td>Sensitive</td>
<td>Insensitive</td>
</tr>
</tbody>
</table>

Table 1.1: Data Retrieval OR Information Retrieval? [1]
documents can be queried within milliseconds, a sequential scan of every word in 10,000 large documents could take hours.

**Retrieval** When a user enters a search query, the retrieval system processes the query, looks up the index, applies some retrieval model (a simple retrieval model can be function of frequency of query words and document length), and assigns a score to each document based on its retrieval model. The documents are then sorted according to the score and a rank list of documents is presented to the user. In an operational system the story ends here. However, in an experimental retrieval system it leaves the evaluation to be done.

**Evaluation** In order to improve the quality of retrieval system it is important to measure the current quality of retrieval system. Traditionally retrieval evaluation has been done by employing human assessors who assess relevance of documents in the rank list produced by retrieval system. Modern day search engines use input from user interaction with the search engine and infer relevance information from their clicks on retrieved documents. But search engines also employ human assessors when they need accurate assessments.

Evaluation of retrieval systems has proven to be very challenging task. Senko [7] in an excellent survey paper states: “Without a doubt system evaluation is the most
troublesome area in ISR ...”. There has been much debate about reliability of relevance judgments made by human beings as human beings are prone to error. This thesis is about evaluation of retrieval systems especially about the cost and efficiency of the process related to collection of relevance assessments from human beings. In the following section, we discuss the evaluation research problems addressed in this thesis.

1.1 Research Problems

1.1.1 Combining Assessments from Experts and Crowd Workers

Test collections are a necessary part of experimental retrieval evaluation. A test collection is made up of a collection of documents, some search queries, and relevance assessments (labelling of documents according to their relevance to the search queries). Progress with test collection and the standardization of evaluation for Information Retrieval has enabled great improvements in algorithm efficacy and research robustness over the last 30 years. The problem of test collection construction is critical for the accurate assessment of existing retrieval techniques, for creating new retrieval techniques, and for generally fostering research and development in the field of Information Retrieval (IR). Obtaining such test collections in an accurate and cost-effective manner is thus important, especially for researchers, groups, or businesses without the resources of a governmental agency or large multi-national corporation. IR evaluation, like many other data-driven fields, requires human feedback in order to calculate performance metrics for retrieval systems. This feedback is expensive and not scalable with the explosive growth of document collections; even research collection construction methodologies for retrieval evaluation require large numbers of relevance assessments.

In response to this problem, crowdsourcing has been explored recently in order to gather assessments more cost-effectively. Crowdsourcing is a process for distributing tasks to a group of workers through outsourcing and is generally used to accomplish human intelligence tasks quickly and at low cost. However, while relevance assessments from crowd workers are inexpensive, they also tend to be inaccurate—to the point that often the collected relevance assessments are not particularly useful.

Much research has focused on eliminating spammers and improving the quality of crowdsourced relevance assessments, but these efforts have not been able to yield high quality assessments from crowd workers alone. Figure 1.2 shows the general trend of cost vs. quality for document relevance assessment for different assessor types. The cost and accuracy values are approximations of actual cost and
accuracy of different types of assessors. For example, the cost and accuracy for crowd worker and graduate student is the actual cost and accuracy of assessments by crowd workers and graduate student in our experiments. The cost and accuracy of expert is in line with TREC (Text Retrieval Conference) assessment efforts and budgets over the years. In practice, one expert judgment is typically good enough for practical purposes (standing at 90% accuracy), while crowdsourced assessments require redundancy, adversarial protection, quality estimation, and so on. This raises numerous questions such as: How many crowd assessments do I need for each document? Which documents would benefit the most from additional crowd assessments? How many crowd assessments should I seek before deciding that there is too much disagreement among crowdworkers to obtain an accurate overall assessment, in which case I might well be better off asking an expert?

1.1.2 Preferences

Traditional Information Retrieval evaluation has focused on obtaining judgments from assessors as to whether or not specific sets of documents, text fragments, or other retrieved data are relevant to a given query. This procedure, while fairly effective for optimizing algorithms for all users, has the huge drawback that it assumes all assessors should prefer all documents similarly. This has clearly been shown to be false even when providing assessors ample time and a comprehensive narrative on document relevance, but especially when simply given a query.
Traditional methods of collecting relevance judgements make binary assumption about relevance i.e. a document is assumed to be either relevant or non-relevant to the information need of the user. This assumption turns relevance judgement into a classification problem. In the modern world, search engines can easily retrieve thousands of documents relevant to the user information need, so it becomes necessary to assign some ranking to these documents based on their relevance. The continuous nature of relevance cannot be expressed through binary relevance judgements, so judgements should be made considering multiple degrees of relevance. There can be two ways to express non-binary relevance judgements, either consider relevance as a relative notion such that one document is relevant in comparison to another document, or consider relevance as a quantitative notion and create multiple grades of relevance. The first notion of relevance can be expressed as preference judgements, and the second notion can be expressed as graded or nominal relevance judgements. Although graded relevance judgements are a natural extension of binary judgements and existing tools are more easily extended to them, they have a number of problems. Number of relevance grades and meaning of each relevance grade should be defined and it is not clear how these choices will affect relative performance measurements [8]. Moreover, there is high assessor disagreement for binary (nominal) judgments [9] as each assessor has his/her own internal threshold of relevance of a document.

Adopting graded relevance has two significant drawbacks. First, the specifics of the gradations (i.e. how many levels to use and what those levels mean) must be defined, and it is not clear how these choices will affect relative performance measurements.

To overcome weaknesses of absolute relevance judgements, researchers have been exploring utility of preference judgments. Use of preference judgements allows the assessor to make a binary decision and frees him/her from the difficulty of deciding between multiple relevance grades. But preferences also come with a challenge. Nominal judgments only require linear number of judgments to get a rank list of documents, whereas the number of possible preferences to compare each pair of \( n \) documents is \( O(n^2) \). A sophisticated mechanism is needed to get rank list of documents from linear number of preference pairs.

1.2 Research Questions

Crowd workers give inaccurate assessments while cost for expert assessments is very high. Our thesis is that instead of using one or the other (experts vs. crowd workers), we can get the best of both worlds (accurate and inexpensive) by com-
bining them, e.g., by asking the “right kind” of likely “easy” question to crowd workers (e.g., a simple preference pair) and only leaving the “hard” questions to the (expensive) experts. Our thesis can be summarized as “get low cost answers to easy questions, and only engage the expensive experts when their expertise is really necessary”. In order to achieve this, we have devised a framework for potentially combining mixed nominal and preference judgments from mixed crowd and expert workers; and we have devised a method for optimally asking the “right” question of the “right” worker at the “right” time. In order to achieve this, we need to model the accuracy and cost of various kinds of workers (for various kinds of questions). In short, we have addressed following research questions in this thesis:

1. How to identify patterns and biases in crowd worker’s mistakes in relevance assessments?
2. How to predict crowd worker’s quality of assessment?
3. How to select and combine noisy preference pairs to get a rank list of documents?
4. How to select right assessor (crowd worker or expert) for each question?
5. How to combine assessments from multiple assessors that have multiple levels of expertise?

1.3 Contribution and Results

1.3.1 Comparison of Preferences and Nominal Assessments

We have analysed the assessments made by crowd workers for two different types of relevance assessments: pair preferences and nominal relevance grades. We have found that it is easier to identity a relevant document when it is compared to a non-relevant document as compared to when it is presented on its own. We have concluded from this study that preferences are not always a better choice as compared to nominal assessments and one has to choose the type of relevance assessment very carefully based on the characteristics of the collection and nature of task.

1.3.2 Learning Contexts for Predicting Crowd Worker’s Assessment Quality

We have learned from analysis of assessments made by crowd workers that crowd workers have biases toward certain features of documents. From analysis of crowd-sourcing data we have learned certain contexts where crowd workers are more
likely to make mistakes. We have created contextualized confusion matrices for prediction of crowd worker quality for a particular question. Contextualized confusion matrices are useful for identifying which questions are easier or difficult for crowd workers and weather we should ask a crowd worker about a document with a particular context. For example, crowd workers are good at identifying non-relevant documents that are long and have low query terms, whereas they are not good at identifying relevant document that have low query terms and short length.

1.3.3 Combining Preferences to Get A Rank List

Traditional methods of collecting relevance judgements are based on collecting binary or graded nominal judgements. However, nominal relevance judgements are limited by factors such as inter-assessor disagreement and the arbitrariness of grades. For this reason, we explore the use and utility of preference judgements in crowdsourcing settings. Preferences are easier for assessors to produce and are more useful for training learning-to-rank algorithms. However, their use has been limited due to the polynomial increase in the number of judgments that need to be collected. We have shown how the Elo rating system [10] can be used to combine a linear number of preferences to obtain either an ordered list of documents or document relevance scores. The results of our experiments are encouraging and demonstrate the potential of our Elo-based system for inferring the relevance of documents from a linear number of pairwise preferences.

1.3.4 Framework for Optimally Combining Assessments from Crowd Workers and Experts

In an ideal scenario we should use expensive experts only for difficult questions where crowd workers are more likely to make mistakes and use low cost crowd workers for easier tasks. To facilitate such a process, we propose a Bayesian framework for combining potentially different types of judgments from mixed crowd and expert workers and for the online selection of the most cost-effective (question, assessor) pair at any given time. This requires modeling the quality, cost, and benefit of various types of workers for various types of questions. Our probabilistic framework combines assessments from crowd workers and experts, and optimally selects questions to be asked of each type of assessor.

1.4 Structure of Thesis

The chapters in this thesis do not present standalone work but build on each other. What we learned from one study, helped to initiate the next study. We started with exploration of preferences for IR evaluation. We participated in TREC 2012 Crowd-
sourcing Track and used preferences for collection of relevance assessments, the poor overall results for most track participants motivated us to analyse mistakes of crowd workers. This led to the work presented in Chapter 4. We collected both nominal and preference judgments and analysed the relevance assessments in order to identify patterns and bias in their mistakes. Chapter 5 presents our work related to selecting, collecting and combining preference judgments. The poor quality of crowdsourced assessments led us to believe that crowd workers alone cannot produce high quality relevance assessments and we need some mechanism for combining relevance assessments from assessors of varying quality. We explored various ways of combining assessments from assessors with different expertise levels like experts and crowd workers, and this exploration led to the creation of Bayesian framework for combining assessments from experts and crowd workers. Our Bayesian framework is presented in Chapter 6.

This thesis is organized as follows:

**Chapter 2** presents background on IR evaluation measures.

**Chapter 3** presents related work on Crowdsourcing, preference judgments and active learning in the context of IR evaluation.

**Chapter 4** provides analysis of mistakes, made by crowd workers, in relevance assessments (nominal and preferences).

**Chapter 5** addresses problem of efficiently combining preferences to get a rank list.

**Chapter 6** presents Bayesian framework for combining nominal assessments from experts and crowd workers.

**Chapter 7** concludes the presented research along with discussion on future work.
Chapter 2

Background on Information Retrieval Evaluation

This chapter provides necessary background on IR evaluation. Section 2.1.1 presents description of commonly used test collections in IR, Section 2.1.2 presents evaluation measures for evaluating quality of rankings, and Section 2.1.3 provides measures for assessing quality of judgments from non-experts.

2.1 Information Retrieval Evaluation

This section presents introduction to IR evaluation. To measure quality of a retrieval system, we need a test collection consisting of three things:

1. A document collection
2. A set of queries or information need
3. A set of relevance assessments (binary or multiple grades)

Traditional method of collecting relevance assessments uses binary classification of documents, a document is judged as either relevant or nonrelevant. Documents are judged with respect to a query or information need. Relevance of documents can be on multiple scales and judgements can be on multiple grades.

2.1.1 Test Collections

In the following we briefly introduce some of most well known test collections for IR evaluation which have been used in this thesis.
Text Retrieval Conference (TREC)

“The Text REtrieval Conference (TREC), co-sponsored by the National Institute of Standards and Technology (NIST) \(^1\) and U.S. Department of Defense, was started in 1992 as part of the TIPSTER Text program. Its purpose was to support research within the information retrieval community by providing the infrastructure necessary for large-scale evaluation of text retrieval methodologies [11]”. The most well known test collections are the ones used for the TREC Ad Hoc track during the first 8 TREC evaluations between 1992 and 1999. These test collections comprise 1.89 million documents (mainly newswire articles) and relevance judgments for 450 information needs, which are called topics and specified in detailed text passages. Individual test collections are defined over different subsets of this data. There are no exhaustive relevance judgments for these test collections since their size is so large. Rather, NIST (U.S. National Institute of Standards and Technology) assessors relevance judgments are available only for the documents that were among the top \(k\) returned for some system which was entered in the TREC evaluation for which the information need was developed.

The ClueWeb12 Dataset

The ClueWeb12 dataset [12, 13] was created to support research on information retrieval and related human language technologies. The dataset consists of 733,019,372 English web pages, collected between February 10, 2012 and May 10, 2012. ClueWeb12 is a companion or successor to the ClueWeb09 web dataset. Distribution of ClueWeb12 began in January 2013.

2.1.2 Evaluation of Rank Lists

In this section we introduce most commonly used measures for determining the effectiveness of retrieval systems. Precision \((P)\) is the fraction of retrieved documents that are relevant

\[
Precision = \frac{\text{relevant items retrieved}}{\text{retrieved items}} \quad (2.1)
\]

Recall \((R)\) is the fraction of relevant documents that are retrieved

\[
Recall = \frac{\text{relevant items retrieved}}{\text{relevant items}} \quad (2.2)
\]

These definitions can be made clear by examining the contingency Table 2.1.

---

\(^1\)www.nist.gov
<table>
<thead>
<tr>
<th>Relevant</th>
<th>Nonrelevant</th>
</tr>
</thead>
<tbody>
<tr>
<td>Retrieved</td>
<td>true positive (tp)</td>
</tr>
<tr>
<td>Not Retrieved</td>
<td>false negative (fn)</td>
</tr>
</tbody>
</table>

Table 2.1: Contingency Table [2]

Given the contingency Table 2.1, P and R can be defined as follows:

\[
P = \frac{(tp)}{(tp + fp)}
\]  \hspace{1cm} (2.3)

\[
R = \frac{(tp)}{(tp + fn)}
\]  \hspace{1cm} (2.4)

An information retrieval system can also be judged by its accuracy, that is, the fraction of its classifications that are correct. In terms of the contingency table above, \( accuracy = (tp + tn)/(tp + fp + fn + tn) \). This measure is often used for evaluating machine learning classification problems. There is a good reason why accuracy is not an appropriate measure for information retrieval problems. The IR data is extremely skewed: normally over 99.9% of the documents are in the nonrelevant category. A system tuned to maximize accuracy can appear to perform well by simply labelling all documents nonrelevant to all queries. Even if the system is quite good, trying to label some documents as relevant will almost always lead to a high rate of false positives. However, labeling all documents as nonrelevant is completely useless for user. The measures of precision and recall concentrate the evaluation on the return of true positives, asking what percentage of the relevant documents have been found and how many false positives have also been returned.

Precision and recall are set-based measures. They are computed using unordered sets of documents. For evaluation of raked retrieval results we need some other measure. Most standard measure among the IR community is Mean Average Precision (MAP), which provides a single-figure measure of quality across different recall levels. For a single topic, Average Precision (AP) is the average of the precision value obtained for the set of top \( k \) documents existing after each relevant document is retrieved, and this value is then averaged over topics. That is, if the set of relevant documents for a topic \( q_j \in Q \) is \( \{d_1, \ldots, d_{m_j}\} \), \( m_j \) is total number of relevant documents for topic \( j \), and \( R_{jk} \) is the set of ranked retrieval results from the top result until you get to document \( d_k \), then
\[ MAP(Q) = \frac{1}{|Q|} \sum_{j=1}^{|Q|} \frac{1}{m_j} \sum_{k=1}^{m_j} Precision(R_{jk}) \]  \hspace{1cm} (2.5)

When a relevant document is not retrieved at all, the precision value in the above equation is taken to be 0.

### 2.1.3 Evaluation of Relevance Assessments

Traditionally, relevance assessments are made by human experts. The high cost of producing human judgments, especially when evaluations must be performed quickly and frequently, has encouraged many researchers to explore other methods of obtaining relevance assessments (automatic or non expert human assessors). We need a measure of evaluation for determining quality of non-expert generated relevance assessments. Lapata [14] proposed the use of Kendall’s \( \tau \) a measure of rank correlation, as a means of estimating the distance between a non-expert-generated and a gold-standard order. Rank correlation is an appealing way of evaluating information ordering: Let \( Y = y_1...y_n \) be a set of items to be ranked. Let \( \sigma \) and \( \pi \) denote two distinct orderings of \( Y \), and \( S(\pi, \sigma) \) the minimum number of adjacent transpositions needed to bring \( \pi \) to \( \sigma \). Kendall’s \( \tau \) is defined as:

\[ \tau = \frac{2S(\pi, \sigma)}{N(N-1)/2} \]  \hspace{1cm} (2.6)

where \( N \) is the number of objects (i.e., items) being ranked. Kendall’s \( \tau \) is based on the number transpositions, that is, interchanges of consecutive elements, necessary to rearrange \( \pi \) into \( \sigma \).

Kendall’s \( \tau \) gives same weight to errors at top of rankings as compared to errors at bottom of rankings. Yilmaz et al. [15] have proposed \( \tau_{AP} \), as an alternative to Kendall’s \( \tau \), it gives more weight to the errors at high rankings. Let \( list1 \) and \( list2 \) be two lists of items of length \( N \) and suppose \( list2 \) is the actual ranking of items and \( list1 \) is a ranking of items whose correlation with the actual ranking (\( list2 \)) we would like to compute. Let \( C(i) \) be number of items above rank \( i \) and correctly ranked with respect to the item at rank \( i \) in \( list1 \), then \( \tau_{AP} \) is defined as follows:

\[ \tau_{AP} = \frac{2}{N-1} \sum_{i=2}^{N} \frac{C(i)}{i-1} - 1 \]  \hspace{1cm} (2.7)

We can use the two measures defined above for measuring quality of non-expert generated relevance assessments in the following way. Suppose we have a ranked lists of results from multiple search engines, we can use expert generated
relevance assessments (QREL) to evaluate the ranked lists using some evaluation measure for ranked results e.g. MAP. The resulting MAP values can be used to rank the search engines (suppose this rank list of search engines is called $\sigma$). Similarly we can use non-expert generated relevance assessments (QREL) to evaluate the ranked lists from search engines using MAP, these resulting MAP values can be used to create another ranking of the search engines (suppose this rank list of search engines is called $\pi$). Given two rankings $\sigma$ and $\pi$ we can use Kendall’s $\tau$ and $\tau_{AP}$ to measure the correlation between these two rankings. High correlation between these rankings would mean that non-expert generated relevance assessments have high quality and vice versa.
Chapter 3

Related Work

This chapter provides a brief review of past research and recent findings related to crowd sourcing, preference judgements, active learning in context of IR evaluation, and combining assessments from experts and crowd workers. Section 3.1 presents survey of resent papers on crowdsourcing for IR evaluation, techniques for ensuring quality in crowd sourced tasks, and impact of training on crowd worker’s performance. Section 3.2 discusses research done on combining assessments from expert assessors and crowd workers. Section 3.3 presents summary of preference judgments usage in domains other than IR, followed by brief discussion on recent work on preferences for IR evaluation. Section 3.4 sheds some light on active learning in general and active learning in context of crowdsourcing for IR evaluation and Section 3.5 summarizes the related work.

3.1 Crowd Sourcing

Jeff Howe defines crowdsourcing as “the act of taking a job traditionally performed by a designated agent (usually an employee) and outsourcing it to an undefined, generally large group of people in the form of an open call” [16]. Global market forces are increasingly moving computer work to regions of the world where it can be completed more quickly and cost-effectively. Lower cost of living in other geographic regions and the ability to decompose work into smaller units which can be efficiently distributed is giving rise to crowd sourcing. In the following a widely used tool for crowdsourcing (Amazon Mechanical Turk) is introduced, followed by a brief discussion on crowdsourcing in the context of IR evaluation.

Amazon Mechanical Turk

Amazon Mechanical Turk (AMT) is a crowdsourcing Internet marketplace that gives developers the ability to use human intelligence for tasks that computers are currently unable to do [17]. A person or organization,
termed the “requestor,” creates a task definition in the marketplace which workers
may then carry out in return for payment. A task definition is called a Human
Intelligence Task (HIT). When a worker submits work for a given HIT, the quality
of the submission can be automatically assessed and the submission can be either
approved, leading to payment, or rejected. There are around 200,000 registered
workers of AMT from almost 100 countries. More details on how to use this service
can be found in the developer documentation [17].

Crowdsourcing has been used in database community for building hybrid human-
machine database systems [18, 19]. Crowd workers are used for explaining null
values in returned results or for defining subjective operators in order to express
queries such as “I want pictures for motivational slides”. Crowdsourcing has also
been used by semantic web community. Demartini et al. [20] proposed a system
called ZenCrowd that identifies entities from natural language text automatically
connects them to the Linked Open Data cloud. They have used human intelligence
to improve the quality of the links by dynamically generating micro-tasks on an
online crowdsourcing platform. Seemakurty et al. [21] have explored a strategy for
using the linguistic abilities of human beings (through crowdsourcing) to develop
datasets that can be used to train machine learning algorithms for word sense dis-
ambiguation.

3.1.1 Crowd Sourcing for Information Retrieval Evaluation

Cranfield-based evaluation of IR systems [22] requires relevance assessments and
traditionally these relevance assessments are collected from human assessors which
is very time consuming and expensive. As a result there has been tremendous inter-
est in exploring more cost effective and scalable methods for collection of relevance
assessments. Sampling has been used extensively for retrieval system evaluations
using much fewer human judgments as compared to traditional pooling [23–28].
However, does sampling work for system evaluations in presence of noisy judg-
ments is an open question. Commercial search engines infer implicit judgments
from search logs but they still use human judges if they need highly accurate judg-
ments [29].

Crowdsourcing [30] has emerged as a viable platform for various relevance as-
essment tasks in recent years, but the benefits of crowdsourcing come with the risk
of unreliable workers. Recently, several scientific workshops have been dedicated
to exploring utility of crowdsourcing [31–37]. Some studies have shown utility of
crowdsourcing for natural language annotation tasks (word similarity, textual en-
tailment, and event temporal ordering) [38] but the usefulness of crowdsourcing

for intellectually complex tasks (relevance assessment of documents) has not been established yet. Crowdsourcing is also being used for book search and Kazai [39] has done a study of different parameters (worker quality, required effort and monetary reward) in context of crowdsourcing for book search. More work related to crowdsourcing for book search can be found in [40, 41].

Potential for crowdsourcing methods for cost effective relevance assessments and a replacement to traditional use of expert assessors has been studied extensively [42–48]. Bailey et al. [43] concluded from their study that non-expert judges are not reliable substitute for expert judges for relevance assessments tasks and there are significant differences in system evaluation scores using assessments from experts and non-experts. Alonso et al. [49] have found that the agreement on relevance assessments between crowd worker and the expert assessor is not high when measured individually, but it increases when crowd workers are grouped.

3.1.2 Quality Assurance Techniques

Crowdsourcing techniques have been investigated by evaluating crowd worker’s answers using assessments made by reliable assessors [50, 51], estimating the cost incurred due to redundant labels [42, 48, 52], aggregating inputs from multiple workers [38, 53], and investigating quality assurance techniques to remove unreliable workers [54–56]. There has been much research on how to fix crowd workers’ errors by inferring worker quality and then weight “good” and “bad” workers appropriately — generally with variations of the Expectation Maximization (EM) algorithm [47]. Eickhoff et al. [57] investigated various ways of making crowdsourcing HITs more robust against cheat submission. They concluded that cheaters are less frequently encountered in novel tasks that involve creativity and abstract thinking. They also observed that large batch sizes of HITs that offer a high reuse potential attracted more cheaters. Carterette and Soboroff [45] did a study on assessor errors in the context of crowdsourcing for large collections but they did not undertake a true crowdsourcing experiment. They presented eight models of possible errors and showed how each affects an estimate of average precision.

Joon et al. [58] presented a label inference method based on Probabilistic Matrix Factorization (PMF) by transforming crowdsourcing data into collaborative filtering data. The performance using unsupervised consensus labeling accuracy with PMF matched EM algorithm. Jung [59] has proposed a method for routing a crowdsourced task using matrix factorization to a mostly appropriate worker for quality assurance. Tang et al. [60] proposed a semi-supervised Naive Bayes approach for accurately inferring consensus labels given relatively less labeled training data for
estimating worker accuracy.

Alonso [61] has presented as industrial perceptive on crowdsourcing experiments, he stressed that designing and implementing experiments that require thousands or millions of labels is fundamentally different than conducting small scale experiments. Alonso emphasized different aspects of quality including clear instructions, a well-designed user interface, content quality, inter-rater agreement metrics, and worker feedback analysis. Panagiotis et al. [52] proposed an adversarial approach to prevent cheating, including an adaptive scheme for “trap questions” that prevents workers from exploiting easy-hit payable answers. Kittur et al. [62] has recommended that for avoiding spammers a task should be given in such a way that cheating takes approximately the same effort as faithfully completing the task. Zhu et al. [63] did a pilot study using Amazon’s Mechanical Turk to collect preference judgments between pairs of full-page layouts including both search results and image results. They have found that unreliable assessors may reveal their unreliability in different ways: some through periodic or fixed timings, and some through periodic or fixed ratings. They have also found that trap questions help identify unreliable workers. Kazai et al. [41, 64] have done a detailed analysis of human factors and label accuracy in crowdsourcing relevance judgments. They varied the level of pay offered, the effort required to complete a task and the qualifications required of the workers. They observed the effects of these variables on the quality of the resulting relevance labels, and measured based on agreement with a gold set. They have found that paying too little can lead to sloppy work, while paying too much can attract sophisticated spammers. For task design their recommendation is that it is essential to break-down tasks into easily-digestible units for the workers.

Khanna et al. [65] have done a study on HIT design in order to facilitate task understanding and worker efficiency in India. They have found that user interface is a bottleneck for low income users on Mechanical Turk, and simplified interfaces and task instructions can boost completion of task from 0% to 66%. Snow et al. [66] proposed modelling systematic worker bias that can be corrected later, they do not identify cheaters explicitly. Hsueh et al. [67] have used a combination of gold standard labels and majority voting to ensure result quality for their sentiment analysis of political blog posts. Soleymani and Larson [68] have identified high quality workers by offering a pilot HIT in first round of their study. They manually invite these high quality workers for actual task. Hirth et al. [69] presented a workflow in which several crowdsourcing steps are used in order to check on the quality of crowdsourced results. Gabriella Kazai [41, 70] has studied influence of the HIT.
setup on result quality, through pay rate, worker qualification or worker type.

3.1.3 Worker Training

Training can have significant impact on crowd worker’s quality of assessments. Le et al. [71] observed that for relevance assessment task uniform distribution of labels across training data labels produces best performance of workers. They also noted that as the training distribution more closely reflects the skewed distribution of relevant documents in IR collections, recall exceeded precision. A majority of workers labeled documents as non-relevant in order to increase recall at the cost of low precision. Wang et al. [72] did a study on e-discovery tasks and found that assessor training has little effect on reliability. Legally trained assessors had approximately same accuracy as untrained assessors on e-discovery tasks.

3.2 Combining Different Types of Assessors

The survey on crowdsourcing techniques presented in Section 3.1 suggests that crowdsourcing alone has not been able to achieve expert level assessments yet, and there is still ongoing research on improving the quality of results from crowd workers. Most of the research on crowdsourcing has used gold assessments from experts for detecting unreliable crowd workers. There has been little research on optimally combining assessments from experts and crowd workers. Eickhoff [73] did a study on benefits of combining annotations from crowd workers and experts for helping surgeons interpret Breast Cancer images. He has found that crowd workers cannot match the accuracy of experts even if amount of money spent on crowd workers (for collecting labels from multiple crowd workers for same instance) is increased to match cost of getting a label from expert. However, he also found that introduction of low level annotations from crowd workers as initial step increases the efficiency of expert labelling afterwards. These findings show that crowd workers cannot replace experts but they can be used in combination with experts for improving cost and efficiency of labelling. Khattak et al. [74] proposed a general scheme for injecting a few expert labels in a multiple crowd labeling setting in order to do quality-controlled labeling of large-scale data. One drawback of their scheme is that all instances are assigned to crowd workers for labelling which can increase cost.

Venanzi et al. [75] have proposed a community-based Bayesian label aggregation model, which assumes that crowd workers conform to a few different types, where each type represents a group of workers with similar confusion matrices. They have assumed that each worker belongs to a certain community, where the workers confusion matrix is similar to the community’s confusion matrix.
Difallah et al. [76] proposed a system carefully selects which workers should perform a given task based on worker profiles extracted from social networks. Workers and tasks are automatically matched using an underlying categorization structure that uses entities extracted from the task descriptions and categories liked by the user on social platforms.

These studies [75, 76] do not use expert assessors and only divide crowd workers into different categories based on certain worker characteristics.

The TREC 2012 Crowdsourcing track [77] targeted crowdsourcing on a highly specific, intellectually complex task. The goal was to reproduce the existing set of answers (QREL) obtained from highly trained experts over 10 queries randomly selected out of 50 from the TREC 1999 ad-hoc task. Participants were asked to use various crowdsourcing techniques to produce high quality results. Some of the participants used a mixture of automated methods and crowd workers for relevance assessments [78, 79]. Only one participant group [79] was able to achieve high quality assessments, but they did not use crowd workers. Instead, they used an ad-hoc mixture of automated methods, expert assessors, and graduate students whose quality is somewhere between experts and crowd workers. This group was able to leverage some of the benefits of combining different types of assessors but the cost of their assessments was not optimal and could be reduced if they had included crowd workers in their framework. We have built a framework (presented in Chapter 6) that can mix different types of assessors in an optimized fashion to achieve high quality assessments at low cost.

3.3 Preferences

Relevance is widely recognized for being multi-faceted and subjective. Human judgments are influenced by various situational, cognitive, perceptual and motivational biases [80, 81]. Document properties, judgment conditions and scales, and personal factors also affect human judgments [82]. Relevance judgments for information retrieval evaluation have traditionally been made on a graded scale but high inter-assessor disagreement, difficulty in definition and specificity of relevance grades has motivated researchers to explore preference based relevance judgments.

A preference can be defined as an ordering relation between two or more items to characterize which, among a set of possible choices, is the one that best fits user tastes [83].

There is a small amount of work on using preferences for collaborative filtering, novelty and diversity, and music similarity. Urbano et al. [84] used preferences
for getting ordered lists of music pieces according to their similarity to a particu-
lar music piece. They have used quicksort to get an ordered list of music pieces
by collecting judgements for $O(n \log n)$ pairs using crowd sourcing. Their ordered
lists were not large so it was not expensive to collect judgements for $O(n \log n)$
pairs. For creation of text test collections, we need judgements for a large number
of documents so it is very expensive to collect judgements for $O(n \log n)$ pairs of
documents. Brun et al. [85] studied the expression of preferences in collaborative
filtering under the form of preference relations instead of ratings. They have shown
that the use of preferences instead of ratings sometimes improves the performance
of recommendation systems. Chandar and Carterette [86] used user preferences to
test their hypotheses about novelty and diversity. They have done some experi-
ments using preferences to find out if users prefer to see novel information (same
subtopic) over diverse information (new subtopics).

### 3.3.1 Preferences for Information Retrieval Evaluation

Researchers have been exploring utility of preferences for IR evaluation since past
two years but sheer number of all possible preference pairs ($\binom{n}{2}$) makes the task dif-
ficult. There have been a few attempts to use preference judgements by inferring
them from absolute judgements [87, 88]. Joachims used click data to infer prefer-
ences [29]. He hypothesized that a document being clicked is preferred to all doc-
uments ranked above it. Chandar et al. [89] proposed graph algorithm (PageRank)
to minimize the number of judgments required to evaluate systems using pref-
erence judgments. They compared their results to a baseline method that used
majority vote for combining preference votes. They showed that their method out-
performed baseline when they randomly selected smaller sample of pairs from
space of all $\binom{n}{2}$ possible pairs. Carterette and Petkova [90] have used preferences
for combining results from different retrieval systems. They convert rankings pro-
duced by each system to preferences and use logistic regression model to learn a
ranking from pairwise document preferences. Radlinski and Joachims [91] have
used active learning for selection of preferences instead of collecting preferences
passively or naively. They did not collect preferences from users directly, prefer-
ences were inferred from clickthrough data. They were able to show improvement
over baseline (random selection of preference pairs) by using a loss function for
selection preference pairs. Each preference pair had an associated loss and their
algorithm selected the pair that had highest expected reduction in overall loss of
the ranking.

Thomas and Hawking [92] proposed a preference method that displays two
sets of search results side-by-side and asks users to indicate which side they preferred. They inferred preferences from side-by-side presentations of web results using clickthrough data and were able to detect a difference in user preferences between a high-quality set of results and a lower-quality set. Sanderson et al. [93] provided compelling evidence for the utility and reliability of user preferences. They collected actual user preferences for results from different system runs using crowdsourcing, and found high correlation between diversity based evaluation measures and user preferences. Kim et al. [94] investigated the role of various relevance dimensions in preference based evaluation. They proposed a preference based evaluation method to collect both overall and per relevance dimension based preferences (Relevance, Diversity, Authority, Freshness and Caption quality). They have found that overall preferences capture a range of dimensions, most dominantly Relevance and Diversity, and preference judgments combine multiple dimensions of relevance that go beyond the traditional notion of relevance that is centered on topicality. Radlinski et al. [95] compared preference based evaluation to absolute metrics based evaluation for prediction of quality of retrieval systems. They have found that absolute metrics did not give reliable results for the sample size collected in their study. In contrast, two preference based evaluation algorithms, namely Balanced Interleaving as well as the new Team-Draft Interleaving method they proposed, gave consistent and mostly significant results.

To the best of our knowledge, the most notable work done on the collection of direct preferences for relevance assessment of documents is by Carterette et al. [8]. They have done an assessor study to find out if it is easier to make preference judgements as compared to absolute judgements by measuring the inter-assessor consistency and the time spent on each type of judgement. Results of their study show that there is a higher inter-assessor agreement on preference judgements, and it takes less time to make a preference judgement as compared to an absolute judgement. They used 5 expert judges in their study and collected preferences for 5 documents per topic. Since they used a small size of data so it was easier to collect preferences for all possible pairs of documents, and use experts for collecting preference judgements for those pairs. It is very expensive to use their methodology for collecting relevance assessment for large test collections. The main difference between our study and their study [8] is that we have investigated the utility of preferences for obtaining better relevance judgements for large IR test collections, as opposed to using preferences for ranking small number of top documents. Another difference is that we have evaluated utility of crowd sourcing for relevance judgements as opposed to using expert assessors.
3.4 Active Learning

Active learning is the process of intelligently choosing unlabelled instances to be labeled by an oracle to achieve higher accuracy with as few training labels as possible, instead of simply labeling all the data or randomly selecting data to be labeled. Active learning is a growing area of research in Information Retrieval and Machine Learning, no doubt fueled by the reality that data is increasingly easy or inexpensive to obtain but difficult or costly to label for evaluation. Radlinski and Joachims [91] used active learning for selection of preferences pairs but they did not collect actual preference data. They inferred preferences from clickthrough data. Rehbein et al. [96] investigate active learning with human expert annotators for word sense disambiguation, but do not find convincing evidence that active learning reduces annotation cost in a realistic (non-simulated) annotation scenario. While active learning generally is superior to baselines without active learning in simulated experiments, it is not clear that this result carries over to crowdsourcing. Crowdsourcing differs in a number of ways from simulated experiments; first, the difficulty and labeling consistency of examples drawn by active learning differs from that drawn by random sampling; second, crowdsourcing labels are noisy. These studies only evaluate their approach in simulations. We have used the actual labels of human assessors to avoid the risk of unrealistic assumptions when modeling crowdsourced workers, our crowdsourcing experiments are presented in Chapter 4.

3.4.1 Crowd sourcing Assessments using Active Learning

Laws et al. [97] evaluated the utility of active learning in crowdsourcing named entity recognition and sentiment detection and found that active learning outperforms baselines without active learning. There have been many studies on active learning for machine learning without crowdsourced data [96, 98, 99]. Brew et al. [98] used a small set of volunteer labelers instead of anonymous paid workers. Chen and Bennett [100] used active learning for application of crowdsourcing to the task of assessing documents by their reading difficulty. They have generalized the widely applied Bradley-Terry model [101] by incorporating annotator quality. They have used pairwise comparisons and proposed an active learning strategy that can adaptively sample the next assessment pair and annotator. Yan et al. [102] proposed a probabilistic multi-labeler model that allows for learning from multiple annotators, whose expertise across the data space may vary. Their model selects the object to be labelled and the annotator according to annotator quality. They did not test their model for IR test collections. Nallapati et al. [103] have proposed active learning algorithm called CorrActive learning. It addresses the problem where the
set of training data provided to the supervised learner is noisy with respect to its labels. Their algorithm corrects labeling errors in data iteratively by presenting only those examples to users that are most likely to be mislabeled and it simultaneously learns from corrected examples. They have shown improvement using their algorithm as compared to learner that chooses instances for labeling through random sampling (using synthetic experiments).

3.5 Summary

This chapter presents a literature survey on crowdsourcing, preference judgments, and active learning in general as well as in the context of IR evaluation. There is large body of work on crowdsourcing for IR evaluation and on quality assurance measures for crowd sourced tasks but we have found small amount of research on combining labels from crowd workers and experts in IR evaluation. Similarly active learning has been used extensively in past research but there is not enough research and prior work on active learning for combining noisy labels and expert labels for IR evaluation. This literature survey has motivated us to build framework for combining assessments from assessors of variable quality in active learning fashion (presented in Chapter 6). Problems in traditional graded relevance assessments has motivated the need to explore preferences for IR evaluation. We can conclude from this literature survey that “can preferences be used alone or in combination with graded judgments for IR evaluation?” is still an open research question for IR evaluation. We have addressed this question in Chapter 5 and applied a classical rating scheme called Elo rating [10] to the problem of combining preference pairs for IR evaluation.
Chapter 4

Analysis and Comparison of Crowd sourced Relevance Assessments (Nominal vs Pairs)

Information Retrieval strategies require ground truth assessments on the produced results as a basis for optimization, training, and evaluation. Traditionally, the ground truth was obtained through very expensive expert humans looking through the pool of top retrieved documents and judging relevance to the respective queries. Since the cost of this manual effort is prohibitive in many ways, and is definitely not scalable with web search, many researchers are using alternative assessments from crowdsourcing services, like Amazon Mechanical Turk (AMT). This effort is a lot cheaper, but less reliable. It is also far more suitable to trivial yes/no questions like "Does this picture contain a tiger?" than to intellectually complex questions like "What motivates racial profiling cases in the U.S.?". Since crowdsourcing is quickly becoming very popular due to its cheapness, clearly research is needed to understand crowd worker’s behavior on non-trivial questions and in particular to understand the nature of the errors they make. We focus on the nature of mistakes, and ask why crowdsourcing of particularly complex tasks has not yet succeeded in getting a reasonable approximation of the expert assessments. This chapter presents analysis of crowd worker’s mistakes for two assessment types (nominal and preferences). In the following we briefly discuss goals of TREC Crowdsourcing track, followed by experimental setup in Section 4.1, analysis in Section 4.2, and summary of this chapter in Section 4.3.

**TREC Crowdsourcing Track**  The TREC 2012 CrowdSourcing track [77] targeted crowdsourcing on a highly specific, intellectually complex task. The goal was to
reproduce the existing set of answers (QREL) obtained from highly trained experts over 10 queries randomly selected out 50 from the TREC 1999 ad-hoc task. Assessments submitted to the TREC track which disagreed with the QREL were analyzed and eventually regarded as crowdsourcing errors rather than errors in the QREL. As a result of the complexity and specificity of the track queries, most crowdsourcing approaches submitted to the 2012 TREC track failed to achieve a performance good enough to be considered viable substitutes for the expert QREL (Query RELEVANCE). Therefore, we decided to analyze crowdsourcing mistakes made on this task using data we collected via Amazon’s Mechanical Turk service. We investigate two types of crowdsourcing approaches: one that asks for nominal relevance grades for each document, and the other that asks for preferences on many (not all) pairs of documents. Looking at the crowdsourced errors, for various queries and questions asked, we analyze workers behavior and mistakes with respect to:

- **time spent** on the task per document - are mistakes more likely from workers which spend less time?
- **document length** - are mistakes more likely on longer documents?
- **query term presence** - are mistakes likely on documents misleading by presence/absence of query terms?
- **query term presence in title** - are mistakes likely on documents misleading by presence/absence of query terms in title of documents?

**Author’s Contributions**  The study presented in this chapter has already been published [104], following are author’s contributions that helped to shape the direction of the work presented in this chapter.

- Analyzed crowdsourcing mistakes in crowdsourced data and observed biases in these mistakes
- Read the documents with most egregious mistakes in relevance assessments made by crowd workers.
- Identified document and assessor features that cause the bias in crowd workers relevance assessments
- Created plots to verify biases in relevance assessments occur across topics
4.1 Experiments

In this section we have described our experiments for collecting, processing and evaluating relevance judgements using two different approaches i.e. preferences and graded judgements. Following sections give details about each phase for obtaining relevance judgements.

4.1.1 Data

We ran our experiments on the 10 queries proposed by the TREC 2012 crowdsourcing track [35]. Following This track used documents from TREC ad-hoc 1999 task. The TREC ad-hoc 1999 task is one of the most complete sets of query-documents ever judged, with an average of about 1,800 documents assessed per query. The official list of documents contains about 18,200 documents, or about 1,820 per query. Out of these the 1999 TREC ad-hoc QREL marks about 650 as relevant by the experts, or about 65 per query. For the purpose of our analysis, “relevant” or R refers to expert-QREL marked relevant, and “non-relevant” or NR refers to the rest of the documents, i.e. the ones expert-QREL marked nonrelevant.

Topics

Following is a list of 10 topics from TREC 2012 crowdsourcing track used in these experiments. Each topic has a description and a narrative.

- Topic 411: Salvaging, shipwreck, treasure
  - Description: Find information on shipwreck salvaging: the recovery or attempted recovery of treasure from sunken ships.
  - Narrative: A relevant document will provide information on the actual locating and recovery of treasure; on the technology which makes possible the discovery, location and investigation of wreckages which contain or are suspected of containing treasure; or on the disposition of the recovered treasure.

- Topic 416: Three Gorges Project
  - Description: What is the status of The Three Gorges Project?
  - Narrative: A relevant document will provide the projected date of completion of the project, its estimated total cost, or the estimated electrical output of the finished project. Discussions of the social, political, or ecological impact of the project are not relevant.
• Topic 417: Creativity
  - Description: Find ways of measuring creativity.
  - Narrative: Relevant items include definitions of creativity, descriptions of characteristics associated with creativity, and factors linked to creativity.

• Topic 420: Carbon monoxide poisoning
  - Description: How widespread is carbon monoxide poisoning on a global scale?
  - Narrative: Relevant documents will contain data on what carbon monoxide poisoning is, symptoms, causes, and/or prevention. Advertisements for carbon monoxide protection products or services are not relevant. Discussions of auto emissions and air pollution are not relevant even though they can contain carbon monoxide.

• Topic 427: UV damage, eyes
  - Description: Find documents that discuss the damage ultraviolet (UV) light from the sun can do to eyes.
  - Narrative: A relevant document will discuss diseases that result from exposure of the eyes to UV light, treatments for the damage, and/or education programs that help prevent damage. Documents discussing treatment methods for cataracts and ocular melanoma are relevant even when a special cause is not mentioned. However, documents that discuss radiation damage from nuclear sources or lasers are not relevant.

• Topic 432: Profiling, motorists, police
  - Description: Do police departments use profiling to stop motorists?
  - Narrative: A relevant document will report or discuss police department criteria for identifying motorists considered likely to be carrying contraband. Documents discussing the detention of individuals by foreign security forces are not relevant.

• Topic 438: Tourism, increase
  - Description: What countries are experiencing an increase in tourism?
- Narrative: A relevant document will name a country that has experienced an increase in tourism. The increase must represent the nation as a whole and tourism in general, not be restricted to only certain regions of the country or to some specific type of tourism (e.g., adventure travel). Documents discussing only projected increases are not relevant.

- Topic 445: Women clergy
  - Description: What other countries besides the United States are considering or have approved women as clergy persons?
  - Narrative: To be relevant, a document must indicate either a country where a woman has been installed as clergy or a country that is considering such an installation. The clergy position must be as church pastor rather than some other church capacity (e.g., nun or choir member).

- Topic 446: Tourists, violence
  - Description: Where are tourists likely to be subjected to acts of violence causing bodily harm or death?
  - Narrative: A relevant document must contain accounts of known harm to tourists. Evidence of single, isolated incidents are not relevant.

- Topic 447: Stirling engine
  - Description: What new developments and applications are there for the Stirling engine?
  - Narrative: Any discussion of new developments and applications of the Stirling engine (also known as the Stirling cycle) are relevant.

4.1.2 HIT Design
The HITs we created to collect preference judgements from crowd workers had the following design. After accepting the assignment, workers were shown the instructions. In these instructions, we explained that documents should be judged strictly based on whether they provide information about the query, description, and narrative for a particular topic: that a well-written discussion of a related topic should not be preferred to a poorly-written document which is exactly on topic. The documents in the corpus are unformatted. We presented the document title and byline (when present) in bold and separated the paragraphs, but we did not
add any additional formatting. In particular we did not highlight query terms within the documents, so any user bias toward query terms in our findings could not have resulted from any extra emphasis our interface gave them.

Preferences

After dismissing the instructions, the worker is shown the interface presented in Figure A.1. The “Title” field of a TREC topic is displayed on top, along with its description and narrative. This information describes in detail what constitutes a relevant document for this query. Below the query information is a series of buttons, which allow workers to record their preferences. Two documents are displayed side-by-side below the buttons. The leftmost and rightmost buttons are labeled “This One,” with an arrow pointing to the left or right document, respectively. These buttons allow users to choose a winning document. Between these buttons are two buttons for recording ties, labeled “They’re Equally Good” and “They’re Equally Bad.” Each HIT consisted of 20 preference pairs for the same topic, and had a time limit of 30 minutes. Workers were paid $0.15 for each approved HIT. The order in which the preferences were displayed, as well as the order of the documents for each particular preference, was randomized.

Document pairs were selected for judgement in the following manner. First, we calculated a prior relevance score for each document using BM25. This produced an initial ranking of the documents for each topic. For each document below the top six, we selected five documents to be compared against, uniformly at random, from the set of documents with higher BM25 scores. For the top six documents, we collected complete pairwise preferences.

Nominal Judgments

The interface for graded judgements is shown in Figure 4.2. This interface shows a simple one-to-five star voting scheme with a textual description of the meaning of each star appearing both in a tool-tip and in text to the right of the stars when the mouse cursor is placed over a particular star. The users are told that the descriptions will appear when the mouse is placed over each star. Following descriptions are shown when mouse is placed over a particular star.

1. Completely non-relevant and not related to topic
2. Not relevant but somewhat related
3. Somewhat relevant

\[\text{BM25 is a probabilistic retrieval model [105]}\]
4. Relevant

5. Highly relevant

Each HIT for graded judgements consisted of 40 documents for the same topic, and it had a time limit of 30 minutes. Workers were paid $0.15 for each approved HIT for graded judgements.
4.1.3 Quality Control

The workers we employed had no particular training in assessing document relevance, so we needed a means of verifying the quality of their work. There have been many studies on assessing worker quality for crowd sourcing platforms such as AMT [47]. We used trap questions in our study to ensure that workers are making a reasonable effort, instead of clicking randomly.

A trap question is a question inserted into the HIT which looks the same as any other, but for which a correct answer is known ahead of time. We asked five IR graduate students to create our trap questions by pairing documents which appeared to be highly relevant with documents which appeared to be highly non-relevant. We then mixed five of these trap questions, selected at random, into each HIT. As a result, each assignment consisted of five trap questions and fifteen “real” questions. A worker’s submission was accepted if at least two of the five trap questions were answered correctly, and rejected otherwise. For creation of trap questions for graded judgements, we converted the preference judgements for trap questions into graded judgements. If a worker gave higher relevance grade to the preferred document as compared to the other document of preference pair then his/her answer was recorded as correct judgement.

We have also used the well known technique of obtaining multiple labels from different workers for reducing noise through majority rule [52]. For graded relevance judgements, we collected 5 assessments for each document for a total of about 90,000 judgments and used 5 trap questions in batch of 20 documents. For the pair preferences, we collected 5 assessments for each pair and used 5 trap questions in batch of 20 pairs.

Table 4.1 and Table 4.2 show distribution of assessments per worker for each topic. These tables also show average time spent per assessment by crowd workers on nominal judgments and preferences. Overall crowd workers have spent more time on making one nominal assessment as compared to one preference judgement. This shows that workers have to spend less time to decide a preference as compared to deciding between multiple grades.

4.2 Analysis

This section discusses the various factors which seem to lead workers to select non-relevant documents over relevant ones. We discuss how the difference in the time a worker is willing to spend affects judgement quality, as well as several properties of the documents under comparison. Each of these factors varies from topic to topic. We suspect that this rises from either an inherent difference in the difficulty
<table>
<thead>
<tr>
<th>Topic Id</th>
<th>Number of Workers</th>
<th>Average Number of Assessments Per Worker</th>
<th>Average of % of Total Assessments Per Worker</th>
<th>Average Time Spent Per Assessment (Seconds)</th>
</tr>
</thead>
<tbody>
<tr>
<td>427</td>
<td>232</td>
<td>155</td>
<td>0.43</td>
<td>27</td>
</tr>
<tr>
<td>445</td>
<td>147</td>
<td>205</td>
<td>0.68</td>
<td>26</td>
</tr>
<tr>
<td>417</td>
<td>210</td>
<td>301</td>
<td>0.48</td>
<td>28</td>
</tr>
<tr>
<td>446</td>
<td>243</td>
<td>189</td>
<td>0.41</td>
<td>29</td>
</tr>
<tr>
<td>416</td>
<td>204</td>
<td>150</td>
<td>0.49</td>
<td>23</td>
</tr>
<tr>
<td>447</td>
<td>168</td>
<td>216</td>
<td>0.6</td>
<td>24</td>
</tr>
<tr>
<td>432</td>
<td>195</td>
<td>274</td>
<td>0.51</td>
<td>22</td>
</tr>
<tr>
<td>420</td>
<td>191</td>
<td>149</td>
<td>0.52</td>
<td>26</td>
</tr>
<tr>
<td>411</td>
<td>203</td>
<td>221</td>
<td>0.49</td>
<td>28</td>
</tr>
<tr>
<td>438</td>
<td>252</td>
<td>174</td>
<td>0.4</td>
<td>31</td>
</tr>
</tbody>
</table>

Table 4.1: Distribution of assessments among crowd workers for preference pairs

<table>
<thead>
<tr>
<th>Topic Id</th>
<th>Number of Workers</th>
<th>Average Number of Assessments Per Worker</th>
<th>Average of % of Total Assessments Per Worker</th>
<th>Average Time Spent Per Assessment (Seconds)</th>
</tr>
</thead>
<tbody>
<tr>
<td>427</td>
<td>73</td>
<td>117</td>
<td>1.37</td>
<td>38</td>
</tr>
<tr>
<td>417</td>
<td>88</td>
<td>135</td>
<td>1.14</td>
<td>38</td>
</tr>
<tr>
<td>445</td>
<td>70</td>
<td>109</td>
<td>1.43</td>
<td>34</td>
</tr>
<tr>
<td>446</td>
<td>104</td>
<td>113</td>
<td>0.96</td>
<td>44</td>
</tr>
<tr>
<td>416</td>
<td>119</td>
<td>64</td>
<td>0.84</td>
<td>27</td>
</tr>
<tr>
<td>447</td>
<td>40</td>
<td>208</td>
<td>2.5</td>
<td>40</td>
</tr>
<tr>
<td>432</td>
<td>72</td>
<td>130</td>
<td>1.39</td>
<td>35</td>
</tr>
<tr>
<td>420</td>
<td>60</td>
<td>107</td>
<td>1.67</td>
<td>42</td>
</tr>
<tr>
<td>411</td>
<td>87</td>
<td>130</td>
<td>1.15</td>
<td>35</td>
</tr>
<tr>
<td>438</td>
<td>77</td>
<td>130</td>
<td>1.3</td>
<td>36</td>
</tr>
</tbody>
</table>

Table 4.2: Distribution of assessments among crowd workers for nominal judgments
of topics or a difference in the difficulty of the pairs we selected for judgement.

Most of our charts present some measurement versus an error rate, per topic. These error rates are the observed probability of a randomly-selected worker making an error of a certain type. We consider a relevance grade to be correct if a worker assigns a grade of 3 or 4 to a relevant document or a grade of 0, 1, or 2 to a non-relevant document, and a pair preference to be correct if a worker prefers a relevant document over a non-relevant document. For consistency reasons, we exclude preferences between two relevant or two non-relevant documents. X-axis on all plots represents TREC specific topic numbers. We use box and whisker plots in order to present the distribution over workers. The line in the center of each box is its median value; each box covers the inner 50% of the data, and the whiskers extend to the highest and lowest observed values which lie within 1.5 times the range covered by the box. We ran Welch two sample t-tests (one sided) for the hypothesis that the mean of one distribution is greater than the mean of the other, and mark with asterisks (*) those topics which have statistically significant findings ($p < 0.05$).

4.2.1 Quality of Assessments

We present the recall and precision of our workers for preference judgments in Table 4.3. Calculating these values for preference pairs is not straightforward as defined in equations 4.1 and 4.2.

\[
\text{Precision} = \frac{|J(d_i, d_j) \in \{>\}; d_i \in R, d_j \notin R|}{|J(d_i, d_j) \in \{>\}|} \quad (4.1)
\]

\[
\text{Recall} = \frac{|J(d_i, d_j) \in \{>\}; d_i \in R, d_j \notin R|}{|J(d_i, d_j) \in \{>, <, =\}; d_i \in R, d_j \notin R|} \quad (4.2)
\]

where $J(d_i, d_j)$ represents the crowdworker preference between document $d_i$ and document $d_j$.

Recall is calculated as the ratio of the number of correct judgements between a relevant and a non-relevant document to the total number of judgements between a relevant and a non-relevant document, and precision as the ratio of the number of correct judgements between a relevant and a non-relevant document to the number of judgements which were not ties. Note that we count individual worker judgements here; we do not attempt to aggregate judgements for the same document. We chose to count in this manner in order to produce numbers which best reflected the decisions of individual workers.

For most topics, recall was much higher than precision. That is, workers seem
Table 4.3: Recall and precision of preference judgments

<table>
<thead>
<tr>
<th>Topic Id</th>
<th>Recall</th>
<th>Precision</th>
</tr>
</thead>
<tbody>
<tr>
<td>427</td>
<td>0.75</td>
<td>0.35</td>
</tr>
<tr>
<td>445</td>
<td>0.70</td>
<td>0.26</td>
</tr>
<tr>
<td>417</td>
<td>0.74</td>
<td>0.2</td>
</tr>
<tr>
<td>446</td>
<td>0.60</td>
<td>0.32</td>
</tr>
<tr>
<td>416</td>
<td>0.72</td>
<td>0.26</td>
</tr>
<tr>
<td>447</td>
<td>0.75</td>
<td>0.23</td>
</tr>
<tr>
<td>432</td>
<td>0.56</td>
<td>0.03</td>
</tr>
<tr>
<td>420</td>
<td>0.66</td>
<td>0.15</td>
</tr>
<tr>
<td>411</td>
<td>0.65</td>
<td>0.09</td>
</tr>
<tr>
<td>438</td>
<td>0.65</td>
<td>0.39</td>
</tr>
<tr>
<td>Average</td>
<td>0.68</td>
<td>0.23</td>
</tr>
</tbody>
</table>

to mislabel a lot of documents as being relevant. While the precision results may seem low overall for pair preferences, they are above the median results submitted at TREC 2012 Crowdsourcing track; more reason for this analysis.

Table 4.4 shows sensitivity and specificity of assessments for nominal judgments. Sensitivity and specificity are defined as follows:

\[
Sensitivity = \frac{\text{number of true positives}}{\text{number of true positives + number of false negatives}} \tag{4.3}
\]

\[
Specificity = \frac{\text{number of true negatives}}{\text{number of true negatives + number of false positives}} \tag{4.4}
\]

Sensitivity is equivalent to probability of relevant label (label 3 or 4) given the document is relevant whereas specificity is equivalent to probability of non-relevant label (label 0,1 or 2) given the document is non-relevant. Specificity for all topics is much higher as compared to sensitivity which shows that crowd workers have a hard time to identify relevant documents but they can easily identify non-relevant documents.

### 4.2.2 Worker Quality vs. Time Spent

We calculated both the normalized number of correct answers (total number of correct answers divided by total number of questions attempted by each worker) each worker gave and the average time the worker spent producing each answer. We then divided the workers into three groups for each topic: workers whose average
<table>
<thead>
<tr>
<th>Topic Id</th>
<th>Sensitivity</th>
<th>Specificity</th>
</tr>
</thead>
<tbody>
<tr>
<td>420</td>
<td>0.28</td>
<td>0.9</td>
</tr>
<tr>
<td>417</td>
<td>0.43</td>
<td>0.9</td>
</tr>
<tr>
<td>432</td>
<td>0.28</td>
<td>0.88</td>
</tr>
<tr>
<td>447</td>
<td>0.36</td>
<td>0.96</td>
</tr>
<tr>
<td>416</td>
<td>0.43</td>
<td>0.7</td>
</tr>
<tr>
<td>438</td>
<td>0.41</td>
<td>0.84</td>
</tr>
<tr>
<td>446</td>
<td>0.</td>
<td>0.86</td>
</tr>
<tr>
<td>411</td>
<td>0.46</td>
<td>0.89</td>
</tr>
<tr>
<td>427</td>
<td>0.35</td>
<td>0.93</td>
</tr>
<tr>
<td>445</td>
<td>0.46</td>
<td>0.9</td>
</tr>
<tr>
<td>Average</td>
<td>0.4</td>
<td>0.88</td>
</tr>
</tbody>
</table>

Table 4.4: Sensitivity and specificity of assessments for nominal judgments

Figure 4.3: Quality of workers versus time spent for pair preferences. The y axis shows the distribution over average fraction of correct answers per worker. Workers are split into three groups based on their average durations. T-tests were run between the fastest and slowest groups; asterisks indicate low p-value ($p < 0.05$).
Worker Quality

\[ T_{\text{ime}} < \mu - \sigma \]
\[ T_{\text{ime}} \in \mu \pm \sigma \]
\[ T_{\text{ime}} > \mu + \sigma \]

Relevance Grades: Time vs. Quality

Figure 4.4: Quality of workers versus time spent for relevance grades. The y axis shows the distribution over average fraction of correct answers per worker. Workers are split into three groups based on their average durations. T-tests were run between the fastest and slowest groups; asterisks indicate low p-value (\( p < 0.05 \)).

duration lies within one standard deviation of the mean, workers whose average duration was more than a standard deviation below the mean, and workers whose average duration was more than a standard deviation above. We found that for most topics, for both relevance grades and pair preferences, workers who spent more time produced more accurate responses. The best workers spent more time on average per assessment on both relevance grades and pair preferences, but the effect is smaller for pair preferences. Our results are presented in Figure 4.3 and Figure 4.4.

Yes the worker quality is nearly random for TREC 8 data. TREC 8 documents and topics were complex and difficult for crowd workers so their quality for this collection is nearly random. Some topics had restrictive narrative, for example the topic 432 “profiling motorists police”, documents are relevant only if they discuss police department criteria for identifying motorists considered likely to be carrying contraband. Documents discussing the detention of individuals by foreign security forces are not relevant. Some crowd workers may consider all documents including keywords of “police”, “motorists”, and “profiling” relevant. The topic 417 “Creativity” has a very abstract definition and includes definitions of creativity, descriptions of characteristics associated with creativity, and factors linked to
creativity. There is no concrete criteria for considering a document to be relevant for this topic so crowd workers had to use much effort reading and understanding the documents for this topic. Most crowd workers are not motivated enough to spend this amount of effort so their judgments are often wrong.

Table 4.5 and Table 4.5 show averages of time spent per assessment by crowd workers. Both tables show that as workers spend more time per assessment when they make fewer assessments. Workers who have made most assessments have spent least time per assessment. The reason might be that as workers make more assessments they get more familiar with the task and the topic so they need less time to make an assessment.

### 4.2.3 Document Length vs. Error

In the process of diagnosing the nature of the problem, we observed that many of the non-relevant documents which were preferred over relevant documents were much longer than the relevant documents. Perhaps workers tend to feel that a
longer document is more authoritative. In Figure 4.5 and Figure 4.6, we compare
the error rate of workers for comparisons when the relevant document is longer to
the error rate when the non-relevant document is longer. As a baseline, we compare
this data to the error rate of relevance grades for documents whose lengths are
above or below the mean document length for each topic. For relevance grades, we
find that in most topics worker assessments are more reliable for longer documents.

This trend continues for preference data. The success rates for pairs in which
the relevant document was longer (the red bars) are almost all higher than the
success rates when the non-relevant document was longer (the blue bars). This
suggests to us that many workers will give a preference for a longer document over
a shorter, perhaps without carefully considering the relative information content of
those documents.

**4.2.4 TF-IDF vs. Error**

In Figure 4.7 and Figure 4.8 we compare Tf-IDF score with error. We observed
several cases where workers preferred non-relevant documents containing many
query terms over relevant documents, so we wanted to measure how often that is
the case, by calculating the TF-IDF score for each document and measuring that
against assessment quality.

For our baseline, we plot the success rates for workers assigning relevance
Figure 4.6: Document Length vs. Error rate for pair preferences. Error increased when on pairs of a non-relevant document VS a shorter relevant document.

grades to documents whose TF-IDF scores are either above or below the mean score for that topic. We find no consistent trend across topics: for many topics, the success rate is similar on both sides of the mean. On some topics, however, there is a statistically significant increase in mistakes for documents with small TF-IDF scores. For pair preferences, on the other hand, there is a clear trend: when the relevant document has a higher TF-IDF score than the non-relevant document, the assessments are more reliable.

4.2.5 Document Title Relevance vs. Error

For our final plot, we speculated that the presence of query terms in the title of a document would strongly influence whether workers preferred it. This notion is similar to the “TitleStat” notion introduced by Buckley et al. [106]. Note that we empirically tested that the presence of query terms in the title of document has no correlation with TF-IDF score of document. Therefore, it made sense to analyze the influence of query terms in title of documents separately. We measure document title relevance as the number of query terms appearing in the document’s title. Note that many documents either have no title or have no query terms in the title; these documents are assigned a score of zero. In our baseline, we find that query terms in the document title are well correlated with worker accuracy for most topics. This trend continues with pair preferences: the presence of more query terms in the title
Figure 4.7: TF-IDF vs. Error for relevance grades. Only half of the topics show a statistically significant result ($p < 0.05$) for TF-IDF.

Figure 4.8: TF-IDF vs. Error for pair preferences. Workers preferred non-relevant documents over relevant documents with lower TF-IDF scores.
Figure 4.9: Title Relevance vs. Error. Non-relevant documents are frequently mislabeled when their titles are rich in query terms.

Figure 4.10: Title Relevance vs. Error. Non-relevant documents are frequently preferred over relevant documents when their titles are rich in query terms.

of the non-relevant than the relevant document led to uniformly higher error rates. These plots are shown in Figure 4.9 and Figure 4.10.
4.3 Summary

In this chapter, we have analyzed the assessments made by crowd workers for two different types of relevance judgments: pair preferences and nominal relevance grades. We have observed that crowd workers have biases toward certain features of documents. For example, they prefer lengthy documents over short documents, documents with high TF-IDF scores over documents with low TF-IDF scores, and documents with query terms in the title over documents that have no query terms in the title. We found a correlation between time spent on judgment and worker quality, that workers who took more time to make judgments were more accurate. We have also found that it is easier to identity a relevant document when it is compared to a non-relevant document as compared to when it is presented on its own. We can conclude from this study that preference judgments are not always a better choice as compared to nominal judgments and one has to choose the type of relevance assessment very carefully based on the characteristics of the collection and nature of task.
Relevance judgments are usually collected on binary or multi-grade scale, but such judgments are limited by factors such as inter-assessor disagreement and the arbitrariness of grades. Previous research has shown that it is easier for assessors to make pairwise preference judgments. However, unless the preferences collected are largely transitive, it is not clear how to combine them in order to obtain document relevance scores. Another difficulty is that the number of pairs that need to be assessed is quadratic in the number of documents. In this chapter, we consider the problem of inferring document relevance scores from pairwise preference judgments by analogy to tournaments using the Elo rating system. We show how to combine a linear number of pairwise preference judgments from multiple assessors to compute relevance scores for every document. The study presented in this chapter has been published in [107]. This chapter is organized as follows: Section 5.1 gives motivation of using preference judgments, Section 5.2 presents details of combining preferences using Elo rating system, Section 5.3 describes our experimental set-up for collection of preferences, Section 5.4 presents evaluation results using preferences, Section 5.5 analyses the results, Section 5.6 summarizes this chapter.

5.1 Motivation

High quality relevance judgments are essential for the evaluation of information retrieval systems. Traditional methods of collecting relevance judgments make binary assumption about relevance i.e. a document is assumed to be either relevant or non-relevant to the information need of a user. This assumption turns relevance
judgment into a classification problem. In the modern world, search engines can easily retrieve thousands of documents at least somewhat relevant to the user’s information need. Therefore it becomes necessary to assign a ranking to these documents based on their degree of relevance. This somewhat more continuous notion of relevance cannot be expressed through binary relevance judgments; researchers have developed two ways to express non-binary relevance judgments: either consider relevance as a relative notion such that one document is more or less relevant than another document, or consider relevance as a quantitative notion and create multiple grades of relevance. The first notion of relevance can be expressed as pairwise preference judgments; the second notion can be expressed as nominal graded relevance judgments, which appear far more prevalently in the literature.

Graded relevance has two significant shortcomings. First, the total number of grades must be defined in advance, and it is not clear how this choice effects the relative measurement of system performance. Second, graded judgments require assessors to choose between arbitrarily defined grades, a choice on which different assessors can easily disagree. The alternative, pairwise preference judgments, allows the assessor to make a binary decision, freeing him or her from the difficulty of deciding between multiple relevance grades. Another advantage of using preferences is that many popular learning-to-rank algorithms, e.g. RankBoost [108] and RankNet [109], are naturally trained on preferences; thus a better training set can be obtained from direct preference judgments, as opposed to pairwise preferences inferred from nominal judgments.

There have been several attempts to use preference judgments by inferring them from absolute judgments [88] and from click data [29]. Nie et al. [110] used preferences for relevance assessments and showed that labelling effort can be reduced by focussing on top ranked documents. Chen et al. [100] also used preferences but focused more on estimating worker quality. Carterette et al. [8] did a study where assessors were asked for direct pairwise preferences as well as absolute relevance judgments for the comparison of the two assessment approaches. The authors showed that rate of inter-assessor agreement is higher on preference judgments, and that assessors take longer to make absolute judgments than preference judgments.

If a simple routine is to be used to infer document relevance from pairwise preferences, it is essential that the preferences be transitive, so that we may sort documents by preference and decide which and how many pairs to judge. Carterette et al., by collecting all $O(n^2)$ preference judgments found that the preferences they collected are transitive 99% of the time. However, the study used experts assessors.
The critical property of transitivity might not hold when judgments are collected through the much noisier process of crowdsourcing.

In order to obtain document grades (or scores) from a smaller number of preference judgments, we draw an analogy to the tournament problem. In a typical tournament, pairs of players or teams compete in matches of one or more games. The desired outcome is a final ranking (or scoring) of each competitor. A common solution is to use the Elo rating system [10], in which players are assigned ratings which are updated iteratively each time the player competes in a match. Using the Elo rating system to combine preference judgments into document grades has the following benefits:

1. The judgments do not need to be transitive. We cannot simply sort the documents by preference since humans assessors can be intransitive in their assessments; especially when we are combining preference judgments from noisy assessments (e.g. through crowdsourcing). The Elo rating system produces a ranking of documents even if the preferences are not transitive.

2. We do not need a quadratic number of pairwise assessments for inferring the relevance of documents. The Elo rating system can be applied to any number of assessments. Indeed, it can infer highly reliable relevance scores using only a linear number of pairwise assessments.

3. For any pair of documents, the document scores produced using the Elo rating system can be used to compute the likelihood of one document is more relevant than the other. In this way we can predict all $O(n^2)$ preferences while only collecting $O(n)$ judgments.

5.2 Combining Preferences

5.2.1 Elo Rating

We have collected preference judgements for each document pair from multiple assessors. We need some model for combination of preference judgements that can infer correct order of all documents from only $O(n)$ preference judgements, and it should also be able to handle ties. We have used the Elo rating system for combining $O(n)$ preference pairs. The Elo rating system is a method for calculating the relative rating of players in two player games [10]. The rating system assigns each player a rating score, with a higher number indicating a better player. Each player’s rating is updated after he or she has played a certain number of matches, increasing or decreasing in value depending on whether the player won or lost.
each match, and on the ratings of both players competing in each match—beating a highly rated player increases one’s rating more than beating a player with a low rating, while losing to a player with a low rating decreases one’s score more than losing to a player with a high rating. These scores are used in two ways: 1) players are ranked by their scores, 2) the scores can be used to compute the likelihood of each player beating every other player. If the matches are selected intelligently, this can be accomplished even if only $O(n)$ matches are played.

For our problem, we treat each document as a player and each preference judgement between two documents as a match between players. All documents enter the “tournament” rated equally. After each document “plays” a certain number of matches, we update each document’s rating according to equation 5.3. After all the matches are played, we can rank the documents by their final score. This list can be thresholded to produce absolute relevance judgements. We can also use the scores to compute transitive preference judgements.

**Mathematical Details of Elo Rating System**

The Elo rating system measures the skill level of players in relative terms. Each player’s rating depends on the rating of his or her opponents, as well as the of wins or losses. The expected score for each player in a match can be estimated from the ratings of the players. If player $A$ has rating $R_A$ and player $B$ has rating $R_B$, then the expected score of player $A$ is calculated as follows:

$$E_A = \frac{1}{1 + 10^{\frac{R_B - R_A}{400}}}$$

(5.1)

Similarly the expected score for player $B$ is as follows:

$$E_B = \frac{1}{1 + 10^{\frac{R_A - R_B}{400}}}$$

(5.2)

The $400$ in denominator is a parameter for controlling the extent to which rating difference of players will impact their chance of winning. For each 400 extra points as compared to the opponent, the chance of winning is increased ten times in comparison to the opponent’s chance of winning. When a document’s score in a match is lower than its expected score then Elo decreases its rating, similarly when a document’s score in a match is higher than its expected score, then Elo increases its rating. Suppose document $A$ has expected score $E_A$ and its actual score is $S_A$ (possible actual scores are between ) then its rating will be updated according to following equation:

$$R'_A = R_A + K(S_A - E_A)$$

(5.3)
The maximum possible adjustment per is called the K-factor. This update can be made after every match or after a tournament of matches according to the situation. For our problem, we consider each judgement for each document pair a match between two documents, and all judgements for each pair are considered a tournament between two documents. We update score of each document in a pair after each tournament i.e. all matches have been played between two documents (all 5 preference judgements have been recorded for the two documents). Suppose document A has a rating of 1500, and plays 5 matches with document B that has rating of 1550. Document A wins 3 matches, losses 1 match and draws 1 match with document B. Actual score of document A $S_A$ is $(1 + 1 + 1 + 0 + 0) 3.5$, and actual score of document B is $S_B$ is $(0 + 0 + 0 + 1 + 0.5) 1.5$. The expected score of document A $E_A$ will be $0.43 \times 5 = 2.15$ and expected score of document B $E_B$ will be $0.57 \times 5 = 2.85$. Therefore, the new rating of document A will be 1527 and new rating of document B will be 1523 (assuming $K$ to be 20).

We sort all documents based on their ratings after all $O(n)$ tournaments have been played. The final ratings of documents can be converted into relevance probabilities for each topic by normalization of Elo rating scores.

In addition to using Elo rating system for combing preferences, we also experimented with some enhancements to the basic Elo rating system. These enhancements are described as follows:

**Elo Rating with Variance**

The Elo rating system assumes that the uncertainty about a player’s skill rating does not change over time. Therefore, all skill rating updates are computed with the same variance, and any uncertainty about the changes in a player’s skills over time is not modeled. Glickman proposed to solve this problem by incorporating the variance over time in the player’s skill rating [111]. He presented the idea of modeling the belief about a player’s skills as a Gaussian distribution whose mean corresponds to the player’s rating. As a player plays more matches, the uncertainty about his/her skills is decreased, and this is reflected by a decrease in the variance of the player’s associated Gaussian distribution. Variance reflects uncertainty in document scores. Initially we have high uncertainty in document scores, but as documents play more matches our uncertainty about document scores is decreased and our confidence in document scores is increased. This is reflected by change in variance of documents. Initially all documents will have high variance. At any point during matches, the documents that have already played many matches will have lower variance. Suppose a match is played (preference judgement) between two documents, one document has already played 5 matches so its variance is low
whereas other document has not played any match so its variance is high. After the match rating of the document with higher variance will have high increase or decrease in its rating whereas the document with low variance will have smaller change in its rating. Rather than using equation 5.3, the mean rating \( R_A \) and variance \( \sigma^2 \) of each document is updated using equation 5.4 and equation 5.5 as follows:

\[
R'_A = R_A + Kg(\sigma^2_B)(S_A - E_A)
\]  
\[\tag{5.4}

\[
g(\sigma^2) = \frac{1}{\sqrt{1 + 3q^2\sigma^2}}
\]  
\[\tag{5.5}

\[
E_A = \frac{1}{1 + 10^{-g(\sigma^2_B)\frac{E_B - E_A}{200}}}
\]  
\[\tag{5.6}

\[
K = \frac{q}{\sigma^2_A + \delta^2}
\]  
\[\tag{5.7}

\[
\delta^2 = \frac{1}{q^2 \sum_{j=1}^{m} n_j g(\sigma^2_j)^2E_A(1 - E_A)}
\]  
\[\tag{5.8}

\[
\sigma^2 = \frac{1}{\sigma^2 + \delta^2}
\]  
\[\tag{5.9}

\[
q = \frac{\log 10}{200}
\]  
\[\tag{5.10}

Every document is initialized with a rating of 10 and a variance of 1.

**Use of Expert Judgements from Trap Questions** To ensure the quality of our crowd sourced preferences, we created trap questions consisting of preference pair judgements made by Information Retrieval graduate students. As these judgements were much more reliable than judgements made by crowd workers, these judgements were given as input to the Elo rating system along with the preference judgements made by crowd workers. Each expert judgement was treated as five separate matches. In this way, expert judgements were given a weight five times greater than the judgements of crowd workers.

**Prior Document Ratings using BM25 Retrieval Model and Relevance Feedback** In our preliminary formulation, all documents were given a uniform initial rating. We also experimented with non-uniform initial ratings. In our experiment, we used a standard BM25 retrieval model to generate a ranked list of documents for each topic. BM25 is a commonly used retrieval model by search engines to rank matching documents according to their relevance to a given search query. It is based on the probabilistic retrieval framework developed in the 1970s and 1980s
by Robertson, Karen Sprck Jones, and others [105].

Given a query $Q$, containing keywords $q_1, ..., q_n$, the BM25 score of a document $D$ is:

$$
\text{score}(D, Q) = \sum_{i=1}^{n} \text{IDF}(q_i) \cdot \frac{f(q_i, D) \cdot (k_1 + 1)}{f(q_i, D) + k_1 \cdot (1 - b + b \cdot \frac{|D|}{\text{avgdl}})}
$$

(5.11)

where $f(q_i, D)$ is $q_i$’s term frequency in the document $D$, $|D|$ is the length of the document $D$ in words, and $\text{avgdl}$ is the average document length in the text collection from which documents are drawn. $k_1$ and $b$ are free parameters, usually chosen, in absence of an advanced optimization, as $k_1 \in [1.2, 2.0]$ and $b = 0.75$. $\text{IDF}(q_i)$ is the $\text{IDF}$ (inverse document frequency) weight of the query term $q_i$. It is usually computed as:

$$
\text{IDF}(q_i) = \log \frac{N - n(q_i) + 0.5}{n(q_i) + 0.5}
$$

(5.12)

where $N$ is the total number of documents in the collection, and $n(q_i)$ is the number of documents containing $q_i$.

The top 10 documents for each topic were extracted and judged using the TREC 8 Qrel (the use of the Qrel could have been avoided by using expert judgments). These relevant documents were then used to extract expansion terms for each topic. Using a standard TF-IDF model, we generated 20 expansion terms from these documents. These expansion terms were used with the topic for a second phase of retrieval. We normalized the document scores in the resulting ranked list to fall between 0 and 10. These scores were used as the initial document ratings for the Elo system.

**Use of Expectation Maximization to Extract Preference Pair probabilities and Worker Quality** Expectation Minimization (EM) has been shown to work well for aggregating labels from multiple crowd workers to infer relevance of documents [47]. We have used EM to get probabilities for each preference pair in such a way that the probability reflects the chance that document A is more relevant than document B. EM [47] also incorporates some notion of worker quality so these probabilities serve as more reliable input to Elo system as compared to raw preference judgements made by crowd workers.

Expectation Maximization (EM) is an iterative method for finding the maximum likelihood estimate of the parameters of a probability distribution using a model incorporating unobserved latent variables. In our model, we treat the observed preference judgments from Crowdsource workers as being drawn from a distribution parameterized by the “quality” of each worker. The relevance grade
of each document is treated as a latent variable. For the purposes of this task, the end result is a probability distribution over each latent variable, i.e. the probability that each document is relevant.

**The EM Algorithm** We take a similar approach to Hosseini et al. [47] to implement our EM algorithm, but document pairs are the objects of interest instead of the documents themselves. Assuming there is a set of \( N \) document pairs and \( M \) workers. \( G \) is the set of preference classes. We assign each worker \( j \) a \( G \times G \) latent confusion matrix. Each element \( c_{ij}^{l,g} \) is the probability that worker \( j \) labels the document as \( l \), given ground truth \( g \). \( pr(R_i = g) \) refers to the probability that document pair \( i \) has the true preference class \( g \). \( p_g \) is the prior probability distribution of preference class \( g \in G \). \( n_{ij} \) is a binary digit indicating whether worker \( j \) has labeled document pair \( i \) as \( l \), where 1 indicates “yes”.

There are four steps in the EM algorithm. We input all preference judgements including ties, but restrict output preference classes, for each arbitrary pair, there are two preference classes in set \( G \), “more relevant” or “less relevant”.

\[
G = \{mr, lr\}
\]

There are three input preference types in set \( L \),

\[
L = \{mr, lr, tie\}
\]

**Step 1: Initialization**

We initialize the preference class for each document pair. For all document pairs \((doc1, doc2)\), we assume the probability that \(doc1\) is more relevant than \(doc2\) to be 0.5. Likewise we assume the probability that \(doc1\) is less relevant than \(doc2\) to be 0.5.

\[
Pr(R_i = mr) = 0.5
\]

\[
Pr(R_i = lr) = 0.5
\]

With the above initialization we can calculate the \(p_g \forall g \in G\). We calculate this using the sum of all document pair probabilities having class \( g \), divided by the sum of all document pair probabilities for all classes. This denominator equals the number of document pairs, because the sum of the probability distribution for each is 1 (\(\sum_{K \in G} Pr(R_i = k) = 1\)).
\[ p_g = \frac{\sum_i^n Pr(R_i = g)}{\sum_i^n \sum_k \Pr(R_i = k)} \quad (5.13) \]

**Step 2: Confusion matrix**

Estimate the maximum likelihood of worker quality, represented by a $3 \times 2$ matrix of $c_{l,g}$.

\[ M = \begin{bmatrix} c_{mr, mr}^j & c_{mr, lr}^j \\ c_{lr, mr}^j & c_{lr, lr}^j \\ c_{tie, mr}^j & c_{tie, lr}^j \end{bmatrix} \]

\[ c_{l,g}^j = \frac{\sum_i^n \Pr(R_i = g) \times n_{il}^j}{\sum_i^n \sum_k \Pr(R_i = g) \times n_{ik}^j} \quad (5.14) \]

**Step 3: Document pair preference class**

Assuming the labels observed from workers are totally independent, using the confusion matrix we obtained from the last step, for each document pair $(d_1, d_2)$ we estimate the preference relevance probability that each document pair being $mr$ or $lr$:

\[ \Pr(R_i = mr \mid C_j, \forall j \in M) = \frac{p_{mr} \cdot \prod_{j=1}^M \prod_{l=0}^L (c_{l, mr}^j)^{n_{il}^j}}{p_{mr} \cdot \prod_{j=1}^M \prod_{l=0}^L (c_{l, mr}^j)^{n_{il}^j} + p_{lr} \cdot \prod_{j=1}^M \prod_{l=0}^L (c_{l, lr}^j)^{n_{il}^j}} \quad (5.15) \]

\[ \Pr(R_i = lr \mid C_j, \forall j \in M) = \frac{p_{lr} \cdot \prod_{j=1}^M \prod_{l=0}^L (c_{l, lr}^j)^{n_{il}^j}}{p_{mr} \cdot \prod_{j=1}^M \prod_{l=0}^L (c_{l, mr}^j)^{n_{il}^j} + p_{lr} \cdot \prod_{j=1}^M \prod_{l=0}^L (c_{l, lr}^j)^{n_{il}^j}} \quad (5.16) \]

**Step 4: Iterate Step 2 and 3 until convergence**

Repeat step 2 and step 3 until results converged. We repeat calculation of worker confusion matrices and estimation of document relevance grades un-
ti the result converge. We consider this process to have converged when, for each document pair $i$, the difference between $P_r(R_i = g), \forall g \in G$ at iteration $t - 1$ and at iteration $t$ is less than or equal to 0.01 for all $g$.

5.2.2 Selection of Preference Pairs

For our experiments, we select $O(n)$ matches stochastically. The order of matches is random, since we run multiple iterations of Elo Rating system as described in section 5.3.5, this random order will not effect the performance. We wish to sample pairs in such a way that we create a bias towards relevant documents. In this way, relevant documents will play more matches than non-relevant documents, giving them more opportunities to improving their ratings and move up the list. We begin by ranking documents for each topic using the BM25 retrieval method. The first five documents all “play” one another. Each remaining document in the list plays against 5 randomly selected documents with higher ranks. In our experiments, every document plays at least 5 matches. On average, every document plays 11 matches. We sort all documents based on their ratings after all $O(n)$ matches have been played.

5.3 Experiments

5.3.1 Data

To the best of our knowledge, preference judgements are not available for any of the popularly used large IR test collections. So in order to make a comparison of our methodology for relevance judgements with methodology used by other research groups in IR, we participated in TREC Crowd Sourcing Track for year 2012. The data used in our experiments is same as released by TREC for that year’s crowd sourcing track. We will compare our methodology for collecting relevance grades from pairwise preferences to the results of the TREC 2012 Crowdsourcing track\(^1\). The goal of the track was to evaluate approaches to crowdsourcing high quality relevance judgments for text documents and images. Track participants were asked to provide new binary relevance grades, as well as probabilities of relevance, for 18,260 documents that had previously been judged with respect to ten topics (detailed description of these topics can be found in Section 4.1.1) selected randomly from the TREC 8 ad-hoc collection.

\(^1\)http://sites.google.com/site/treccrowd
5.3.2 Crowdsourcing

We crowdsourced our preference judgments using Amazon Mechanical Turk (AMT)\(^2\). Please see Section 4.1 for details of HIT design of our crowdsourcing experiments for preferences and nominal grades. We have also described in Section 4.1 how we used trap questions for quality control. As another means of ensuring the quality of the collected judgments, we also employed Expectation Maximization (EM). In this context EM, is a means of estimating the “true” pairwise preferences from crowd workers as latent variables in a model of worker quality. For every pair of documents about which we collected judgments from workers, EM provides a probability that one document beats the other. EM has been shown to work well for aggregating labels from multiple crowd workers on AMT [52], and in particular with regarding to collecting relevance judgments [47].

5.3.3 Graded Judgements

We have also collected nominal judgments for same data to make a comparison with preferences. Graded judgements can be combined by either majority vote or by Expectation Maximization (EM) [47]. We collected only 5 judgements for each document so using majority vote is not a good idea for such a small number of judgements (the total number of grades is also 5 so 5 votes is not enough to calculate majority vote, if the grades were binary then 5 votes would be reasonable for calculation of majority vote). We used average of all grades assigned to a document and used that as relevance score of the document. We have also used EM to obtain relevance grades which apart from combining judgements from multiple workers, also utilizes a notion of worker quality.

5.3.4 Experimental Set-up

We have set \(K\) equal to 20 in Elo rating update equation 5.3 and initial mean and variance of each document is set to 100 and 10 respectively. Every document pair selected for comparison was judged by 5 crowd workers. For each of the 10 topics, participants of TREC Crowdsourcing track were required to provide both binary relevance judgements and probabilities of relevance for each document. Probabilities were produced by normalizing the Elo scores. We computed binary relevance grades by labelling the top \(R\) documents for each topic as relevant, where \(R\) is the actual number of relevant documents according to the TREC 8 Qrel. We need to convert rank list to binary grades for measuring LAM, we don’t need it for AUC and MAP. We can estimate \(R\) by reading few top documents from the rank list or

\(^2\)http://www.mturk.com
we can use Elo ratings for calculating the threshold of relevance.

### 5.3.5 Iterations of Elo Rating

In Elo rating system, the score of each document depends on the score of its opponent document in a match. The order in which matches are played has an impact on scores of documents. For example, if a document wins a match against a relevant document, and the relevant document has not played any match yet, then the score of the document would not increase significantly. If the relevant document has already played few matches and has raised its score, then winning a match against it would increase the score of a document to a large extent. Because of this, if we run only one iteration of Elo rating algorithm (through all pairs) then some document scores may not be reliable; we instead run several iterations of Elo rating algorithm so that scores of documents converge. Figure 5.1 shows the relationship of number of Elo rating iterations to percentage of pairs inverted, after the initial run through all pairs. Note that as we run more iterations, the percentage of pairs whose order is changed decreases. Stopping criteria for Elo iterations is when percentage of inverted pairs in top 200 documents is less or equal to 4%.

### 5.3.6 Baseline

In order to measure the quality of our Elo-based system, we also implemented a naive system as a baseline. In our naive system, each document is given a score based on the percentage of its matches that it won and the number of matches it
competed in. The score of a document $A$ is calculated as:

$$\text{score}(A) = \lambda \frac{\text{wins}_A}{\text{matches}_A} + (1 - \lambda) \frac{\text{matches}_A}{\text{matches}}$$

(5.17)

where $\text{wins}_A$ is number of matches won by document $A$, $\text{matches}_A$ is total number of matches played by a document $A$, and $\text{matches}$ is total number of matches played. Since we did not have enough data to properly tune $\lambda$, $\lambda$ is set to 0.5.

5.4 Results

5.4.1 Evaluation of Quality of Rank Lists

We combined relevance judgements, collected using preferences and nominal grades, using various methodologies as described in previous section. The quality of these rank lists was evaluated using accuracy (AUC) (defined as area under ROC curve) and Mean Average Precision (MAP). Evaluation of binary judgements is done using Logistic Average Misclassification (LAM). LAM and AUC have been used by TREC Crowd Sourcing track for evaluation of binary relevance judgements, LAM can be defined as follows:

$$LAM = \logit^{-1}\left(\frac{\logit(fpr) + \logit(fnr)}{2}\right)$$

(5.18)

where $fpr$ is the false positive rate, $fnr$ is the false negative rate, and $\logit$ is defined as follows:

$$\logit(p) = \log\left(\frac{p}{1 - p}\right)$$

(5.19)

$$\logit^{-1} = \frac{e^x}{1 + e^x}$$

(5.20)

$fpr$ and $fnr$ are smoothed for calculation as follows:

$$fpr = \frac{|FP| + 0.5}{|FP| + |TN| + 1}$$

(5.21)

where $FN$ is False Negative, $TN$ is True Negative, $FP$ is False Positive, and $TP$ is True Positive.

Table 5.1 shows the AUC results of the Elo rating system with various enhancements. These results show that accuracy improves when we apply our Elo system to preference pair probabilities produced from EM (EloEM). These results also show that Elo rating produces slightly better accuracy with variance as compared to Elo rating without variance but it is not significant. The results also improve by adding a small number of judgements from trap questions. Finally, Table 5.1
<table>
<thead>
<tr>
<th>Topic ID</th>
<th>Elo Rating</th>
<th>Expectation Maximization</th>
<th>Elo Rating with Variance</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>Using Trap Question Judgements</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Using Non-uniform Prior Documents</td>
</tr>
<tr>
<td>411</td>
<td>0.811</td>
<td>0.862</td>
<td>0.857</td>
</tr>
<tr>
<td>416</td>
<td>0.940</td>
<td>0.939</td>
<td>0.944</td>
</tr>
<tr>
<td>417</td>
<td>0.897</td>
<td>0.914</td>
<td>0.887</td>
</tr>
<tr>
<td>420</td>
<td>0.834</td>
<td>0.853</td>
<td>0.823</td>
</tr>
<tr>
<td>427</td>
<td>0.871</td>
<td>0.907</td>
<td>0.882</td>
</tr>
<tr>
<td>432</td>
<td>0.536</td>
<td>0.558</td>
<td>0.637</td>
</tr>
<tr>
<td>438</td>
<td>0.731</td>
<td>0.774</td>
<td>0.708</td>
</tr>
<tr>
<td>445</td>
<td>0.748</td>
<td>0.843</td>
<td>0.790</td>
</tr>
<tr>
<td>446</td>
<td>0.716</td>
<td>0.865</td>
<td>0.720</td>
</tr>
<tr>
<td>447</td>
<td>0.995</td>
<td>1</td>
<td>0.859</td>
</tr>
<tr>
<td>all</td>
<td>0.808</td>
<td>0.851</td>
<td>0.811</td>
</tr>
</tbody>
</table>

Table 5.1: Accuracy (AUC) results using Elo rating system with various enhancements

<table>
<thead>
<tr>
<th>Topic Id</th>
<th>EloEM</th>
<th>Average Grades</th>
<th>Mean Grades using EM</th>
</tr>
</thead>
<tbody>
<tr>
<td>411</td>
<td>0.2568</td>
<td>0.5028</td>
<td>0.3789</td>
</tr>
<tr>
<td>416</td>
<td>0.4586</td>
<td>0.1430</td>
<td>0.1790</td>
</tr>
<tr>
<td>417</td>
<td>0.3618</td>
<td>0.2806</td>
<td>0.2692</td>
</tr>
<tr>
<td>420</td>
<td>0.2023</td>
<td>0.2471</td>
<td>0.2638</td>
</tr>
<tr>
<td>427</td>
<td>0.4658</td>
<td>0.2257</td>
<td>0.2019</td>
</tr>
<tr>
<td>432</td>
<td>0.0417</td>
<td>0.1265</td>
<td>0.0671</td>
</tr>
<tr>
<td>438</td>
<td>0.3119</td>
<td>0.3484</td>
<td>0.3324</td>
</tr>
<tr>
<td>445</td>
<td>0.3128</td>
<td>0.3849</td>
<td>0.4408</td>
</tr>
<tr>
<td>446</td>
<td>0.4207</td>
<td>0.4262</td>
<td>0.4970</td>
</tr>
<tr>
<td>447</td>
<td>0.9875</td>
<td>0.6624</td>
<td>0.9588</td>
</tr>
<tr>
<td>All</td>
<td>0.3820</td>
<td>0.3348</td>
<td>0.3589</td>
</tr>
</tbody>
</table>

Table 5.2: Evaluation Results using Mean Average Precision for preference based relevance judgements (EloEM) and graded relevance judgements (average grades, and mean grade using EM)
also shows the improvement achieved by using non-uniform prior ratings of documents instead of uniform initial document ratings.

Figure 5.2 and Figure 5.3 compare the Elo rating system and average grades to the median scores of the systems that participated in the TREC 2012 crowd sourcing task. Our Elo rating system and mean grades using EM exhibit higher than the median AUC and lower than median LAM (smaller value of LAM reflects higher quality of results) score for all but one topic on submitted runs. The performance of nominal judgements and preference judgements is roughly same and none of these methods performs better than the other over all topics.

5.4.2 Comparison With Nominal Grades

Table 5.2 shows MAP of rank lists produced from preference judgements (EloEM) and nominal judgements (average grades, and mean grades using EM). These results show that using EM for extracting relevance grades improves MAP of the rank list. This table shows a comparison of MAP results per topic for preference judgements and graded judgements. We cannot infer from these results that one type of judgement method is better over the other for all topics. For example, for topic 416, MAP using preference judgements is much better than MAP using graded judgements, but for topic 411, MAP using graded judgements is much better than MAP using preference judgements. We looked at few top ranked and bottom ranked relevant documents for these topics, for both types of judgements, to find out the reason for this drastic variation of results per topic. We have found that for topics where preferences have better performance, relevant documents were paired against non-relevant documents so they got very high scores. Similarly for topics where preferences have worse performance, some non-relevant documents (somewhat relevant and related to the topic) are paired against other very non-relevant documents, so they get very high scores. These results show that the way we choose preference pairs has a huge impact on the quality of relevance judgements. These results show potential of preference judgements, given we select preference pairs using some strategy to maximize the score of relevant documents.

5.5 Analysis

Results of our preliminary experiments using preferences and nominal grades are roughly of same quality. These experiments were our first step towards exploration of preference judgements so there were some weaknesses in our methodology for collecting and combining preferences. Due to these weaknesses preference judgements have not shown any significant improvement over nominal judgements. We have done some analysis of relevance judgements collected using both methods
**Figure 5.2:** Comparison of AUC scores using EloEm method, mean grades using EM and Median of TREC participant runs

**Figure 5.3:** Comparison of LAM scores using EloEm method, mean grades using EM and Median of TREC participant runs
and identified some issues that can be resolved by using preference judgements. Following sections describe weaknesses of our current methodology and next section proposes solutions for eliminating these weaknesses.

5.5.1 Weaknesses in Design of Matches for Elo Algorithm

For selection of document pairs, we used a prior rank list based on BM25 relevance scores of documents. Every document was paired with 5 randomly selected documents ranked higher than that document. The goal behind using this strategy was to give a chance to documents at top of the prior rank list to play more matches against other documents. This strategy was successful in achieving this goal but it also created few problems as listed below:

1. If we have a perfect prior rank list, then it would have all relevant documents ranked higher as compared to all non-relevant documents. Using this perfect prior rank list, our strategy for designing matches would give a chance to relevant documents to play more matches, thus increasing score of relevant documents. Since we don’t have a perfect prior rank list so it has many non-relevant documents at top, along with many relevant documents. As a result, many non-relevant documents played a large number of matches against other documents. Table 5.3 shows comparison of average and standard deviation of matches played by all documents and documents ranked in top 10 documents in the final rank list produced by Elo algorithm. It is clear from this table that documents ranked higher by Elo algorithm played much more matches as compared to other documents. We plan to change our strategy for selection of document pairs so that all documents play almost equal number of matches.

2. Our document pair selection strategy does not ensure that every document gets a chance to play against equal number of relevant and non-relevant documents. As a result, some non-relevant documents play only against other more non-relevant documents and win majority of matches. These non-relevant documents get high Elo scores and are ranked at top of the rank list dropping the quality of rank list. Table 5.4 shows the outcomes of matches played by some non-relevant document against other documents. These non-relevant documents won most of the matches because most of the time they were playing against other more non-relevant documents. We plan to change our strategy for the selection of document pairs so that every document gets a chance to play against approximately equal number of relevant documents and non-relevant documents.
<table>
<thead>
<tr>
<th>Topic ID</th>
<th>Average # of matches</th>
<th>Standard Deviation of # of matches</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>All Documents</td>
<td>Non-relevant Documents Ranked in Top 10</td>
</tr>
<tr>
<td>411</td>
<td>11.7</td>
<td>19.6</td>
</tr>
<tr>
<td>416</td>
<td>13.9</td>
<td>34.5</td>
</tr>
<tr>
<td>417</td>
<td>10.1</td>
<td>20.5</td>
</tr>
<tr>
<td>420</td>
<td>13.4</td>
<td>29.8</td>
</tr>
<tr>
<td>427</td>
<td>10.9</td>
<td>23</td>
</tr>
<tr>
<td>432</td>
<td>10.5</td>
<td>16</td>
</tr>
<tr>
<td>438</td>
<td>12.5</td>
<td>18.5</td>
</tr>
<tr>
<td>445</td>
<td>11.4</td>
<td>26.2</td>
</tr>
<tr>
<td>446</td>
<td>13</td>
<td>31</td>
</tr>
<tr>
<td>447</td>
<td>12.9</td>
<td>16</td>
</tr>
</tbody>
</table>

Table 5.3: Average and standard deviation of number of matches played by a document per topic

<table>
<thead>
<tr>
<th>Document ID</th>
<th>Number of Matches Played</th>
<th>No. of Wins</th>
<th>No. of Ties</th>
<th>No. of Loses</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Against Relevant Documents</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LA050790-0039</td>
<td>3</td>
<td>22</td>
<td>18</td>
<td>5</td>
</tr>
<tr>
<td>LA021390-0070</td>
<td>14</td>
<td>27</td>
<td>29</td>
<td>2</td>
</tr>
<tr>
<td>LA052290-0102</td>
<td>17</td>
<td>20</td>
<td>26</td>
<td>5</td>
</tr>
<tr>
<td>FT921-13642</td>
<td>4</td>
<td>24</td>
<td>18</td>
<td>7</td>
</tr>
<tr>
<td>FT931-3144</td>
<td>4</td>
<td>16</td>
<td>14</td>
<td>4</td>
</tr>
</tbody>
</table>

Table 5.4: Outcome of all matches played by some non-relevant documents that have high Elo score for topic ID 446.
5.5.2 Wrong Judgements by Crowd Workers

Crowd workers are not trained persons for relevance assessment tasks, so sometimes they make mistakes in relevance judgements of documents. We have done some analysis of most egregious mistakes made by crowd workers, by reading documents at various ranks in rank lists, produced from preference judgements and graded judgements. We made some observations about biases of crowd workers towards certain characteristics of documents, these are explained in the following.

1. We have found some relevant documents at bottom of rank lists produced from crowd sourced relevance judgements. These relevant documents have one of the following two characteristics:

   - The document is very short, it is almost one paragraph long. Crowd workers seem to have a bias towards lengthy documents and they tend to give low preference to short documents even if they are relevant.
   - The document is lengthy and the main topic of document is not relevant to the query but it has one or two lines relevant to the query. The relevant lines are lost in the sea of non-relevant text.

2. We examined documents at top of rank lists and found following characteristics in those documents:

   - The document had a title, and the title contained at least one of the query words. This pattern was found in non-relevant documents at top of rank lists. The crowd workers seem to have bias towards documents that have title containing one of the query words. They don’t read the entire document carefully and make their judgements based on the title of the document.
   - The document has a large number of query words. The non-relevant documents at top of rank lists have this characteristic and this might be the reason why crowd workers preferred these documents.

3. For topics that have multiple aspects, some crowd workers prefer one aspect over another. For example, for topic 446 (title tourist violence), there are two aspects of the topic, tourism and violence. Some documents are only about one of the aspects of this topic, so some workers preferred a non-relevant document, that only talks about violence in great detail, over a relevant document that talks about tourism but also has some relevant information about violence against tourists. Both documents are about two different aspects of
the topic and only one document has small amount of relevant information. This mistake is also being made due to same reason that workers are not spending enough time on reading documents.

These mistakes are highly correlated with the mistakes made by an IR model. One consequence of this observation is that when researchers design experiments with synthetic judgments and introduce noise in judgments to simulate noisy crowd workers, then this noise should not be added randomly. For example suppose there was 65% noise in crowdsourced judgments for TREC 8 data used in these experiments. Suppose we obtain two rank lists for this data using some IR retrieval model. We add synthetic judgments (Qrel with 65% noise) to first rank list and add original crowdsourced judgments (with 65% noise) to second rank list. MAP of former rank list will be better than MAP of latter rank list although the noise level in both synthetic and real judgments is same. The reason for this behavior is that mistakes of crowd workers are highly correlated with IR retrieval model’s mistakes. When we use real crowdsourced judgments on top of rank list retrieved by IR model then there is not much improvement in the MAP score of list since crowd workers make same mistakes as IR retrieval model. Most of relevant document at bottom of rank list will not be identified by crowd workers as relevant documents. If we use synthetic judgments with same level of noise on top of rank list retrieved by IR model then there is more improvement in MAP score of list. The reason is that synthetic judgments with random noise will fix some mistakes of the IR retrieval model.

5.5.3 Assessor Agreement

Table 5.5 and Table 5.6 show assessor agreement between crowd workers on preferences and nominal judgements respectively. These tables show agreement about each judgement over all assessors. Each cell \((J_1, J_2)\) is the probability that one assessor would say \(J_2\) (column) given that another said \(J_1\) (row). They are normalized by row, which is why columns do not add to 1. The assessor agreement is not very high among crowd workers for both types of judgements. Lack of careful judgements from crowd workers due to low incentive can be the reason for this low assessor agreement.

5.6 Summary

Preference judgments are easier for assessors to produce and are more useful for training learning-to-rank algorithms. However, their use has been limited due to the polynomial increase in the number of judgments that need to be collected. In
Table 5.5: Inter-assessor agreement for preference judgements

<table>
<thead>
<tr>
<th></th>
<th>A &gt; B</th>
<th>A = B</th>
<th>A &lt; B</th>
</tr>
</thead>
<tbody>
<tr>
<td>A &gt; B</td>
<td>0.24</td>
<td>0.45</td>
<td>0.32</td>
</tr>
<tr>
<td>A = B</td>
<td>0.3</td>
<td>0.42</td>
<td>0.28</td>
</tr>
<tr>
<td>A &lt; B</td>
<td>0.25</td>
<td>0.35</td>
<td>0.4</td>
</tr>
</tbody>
</table>

Table 5.6: Inter-assessor agreement for graded judgements

<table>
<thead>
<tr>
<th></th>
<th>Grade 0</th>
<th>Grade 1</th>
<th>Grade 2</th>
<th>Grade 3</th>
<th>Grade 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Grade 0</td>
<td>0.23</td>
<td>0.17</td>
<td>0.24</td>
<td>0.23</td>
<td>0.12</td>
</tr>
<tr>
<td>Grade 1</td>
<td>0.28</td>
<td>0.22</td>
<td>0.2</td>
<td>0.2</td>
<td>0.11</td>
</tr>
<tr>
<td>Grade 2</td>
<td>0.28</td>
<td>0.14</td>
<td>0.27</td>
<td>0.2</td>
<td>0.11</td>
</tr>
<tr>
<td>Grade 3</td>
<td>0.28</td>
<td>0.14</td>
<td>0.2</td>
<td>0.28</td>
<td>0.1</td>
</tr>
<tr>
<td>Grade 4</td>
<td>0.26</td>
<td>0.15</td>
<td>0.2</td>
<td>0.2</td>
<td>0.2</td>
</tr>
</tbody>
</table>

In this chapter, we have shown how the Elo rating system can be used to combine a linear number of preferences to obtain either an ordered list of documents or document relevance scores. The results of our experiments are encouraging and demonstrate the potential of our Elo-based system for inferring the relevance of documents from a linear number of pairwise preference judgments.
Chapter 6

A Bayesian Framework for Optimally Combining Assessments from Crowd Workers and Experts

In this chapter we present a Bayesian framework for combining assessments from crowd workers and experts. This study has been accepted for funding of 5 million dollars NSF Grant (Grant Id 1256172). This chapter is organized as follows: Section 6.1 presents motivation for creating our Bayesian framework, Section 6.2 provides details of the proposed framework and theoretic aspects of our model; Section 6.3 describes the experimental setup and data used to test the framework; Section 6.4 describes the results and analysis of our experiments; and Section 6.6 summarizes this chapter.

6.1 Motivation

Traditionally, the assessors of documents in a test collection are trained experts; however, the growing size of IR test collections has made it cost-prohibitive to hire professional assessors for obtaining all necessary judgments. Crowdsourcing has been explored recently as an alternative to expensive expert assessments but quality of crowdsourced assessments is very low. For quality control, crowdsourced assessments require redundancy, adversarial protection, quality estimation, and so on. This raises numerous questions such as: How many crowd assessments do I need for each document? Which documents would benefit the most from additional crowd assessments? How many crowd assessments should I seek before
deciding that there is too much disagreement among crowd workers to obtain an accurate overall assessment, in which case I might well be better off asking an expert?

Our Bayesian framework for selecting and combining assessments from experts and crowd workers is designed to answer just such questions in a mathematically principled way. Instead of using just experts or just crowd workers, we aim to make the most of an assessing budget by combining the assessing resources and asking the appropriate type of question to each assessor: simpler questions to cheaper crowd sourced workers and more challenging questions to more expensive experts. To facilitate such a process, we propose a Bayesian framework for combining potentially different types of judgments from mixed crowd and expert workers and for the online selection of the most cost-effective (question, assessor) pair at any given time. This requires modeling the quality, cost, and benefit of various types of workers for various types of questions.

It has been shown in our study [104] presented in Chapter 4 that crowd workers are more likely to make mistakes on specific questions and/or documents. Based on conclusions from that study, we categorize documents, for each query, into different contexts based on their features (for example “short documents rich in query terms” vs. “long documents sparse in query terms”). We then represent worker accuracy for each of these contexts with a generative model which we term a “contextual confusion matrix”. This model is implemented as a conditional distribution between the true and observed document assessments grades—per assessor type and context—and it is learned from training data.

The proposed framework uses these worker accuracy models for various contexts and decides which questions should be asked to which type of assessor, based on their model of accuracy for each context, in order to maximize benefit per unit cost. The costs themselves are simply input parameters; everything else is part of the probabilistic Bayesian framework. In order to maximize the benefit per unit cost, one must define the benefit. Our objective is to reduce our uncertainty about the correct assessment for every document, but in particular for those documents which are most likely to be most relevant, as they will have the most powerful effect on the accuracy of retrieval evaluation. Given a document’s current assessment distribution over grades, there are any number of measures of the uncertainty in the “correct” grade, such as entropy, variance, and so on. In order to emphasize the importance of obtaining the “correct” grade for those documents likely to be “highly” relevant, we consider the variance in the distribution over gains (in the sense of nDCG) as opposed to grades, since gains (typically defined as $2^l - 1$ where
is the grade) place greater emphasis on higher grades. As such, the components of our Bayesian framework are as follows:

- Contextual confusion matrix: a context-based conditional distribution of assessor grade given true grade. Learned from training data.

- Prior: a prior probability distribution over true grades (per context) before any assessments are observed. The prior is learned from training data or inferred from a retrieval model such as BM25.

- Posterior: a probability distribution over true grades per document once some judgments are observed. Updated in a Bayesian fashion with each new assessment.

- Variance in gain: the variance in gain associated with the current posterior distribution over grades, calculated for each document.

- Objective: maximize the expected reduction in variance in gain per unit cost for any assessment.

6.2 Framework for Optimal Selection of Questions and Assessors

We consider the problem of optimally selecting and obtaining relevance assessment from diverse sets of crowd workers in order to simultaneously maximize utility and minimize cost.

6.2.1 Modeling Crowd Workers

Lets assume that crowd workers can be modeled and characterized by a confusion matrix over true labels and assessments. We denote this confusion matrix by a conditional distribution \( p(A|L) \) over two random variables \( L \) and \( A \), where \( L \) corresponds the true label and \( A \) the crowd worker assessment. The value \( p(l, a) = p(a|l) \cdot p(l) \) corresponds to the following random experiment: Given a collection of documents \( D \) to be assessed, choose a document \( d \in D \) at random. What is the likelihood that the document has true label \( l \) and is given an assessment \( a \) by the crowd worker? Note that in this formulation, \( p(L, A) \) is a true joint distribution, so its marginals \( p(L) \) and \( p(A) \) effectively correspond to prior probabilities over true labels and crowd worker assessments, and its conditionals \( p(L \mid A) \) and \( p(A \mid L) \) correspond to the probabilities of a true label given an assessment and an assessment given a true label, respectively.
We further note that these confusion matrices can be contextualized to incorporate query, document, worker-type, other features. One simple contextualization is to consider worker-type and have separate confusion matrices for crowd workers and experts, for example. Note that each confusion matrix must have the same prior, as this is a function solely of the document collection \( D \). In practice, we calculate \( p(L) \) using normalized BM25 scores for a training set queries and documents; similarly the confusion matrices \( p(L|A) \) are trained on training data, separately for each context.

### 6.2.2 Combining Crowd Worker Assessments

Given any document \( d \in D \), we have a prior distribution \( p(L) \) over the possible true labels of \( d \). Now suppose that \( d \) is given assessments \( a_1, a_2, \ldots, a_n \) by \( n \) different crowd workers; we combine this evidence to obtain a posterior distribution \( p(L | A_1 = a_1, A_2 = a_2, \ldots, A_n = a_n) \) over labels? Consider any candidate label \( l_i \). Applying Bayes Law, we have

\[
p(L = l_i | A_1 = a_1, \ldots, A_n = a_n) = \frac{p(A_1 = a_1, \ldots, A_n = a_n | L = l_i) \cdot p(L = l_i)}{p(A_1 = a_1, \ldots, A_n = a_n)} = \frac{p(A_1 = a_1, \ldots, A_n = a_n | L = l_i) \cdot p(L = l_i)}{\sum_{l_j} p(A_1 = a_1, \ldots, A_n = a_n | L = l_j) \cdot p(L = l_j)}.
\]

We now make the standard naive Bayes independence assumption. This is likely to hold true, especially if the confusion matrices are sufficiently contextualized. For example, two separate crowd workers are both likely to mistake a relevant document as non-relevant if the document is long and contains no query terms, but this would be reflected in a high (contextualized) conditional probability of a non-relevance assessment given a relevant document, rather than a presumed dependence between the workers (e.g., as if they were colluding).

In the absence of sufficient contextualization, the crowd worker assessments may appear to be dependent—because, in fact, they are each dependent on the (hidden) context. Mathematically, in what follows, we assume a sufficient (implicit) context \( C \), such that

\[
p(A_i = a_i, A_j = a_j | L = l_k, C) = p(A_i = a_i | L = l_k, C) \cdot p(A_j = a_j | L = l_k, C).
\]  

(6.1)

We can use the above to actually test whether our context is sufficient, in a data-driven manner: If Equation 6.1 does not hold (because our worker make corre-
lated mistakes), then our (implicit) context $C$ is insufficient, and new features must be added, thus leading to richer confusion matrices. We can actually measure the quality of our context $C$ information-theoretically by comparing, for example, $I(L; A_i, A_j)$ vs. $I(L; A_i) + I(L; A_j)$.

so the posterior (giving probability of true grade) becomes

$$p(L = l_i | A_1 = a_1, \ldots, A_n = a_n) = \frac{p(L = l_i) \cdot \prod_{k=1}^{n} p(A_k = a_k | L = l_i)}{\sum_{l_j} p(L = l_j) \cdot \prod_{k=1}^{n} p(A_k = a_k | L = l_j)}.$$  \hspace{1cm} (6.2)

Note that the marginal $p(L)$ and conditionals $p(A | L)$ can all be derived from the confusion matrix $p(L, A)$, and therefore the above is calculable. Thus, Equation 6.2 can be used to combine the evidence obtained from multiple crowd workers to obtain a posterior distribution over labels.

6.2.3 Selecting Crowd Worker Assessments

At any point in the assessment process, each document $d \in D$ will have a posterior distribution

$$p^n_d = p^n_d (L | A_1 = a_1, \ldots, A_n = a_n)$$  \hspace{1cm} (6.3)

obtained from having asked some $n$ questions of assessors. Note that $n$ will vary throughout the assessment process, and it may not be fixed across documents at any point in the process.

Variance-reduction. It might be useful to reduce the uncertainty of documents thought to have high relevance (as, for instance, measured by the expected value of $p^n_d$) than those thought to have low relevance. Consider the following strategy which has this effect. For each document $d$, we have a current posterior distribution $p^n_d$ over grades (Equation 6.3). For any possible label $l$, we have a gain (in the sense of nDCG), such as $g(l) = 2^l - 1$. For this document $d$, we can calculate our mean (expected) label

$$E_{p^n_d}[L] = \sum_l p^n_d(l) \cdot l$$

the variance on our label

$$Var_{p^n_d}[L] = \sum_l p^n_d(l) \cdot (l - E_{p^n_d}[L])^2$$

$$= \left( \sum_l p^n_d(l) \cdot l^2 \right) - E^2_{p^n_d}[L]$$

$$= E_{p^n_d}[L^2] - E^2_{p^n_d}[L]$$
our mean (expected) gain

\[ E_{p_d^n}[G] = E_{p_d^n}[2^L - 1] = \sum_l p_d^n(l) \cdot (2^l - 1) \]

and the variance in our gain

\[ \text{Var}_{p_d^n}[G] = \sum_l p_d^n(l) \cdot \left((2^l - 1) - E_{p_d^n}[G]\right)^2 \]

\[ = \left(\sum_l p_d^n(l) \cdot (2^l - 1)^2\right) - E_{p_d^n}[G]^2 \]

\[ = E_{p_d^n}[G^2] - E_{p_d^n}[G]^2. \]

Note that we are certain about a document’s label if and only if the variance in that label is zero; similarly, we are certain about a document’s label if and only if the variance in the gain is also zero. ¹ The difference is that the variance in gain is (potentially) much higher for documents with high grades than for those with low grades, so reducing this variance places a premium on learning the “true” grade for documents with high grades—precisely those of interest.

In either sense, variance is very much like bits of uncertainty: Our goal is to reduce the variance to zero, just as our goal is to reduce bits of uncertainty to zero. However, as discussed, variance has the property that it very much emphasizes documents with high grades and gains, and this is especially true when considering variance in gain. For example, consider three documents, one of which has a 50/50 chance of being grade 0 or 1, a second which has a 50/50 chance of being grade 3 or 4, and a third which has a 50/50 chance of being grade 0 or 4. Each document has exactly 1 bit of uncertainty, but they have very different variances in gain: 0.25, 16, and 56.25, respectively. Thus, the variance in gain criteria would greatly dictate that we concentrate on the 0 vs. 4 document, then the 3 vs. 4 document, then the 0 vs. 1 document, and this is probably correct given the task at hand. Note that this formulation is fairly generic in that we can use any reasonable gain function; \( g(l) = 2^l - 1 \) is fairly standard, and the identity function \( g(l) = l \) would devolve to variance in the label. We can calculate

- \( p_{d, a}^n \), the posterior label distribution for any particular document \( d \), given a

---

¹Assuming that the gain function maps different labels to different gains. This is true for all strictly monotonically increasing gain functions, which are the gain functions of interest.
particular crowd worker assessment $a$

$$p^n_{d,a} = p^n_{d,a}(L | A_l = a_1, \ldots, A_n = a_n, A_{n+1} = a),$$

- the posterior variance in our gain, based on $p^n_{d,a}$ which assumes assessment $a$

$$\text{Var}_{p^n_{d,a}}[G] = E_{p^n_{d,a}}[G^2] - E^2_{p^n_{d,a}}[G],$$

- the resulting variance reduction due to assessment $a$, i.e., the difference between the current and new variance, given assessment $a$

$$VR(A_{n+1} = a) = \text{Var}_{p^n_{d,a}}[G] - \text{Var}_{p^n_{d,a}}[G],$$

- $p(a)$, the probability of obtaining assessment $a$ from the particular crowd worker, given our current belief in the true label.

The probability $p(a)$ of obtaining a particular assessment $a$ is dependent on two factor: (1) the probability that the true label is some particular value $l$ and (2) the conditional probability that the assessor returns $a$, given that the true label is $l$. Our current estimate of (1) is given by $p^n_{d}$, and (2) can be obtained from the confusion matrix associated with the particular assessor.

$$p(a) = \sum_l p^n_{d}(l) \cdot p(a | l), and$$

- the expected variance reduction

$$E[VR] = \sum_a p(a) \cdot VR(A_{n+1} = a).$$

We can calculate the expected variance reduction for all combinations of documents and assessors, and in combination with the assessor costs, we can calculate all expected price-performance ratios, i.e., the variance reductions obtained per unit cost. For every document-assessor pair, we calculate the value of the following objective function (expected reduction in variance divided by cost of assessor).

$$ObjectiveFunction = \frac{E[VR]}{Cost_{Assessor}}$$

Greedily, we pick the document-assessor pair with the best price-performance ratio. Figure 6.1 shows system flow of online Bayesian framework. Initially we have
set of documents with some associated uncertainty in grade (measured by variance in probability distribution of grades). We calculate the objective function (expected variance reduction per unit of cost) for each document-assessor pair using assessment quality information from confusion matrices for assessors. Next step is to pick the document-assessor pair with highest objective function value and get new assessment. Once we get new assessment for the selected document from the selected assessor, we update the probability distribution of grades of that document. This changes the variance of probability distribution of grades and the objective function value (expected variance reduction per unit of cost) for that document is also updated. We select the next document-assessor pair with best objective function value and this cycle continues until all budget is exhausted.

6.3 Experiments

In this section we describe the experimental design for the collection of relevance judgements from crowd workers. We have tested our framework using two types of assessors: crowd workers and experts. We consider NIST\(^2\) assessors as experts for this framework. NIST assessors are retired NSA (National Security Agency) analysts and they are highly trained for the document relevance assessment task. We have used Amazon Mechanical Turk to collect the crowd worker assessments, up to 10 per document.

\(^2\)www.nist.gov/
HIT Design

The HITs we created to collect relevance judgments from crowd workers had the following design. After accepting the assignment, workers were shown the instructions (Figure 6.2 top), where we explained that documents should be judged strictly based on whether they provide information about the query. We have used few example documents for each relevance grade to train the crowd workers for relevance assessments task. The crowd workers can read these documents using the provided link. This training helps crowd workers to decide which document belongs to which class of relevance grades.

After dismissing the instructions, the worker is shown the interface presented in Figure 6.2 (bottom). The “Title” field of a TREC topic is displayed on top, along with its description. This information describes in detail what constitutes a relevant document for this query. Below the query information is a series of questions. Each question is about one document and posts the corresponding link to the document. The interface allows workers to rate the document using one of the radio button options (relevance grades) displayed for each question. We have used the same relevance grades used in TREC 2013 Crowdsourcing Track [36]. The four grades...
are “Key” (grade 3), “Highly Relevant” (grade 2), “Relevant” (grade 1), and “Non-relevant” (grade 0).

Each HIT consisted of 10 documents for the same topic, and it had a time limit of 30 minutes. The order in which the documents were displayed was randomized. We collected 10 crowdsourcing assessments for each document. Workers were paid $0.2 for each approved HIT for graded judgements. The cost per judgment for expert assessor was estimated to $0.6 per assessment (private discussion with TREC coordinators).

Quality Control

The crowd workers we employed on AMT have no particular training in assessing document topical relevance, so we needed a means of verifying the quality of their work. There have been many studies on assessing worker quality for crowd sourcing platforms such as AMT [47]. We used trap questions in our study to ensure that workers are making a reasonable effort, instead of clicking randomly.

A trap question is a question inserted into the HIT which looks the same as any other, but for which a correct answer is known ahead of time. We used expert assessments for creation of around 20 trap questions for each topic. We then mixed two of these trap questions, selected at random, into each HIT. We approved a HIT for payment if average absolute difference between true grade and labelled grade on trap questions was at most 1.

6.3.1 Data

We have used the data from 9 out of 10 topics proposed by the TREC 2013 crowdsourcing track [36]. (We did not use one of these topics because it was navigational and had only one relevant document that had a missing image so it could not be judged as relevant by any assessor). The official list contains a total of 3470 documents. Out of these 3470 documents, about 1124 are marked as relevant by the TREC expert assessors.

Topics

Following is a list of 9 topics with descriptions from TREC 2013 crowdsourcing track used in these experiments.

- Topic 214: Capital gains tax rate
  - Description: What does the US capital tax rate consist of and how is it broken down?

- Topic 216: Nicolas Cage movies
• Description: What movies has Nicolas Cage appeared in?

• Topic 221: Electoral college 2008 results
  – Description: What were the results of the electoral college for the 2008 US presidential race?

• Topic 227: I will survive lyrics
  – Description: Find the lyrics to the song “I Will Survive”.

• Topic 230: World’s biggest dog
  – Description: What is the world’s biggest dog?

• Topic 234: Dark chocolate health benefits
  – Description: What are the health benefits associated with eating dark chocolate?

• Topic 243: Afghanistan flag
  – Description: Find pictures of the Afghanistan flag.

• Topic 246: Civil war battles in South Carolina
  – Description: Which civil war battles were fought in South Carolina?

• Topic 250: Ford edge problems
  – Description: What problems have afflicted the Ford Edge car model?

6.3.2 Contextual Confusion Matrices

Our framework uses contextualized confusion matrices for estimating the accuracy of crowd workers for particular types of questions. It has been shown in previous research [104] that document length and frequency of query terms have a significant impact on crowd worker accuracy, so we have used these two features to create six contexts for this study: two categories for document length (long and short) and three categories for frequency of query terms (low, medium and high). Table 6.1 shows document length and query terms counts for six contexts. The confusion matrices for each topic are trained over crowd worker assessments for other topics (leave-one-out setup over topics).

Contextualized confusion matrices are useful for identifying which questions are easier or difficult for crowd workers and whether we should ask a crowd worker
about a document with a particular context. For example, crowd workers are good at identifying non-relevant documents that are long and have low query terms, whereas they are not good at identifying relevant document that have low query terms and short length [104]. Table 6.2 shows confusion matrix for context 1 “low number of query terms and short document length”. Rows represent true grades assigned by expert assessors, whereas columns represent grades assigned by crowd workers. Each cell in this confusion matrix contains the conditional probability $p(A_k = a_k \mid L = l_i)$ in Equation 6.2. For example in Table 6.2 the conditional probability that a document in this context with true grade 0 will be labelled as grade 0 by a crowd worker $p(A_k = 0 \mid L = 0)$ is 0.75, whereas the conditional probability that a document in this context with true grade 0 will be labelled as grade 3 by a crowd worker $p(A_k = 3 \mid L = 0)$ is 0.02. These probabilities show that in this context there is a high chance that a crowd worker will assess a non-relevant document correctly. The rows are normalized and probabilities in a row add to 1. This confusion matrix is created from training over expert and crowd workers assessments for all topics other than the topic of interest (in this example topic 221) similar to a Machine Learning leave-one-topic-out setup. For example if we want to create confusion matrices for topic 221, then the training data will consist of documents from all topics except 221. These documents will be partitioned into different groups based on their features (document length and number of query terms). For our six contexts defined in Table 6.2 the training documents for topic 221 will be partitioned into six groups. A separate confusion matrix is created from training documents in each group. For each topic there will be six confusion matrices, for total nine topics there will be 54 confusion matrices.

A hierarchal model can also be used for building contexts of confusion matrices. We can learn these contexts from training data using decision tree. We can use following steps to create hierarchal confusion matrices:

1. Measure entropy of the confusion matrix (high entropy means high uncertainty) of training data

2. If the entropy is high then split the training data based on some document or assessor features.

3. Select the threshold value for splitting that creates confusion matrices with lowest entropy

4. Keep splitting training data until entropy of confusion matrices is very low.
Table 6.1: Contexts for Confusion Matrices

<table>
<thead>
<tr>
<th>Doc Length</th>
<th>Q Terms</th>
<th>&lt; 3</th>
<th>3 to 6</th>
<th>≥ 6</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt; 400 words</td>
<td>Context 1</td>
<td>Context 2</td>
<td>Context 3</td>
<td></td>
</tr>
<tr>
<td>≥ 400 words</td>
<td>Context 4</td>
<td>Context 5</td>
<td>Context 6</td>
<td></td>
</tr>
</tbody>
</table>

Table 6.2: Confusion Matrix of conditional probabilities (per row) for Topic 221 and Context 1 “low number of query terms and short document length”

<table>
<thead>
<tr>
<th>NIST Assessor</th>
<th>CrowdWorker</th>
<th>Grade 0</th>
<th>Grade 1</th>
<th>Grade 2</th>
<th>Grade 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Grade 0</td>
<td>0.75</td>
<td>0.18</td>
<td>0.05</td>
<td>0.02</td>
<td></td>
</tr>
<tr>
<td>Grade 1</td>
<td>0.46</td>
<td>0.4</td>
<td>0.11</td>
<td>0.3</td>
<td></td>
</tr>
<tr>
<td>Grade 2</td>
<td>0.13</td>
<td>0.53</td>
<td>0.17</td>
<td>0.17</td>
<td></td>
</tr>
<tr>
<td>Grade 3</td>
<td>0.3</td>
<td>0.2</td>
<td>0.2</td>
<td>0.3</td>
<td></td>
</tr>
</tbody>
</table>

We have used above method for creation of hierarchal confusion matrices but quality of these confusion matrices was not better than quality of confusion matrices created from fixed contexts given in Table 6.1. We believe that the reason is not failure of hierarchal model but insufficient training data.

6.3.3 Prior of Relevance Grades

We calculate prior $p(L)$ for calculation of posterior distribution over relevance grades in Equation 6.2 using BM25 normalized scores divided in 10 buckets, where each bucket has equal number of documents. For each document of a topic, $p(L)$ corresponds to the probability of $L$ for all documents in the same bucket for all training topics (leave-one-out setup over topics). We are aware that other researches might use much better and more topic-particular priors; any such prior can be easily plugged in the framework. Our focus in this paper is not on finding a particularly good prior.

6.3.4 Off-line Collection of Expert Assessments

One fundamental design criterion for online assessments (active learning) is the ability to select examples in real time. In a crowdsourcing scenario, it is typically difficult to have available “online” both crowd workers and experts at the same time. Our framework will not encounter this problem for the following reason: the variance of document for which the framework decides to ask an expert drops to zero, since the confusion matrices for experts are perfect in that $p(a \mid l)$ is 1 if $p$ is equal to $l$ and 0 otherwise. Thus a document is never selected again for assessment, once it has been assessed by an expert. (This is not literally correct since the experts
Table 6.3: Confusion Matrix of conditional probabilities (per row) for Topic 221 and
Context 1 “low number of query terms and short document length” (Preferences)

<table>
<thead>
<tr>
<th>NIST Assessor</th>
<th>CrowdWorker</th>
<th>Doc1 Doc2</th>
<th>&gt; Doc1</th>
<th>&lt; Doc1</th>
<th>= Doc1</th>
</tr>
</thead>
<tbody>
<tr>
<td>$G_1 = 0, G_2 = 0$</td>
<td>0.75</td>
<td>0.18</td>
<td>0.05</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$G_1 = 0, G_2 = 1$</td>
<td>0.75</td>
<td>0.18</td>
<td>0.05</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$G_1 = 0, G_2 = 2$</td>
<td>0.75</td>
<td>0.18</td>
<td>0.05</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$G_1 = 0, G_2 = 3$</td>
<td>0.75</td>
<td>0.18</td>
<td>0.05</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$G_1 = 1, G_2 = 0$</td>
<td>0.75</td>
<td>0.18</td>
<td>0.05</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$G_1 = 1, G_2 = 1$</td>
<td>0.75</td>
<td>0.18</td>
<td>0.05</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$G_1 = 1, G_2 = 2$</td>
<td>0.75</td>
<td>0.18</td>
<td>0.05</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$G_1 = 1, G_2 = 3$</td>
<td>0.75</td>
<td>0.18</td>
<td>0.05</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$G_1 = 2, G_2 = 0$</td>
<td>0.75</td>
<td>0.18</td>
<td>0.05</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$G_1 = 2, G_2 = 1$</td>
<td>0.75</td>
<td>0.18</td>
<td>0.05</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$G_1 = 2, G_2 = 2$</td>
<td>0.75</td>
<td>0.18</td>
<td>0.05</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$G_1 = 2, G_2 = 3$</td>
<td>0.75</td>
<td>0.18</td>
<td>0.05</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$G_1 = 3, G_2 = 0$</td>
<td>0.75</td>
<td>0.18</td>
<td>0.05</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$G_1 = 3, G_2 = 1$</td>
<td>0.75</td>
<td>0.18</td>
<td>0.05</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$G_1 = 3, G_2 = 2$</td>
<td>0.75</td>
<td>0.18</td>
<td>0.05</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$G_1 = 3, G_2 = 3$</td>
<td>0.75</td>
<td>0.18</td>
<td>0.05</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

do have an error of about 10% [112], but it is practically reasonable for most tasks
involving crowd workers.) So the expert judgments can be postponed to a later
batch process, and the corresponding documents that need expert assessment can
be removed from the pool. Expert assessments can be collected in “online” fashion
or off line at the end, the framework does not require any expert assessments to be
collected “online” and it can postpone actual collection of expert assessments till
the end.

6.3.5 Baselines

We have compared our framework with following three baselines:

**Crowd Workers Baseline**

This baseline method only uses crowd workers as assessors. Assessments are col-
lected in round robin fashion. In every round 1 judgment is collected for each
document.

**Experts Baseline**

This baseline method only uses experts as assessors. It selects documents for as-
essment according to their BM25 scores in descending order.
Combination Baseline

It is a naive method for combining assessments from crowd workers and experts: collect all crowd worker assessments in the beginning and sort documents by their average grade; then expert assessments are obtained for documents in this rank list in descending order.

6.4 Results

This section discusses results from our experiments. In all plots x-axis shows total cost at any given point spent on all topics. The total money on x-axis is divided among topics according to proportion of documents for each topic such that topics with more documents get more money. As an implementation detail, to avoid degenerated cases, we start all crowd workers runs with 3 crowd assessments per document for each query, overall a cost of about $200 (thus the X axis starts at $200).

6.4.1 Quality of Rank List

Our Framework maintains a “current” posterior distribution over grades for each document. We calculate expected grade from this distribution and rank documents according to their expected grades. To measure directly the quality of these expected grades (without using them for system evaluation) we treat the list as a retrieval ranking and measure its mean Graded Average Precision (GAP) [113]. “GAP generalizes average precision to the case of multi-graded relevance and inherits all the desirable characteristics of AP: it has a nice probabilistic interpretation, it approximates the area under a graded precision-recall curve and it can be justified in terms of a simple but moderately plausible user model” [113]. A high GAP indicates that the order is similar to the order given by the published TREC qrel grades.

Comparison to Baseline Using Only One Assessor Type

Figure 6.3 shows GAP score for our framework labelled as 'Framework’ and the two baselines ‘expertBaseline’ and ‘crowdWorkerBaseline’. The ‘crowdWorkerBaseline’ curve stops very early since we have collected 10 assessments for each document and the total cost for collection of 10 judgments per document on all topics was approximately $600. The curves for ‘crowdWorkerBaseline’ and ‘expertBaseline’ show that we get higher quality of assessments at same cost using crowd workers in the beginning (as compared to experts), but after spending around $300 on crowd workers, getting more crowd worker assessments does not improve the quality of rank list. Our framework for combining crowd worker with experts has a better GAP score, at all cost points, as compared to the two baselines. The curve
‘Framework’ is significantly\(^3\) better than ‘expertBaseline’ between cost of $400 and $900.

**Comparison to Baseline Using Two Assessor Types**

Figure 6.4 shows GAP score for our framework labelled as ‘Framework’ and the baseline labelled as ‘combinationBaseline’ that uses both assessor types (described in Section 6.3.5). Again, the framework has better GAP score at all costs as compared to this baseline. The curve ‘Framework’ is significantly\(^4\) better than ‘combinationBaseline’ between cost of $400 and $900. The curve ‘combinationBaseline’ has very small improvement between costs of $300 and $600 whereas the curve ‘Framework’ continues to improve GAP score. The reason behind this improvement is that the framework figures out which documents should be sent to experts (since crowd workers cannot label these documents accurately) and does not further spend money on crowd worker assessments for these documents. These results demonstrate that the framework is not only improving over baselines using only one type of assessor, but it also intelligently selects and combines the assessments from the two assessor types.

**6.4.2 System Evaluations**

For IR system evaluations, we need to convert the rank list of documents into a QREL (Query Relevance) with absolute grades; once we have a QREL we can use standard TREC measures to score participant TREC IR systems and compare with published scores. To obtain a baseline QREL, we round average grades from baselines into absolute integer grades. For the framework QREL, we use the posterior distribution of grades (at a given cost point) as follows: We calculate \(R_i\) (number of documents labelled with average grade \(i\)) from sum of \(\sum_d p_i\) (where \(p_i\) is current posterior distribution of relevance grade \(i\), and \(N\) is total number of documents for the topic) for all documents \(d\) at a given cost. We label top \(R_i\) documents in the rank list sorted from expected grades with grade \(i\) in descending order of \(i\).

We have used \(\tau_{AP}\) (APCorr) \(^{[15]}\), Kendall’s \(\tau\) and Root Mean Squared Error (RMSE) \(^{[36]}\) for evaluation of our QREL. These three measures were used by TREC 2013 Crowdsourcing track \(^{[36]}\) for evaluation of participant runs. \(\tau_{AP}\) and Kendall’s \(\tau\) show ranking correlation between the ranking of participant runs (induced by the mean ERR@20 \(^{[114]}\) for all topics with the estimated QREL) against the published runs ranking.

---

\(^3\)Statistical significance is determined using a two-tailed T-Test and is measured at a significance level of 0.05.

\(^4\)Statistical significance is determined using a two-tailed T-Test and is measured at a significance level of 0.05.
Figure 6.3: Graded Average Precision: Comparison of framework with baselines using only one assessor (expert or crowd worker).

Figure 6.4: Graded Average Precision: Comparison of framework with baseline method for combining assessments of experts and crowd workers.
Comparison to Baseline Using Only One Assessor Type

Figure 6.5 shows $\tau_{AP}$ score for our framework labelled as 'Framework' and the two baselines 'expertBaseline' and 'crowdWorkerBaseline'. Our framework has better $\tau_{AP}$ score at all costs as compared to the two baselines.

Figure 6.7 shows RMSE score for our framework labelled as 'Framework' and the two baselines 'expertBaseline' and 'crowdWorkerBaseline'. This figure also shows that our framework has lower RMSE as compared to the two baselines.

Comparison to Baseline Using Two Assessor Types

Figure 6.6 shows $\tau_{AP}$ score for our framework labelled as 'Framework' and the baseline labelled as 'combinationBaseline' that uses both assessor types (described in Section 6.3.5). Our framework has better $\tau_{AP}$ score at all costs as compared to this baseline. The plot 'combinationBaseline' has very small improvement between costs of $300 and $600 whereas the plot 'Framework' continues to improve $\tau_{AP}$ score. There is a quick rise in the baseline between cost of $640 to $820. The reason for this jump is that at this point the all crowd worker assessments are exhausted and the baseline starts to use expert assessments. The baseline starts to fix the rank list produced from crowd workers assessments at this point from top to bottom. The documents at top are more important as compared to documents at bottom for $\tau_{AP}$ evaluation measure so the high quality assessments for top documents from experts quickly improve the $\tau_{AP}$ score.

Figure 6.8 shows RMSE score for our framework labelled as 'Framework' and the baseline labelled as 'combinationBaseline' that uses both assessor types. This figure also shows that our framework has lower RMSE as compared to the baseline.

6.4.3 Analysis

The above figures show that the framework performs significantly better as compared to the baselines. In order to verify that this improvement is due to selection of right type of assessor for right type of documents, we have run the following experiment. Figure 6.9 shows difference between true grade and expected grade from crowd worker assessments at any given cost. It compares two sets of documents, first set of documents consists of documents that have been selected by the framework for assessing by experts ('Documents Selected for Expert Assessments') where as second set of documents consists of documents that have not been selected by framework for expert assessments ('Documents Not Selected for Expert Assessments') by that point in the X axis. As the figure shows, indeed the framework selects those documents for sending to experts that have an higher average error (absolute difference between expected and true grade) on expected grade.
Figure 6.5: System Evaluations (Tau AP): Comparison of framework with baselines using only one assessor (expert or crowd worker).

Figure 6.6: System Evaluations (Tau AP): Comparison of framework with baseline method for combining assessments of experts and crowd workers.
**Figure 6.7:** Root Mean Squared Error (RMSE): Comparison of framework with baselines using only one assessor (expert or crowd worker).

**Figure 6.8:** Root Mean Squared Error (RMSE): Comparison of framework with baseline method for combining assessments of experts and crowd workers.
from crowd worker’s assessments. It should be noted here that the error for both sets of documents is calculated only from crowd worker’s assessments for these documents (expert assessments are not used in error calculation).

Similarly Figure 6.10 shows average variance (instead of grade difference) in crowd worker assessments at any given cost. It compares two sets of documents, first set of documents consists of documents that have been selected by the framework for sending to experts (‘Documents Selected for Expert Assessments’) whereas second set of documents consists of documents that have not been selected by framework for expert assessments (‘Documents Not Selected for Expert Assessments’) – by that point in the X axis. The gain function of our framework is defined in terms of variance in assessments so the framework should select those documents for sending to experts that have higher variance in crowd worker’s assessments. For this reason it is not surprising that Figure 6.10 shows that the framework is doing the right thing and ‘Documents Selected for Expert Assessments’ set has higher average variance in crowd worker’s assessments as compared to ‘Documents Not Selected for Expert Assessments’.
6.5 Framework for Preferences

The probabilistic framework presented in this chapter has been shown to work well for nominal assessments. In order to test if our framework also works with preference judgments we have done some preliminary experiments. In this following we explain how we modify framework for handling preference judgments. Suppose that the crowd workers make preference judgments for some pairs and the end goal is to predict the true label of the documents from these preference judgments. In addition to the posterior distribution of true labels of documents, we also keep joint distribution of true grades for every possible preference pair of documents. The individual posterior distribution of true grade of each document in the pair can be calculated from the marginals of this joint distribution of pairs. Once we get a preference judgment from a crowd worker, we update the joint distribution of the two documents in the pair, individual posterior distribution of true grade of each document in the pair is updated from the new joint distribution. Once we get new posterior distribution for true grade of each document, we can use the same method of calculating variance in gain and variance reduction for each document. The variance reduction from a new preference judgment is calculated as average

Figure 6.10: Comparison of average variance in crowd worker assessments between documents selected for sending to experts and documents not selected for sending to experts. Lower average indicates better quality from crowdsourcing alone, thus less benefit from expert assessment.
of variance reduction from two documents in the pair. The preference pairs will be selected by the framework based on expected reduction in variance per unit of cost. We can calculate

- the joint posterior probability distribution (giving joint probability of true grades of both documents in a pair)

\[
p(L_1 = l_i, L_2 = l_j \mid Pref_1 = pref_1, \ldots, Pref_n = pref_n) = \frac{p(L_1 = l_i) \cdot p(L_2 = l_j) \cdot \prod_{k=1}^{n} p(Pref_k = pref_k \mid L_1 = l_i, L_2 = l_j)}{\sum_{l_{m}} \sum_{l_{n}} p(L_1 = l_{m}) \cdot p(L_2 = l_{n}) \cdot \prod_{k=1}^{n} p(Pref_k = pref_k \mid L_1 = l_{m}, L_2 = l_{n})}.
\]

- \(p_{d_1}^{n,pref}\) and \(p_{d_2}^{n,pref}\), the posterior label distributions for the two documents \(d_1\) and \(d_2\) in a pair, given a particular crowd worker preference \(pref\) between the two documents \(d_1\) and \(d_2\)

\[
p^{n,pref}_{d_1} = p(L_1 = l_i) = \sum_{l_j} p(L_1 = l_i, L_2 = l_j \mid Pref_1 = pref_1, \ldots, Pref_n = pref_n)
\]

\[
p^{n,pref}_{d_2} = p(L_2 = l_j) = \sum_{l_i} p(L_1 = l_i, L_2 = l_j \mid Pref_1 = pref_1, \ldots, Pref_n = pref_n)
\]

- the posterior variance in our gain, based on \(p^{n,pref}_d\) which assumes preference \(pref\)

\[
Var_{p_{d}^{n,pref}}[G] = E_{p_{d}^{n,pref}}[G^2] - E^2_{p_{d}^{n,pref}}[G],
\]

- the resulting variance reduction due to preference \(pref\), i.e., the difference between the current and new variance, given preference \(pref\)

\[
VR(Pref_{n+1} = pref) = \frac{(Var_{p_{d_1}^{n,pref}}[G] - Var_{p_{d_1}^{n,pref}}[G]) + (Var_{p_{d_2}^{n,pref}}[G] - Var_{p_{d_2}^{n,pref}}[G])}{2},
\]

- \(p(pref)\), the probability of obtaining preference \(pref\) from the particular crowd worker, given our current belief in the true labels

The probability \(p(pref)\) of obtaining a particular preference \(pref\) is dependent on two factors: (1) the probability that the true labels are some particular
value \( l_1, l_2 \) and (2) the conditional probability that the assessor returns \( \text{pref} \), given that the true labels are \( l_1, l_2 \). Our current estimate of (1) is given by \( p_{d_1}, p_{d_2} \), and (2) can be obtained from the confusion matrix associated with the particular assessor.

\[
p(\text{pref}) = \sum_{l_1} \sum_{l_2} p_{d_1}^{n}(l_1) \cdot p_{d_2}^{n}(l_2) \cdot p(\text{pref} \mid l_1, l_2)
\]

- the expected variance reduction

\[
E[VR] = \sum_{\text{pref}} p(\text{pref}) \cdot VR(\text{Pref}_{n+1} = \text{pref}).
\]

We can calculate the expected variance reduction for all combinations of documents and assessors, and in combination with the assessor costs, we can calculate all expected price-performance ratios, i.e., the variance reductions obtained per unit cost. For every document-assessor pair, we calculate the value of the following objective function (expected reduction in variance divided by cost of assessor). Cost of preference pair is 1.5 times the cost of a nominal assessment for both crowd workers and experts.

\[
\text{ObjectiveFunction} = \frac{E[VR]}{\text{Cost}_{\text{Assessor}}}
\]

### 6.5.1 Combining Preferences and Nominal Assessments

We have shown that our framework can be used to combine assessments from different types of assessors for both preferences and nominal judgments. We can also use our probabilistic framework for combining preferences and nominal assessments. We calculate expected reduction in variance for all possible pairs and documents and then use the framework to pick the best pair or document for assessments based on maximum variance reduction per unit of cost.

### 6.5.2 Experiments

For our preliminary experiments we have inferred preferences from nominal judgments. If a crowd worker has labelled document 1 with higher grade as compared to document 2 then we infer that the crowd worker has preferred document 1 over document 2. Similarly documents that are assigned same grade by a crowd worker are considered ties.
**Pair Selection**

One of the major obstacles in use of preference judgments is quadratic number of preferences pairs ($\binom{n}{2}$ pairs for n documents). Information retrieval data collections are often skewed with large number of non-relevant documents so our goal is to select the pairs with maximum number of relevant documents. We are not interested in relative order between any non-relevant documents. In order to avoid pairs with 2 non-relevant documents we select $10n$ pairs from $\binom{n}{2}$ pairs in the following way. We initialize each document’s prior probability distribution over true grades using the prior described in Section 6.3.3. In next step we calculate expected variance reduction for each document pair using the modified framework for preferences. We rank all pairs based on expected variance reduction in descending order, and select top 10 pairs for each document from this rank list. We run the framework on these selected pairs.

**Baselines**

There is no well known or standard method of selecting $O(n)$ pairs for preference judgments, so we decided to use the framework for pair selection for the baselines. We call these plots baselines since they use the framework with only one assessor type.

**Results**

Figure 6.11 shows GAP score for our framework labelled as ‘ExpertAndCrowdWorker’, and three baselines ‘Expert’, ‘CrowdWorker’, and ‘Baseline’. All of these plots use preference judgments and all of these plots use framework, the difference between these plots is assessor type. The plot ‘ExpertAndCrowdWorker’ combines crowd workers and experts using our Bayesian framework whereas the plots ‘Expert’, and ‘CrowdWorker’ use the framework with only one assessor type. The plot ‘Baseline’ uses the framework with crowd workers till we exhaust all judgments from crowd workers and then it uses experts as assessors. In other words it is a naive way of combining the two assessor types. Our framework for combining crowd workers with experts ‘ExpertAndCrowdWorker’ has a better GAP score, at all cost points, as compared to the three baselines.

Figure 6.12 shows GAP score for our framework (combining both preferences and nominal assessments) labelled as ‘ExpertAndCrowdWorker’, and two baselines ‘Expert’, and ‘CrowdWorker’. All of these plots use both preferences and nominal judgments and all of these plots use framework, the difference between these plots is assessor type. The plot ‘ExpertAndCrowdWorker’ combines crowd workers and experts using our Bayesian framework whereas the plots ‘Expert’, and
Figure 6.11: Graded Average Precision: Comparison of framework (using preferences) with baselines using only one assessor type (expert or crowd worker).

Figure 6.12: Graded Average Precision: Comparison of framework (using preferences and nominal judgments) with baselines using only one assessor type (expert or crowd worker).

‘CrowdWorker’ use the framework with only one assessor type. Our framework for combining crowd workers with experts ‘ExpertAndCrowdWorker’ has a better GAP score, at all cost points, as compared to the two baselines.
6.6 Summary

We have proposed a probabilistic framework for optimally combining assessments from crowd workers and experts in order to obtain high quality assessments cheaply, by asking only the hard questions to expert assessors. Our study shows promise that our approach matches the quality of expert assessments at much lower cost. Further, the techniques underlying our proposed framework are more generally applicable to the problem of optimally leveraging crowd-sourced workers, for important tasks likely quite different from test collection construction.

We believe that we can develop the ability to collect expert-quality relevance assessments at large scale by collecting crowdsourced assessments with careful quality controls and then increasing their accuracy by selecting the most difficult questions to send on to experts. Learning to combine preference pair data with relevance judgments is key for controlling quality, as preference pairs have many qualities that make them easier questions for crowd workers to answer accurately while relevance grades require fewer questions to assess the entire collection. Our preliminary experimental results with preferences show potential of our framework for combining preferences and nominal assessments. Finally, accurately assessing the quality of individual workers is important so we can minimize the amount of work we need experts to perform.
Chapter 7

Conclusion and Future Work

This thesis is about selection, collection and combining of relevance assessments for IR test collection construction. The first step in this study was to analyse the assessments made by crowd workers for two different types of relevance assessments: pair preferences and nominal relevance grades. We have then built a Bayesian framework for combining two different types of relevance assessments: noisy and inaccurate assessments from crowd workers, and accurate assessments from experts. We have also applied a system for calculating chess ratings (Elo rating system [10]) to the problem of combining preferences for IR evaluation. Section 7.1 presents results of our research, Section 7.2 presents evaluation methodology used for obtaining the results, Section 7.3 presents conclusions and discussion on results, and Section 7.4 presents possible future directions for the research.

7.1 Results of the research

In recent years, there has been a boom in crowdsourcing for various applications including IR evaluation. We have investigated utility of crowdsourcing for collection of relevance assessments (preferences and nominal) especially when combined with expert assessments. We present main results of our research in the following:

7.1.1 Analysis of Crowd Worker’s Assessments

We have analysed factors influencing crowd worker’s assessments both for preferences and nominal judgments (relevance grades). We have found the following:

1. Presence of query terms in the title of a document influences whether workers preferred it for both types of assessments (pair preferences and relevance grades).

2. There is no significant correlation between worker accuracy and presence of
query terms in the body of a document for relevance grades but for preferences there is a clear trend: when the relevant document has a higher number of query terms than the non-relevant document, the assessments are more reliable.

3. Long documents are preferred over short documents for pair preferences.

4. Worker assessments are more reliable for longer documents in general for relevance grades.

5. Workers who spent more time reading documents gave more accurate answers. This was found to be true for both nominal judgments and preferences.

6. It is easier to identity a relevant document when it is compared to a non-relevant document as compared to when it is presented on its own.

7. Recall was much higher than precision, across both relevance grades and pair preferences.

### 7.1.2 Combining Preferences to Get A Rank List

In this thesis, we have shown how the Elo rating system [10] can be used to combine a linear number of preferences to obtain either an ordered list of documents or document relevance scores. For these experiments, we have used data from TREC 2012 Crowdsourcing track. The results of our experiments are given below:

- Our Elo rating system and mean grades using EM [47] exhibit higher than the median accuracy on results for all but one topic on submitted runs to TREC 2012 Crowdsourcing track.

- Accuracy of results improve when we apply our Elo system to preference pair probabilities produced from EM.

- Elo rating on preference pairs produces better results with variance as compared to Elo rating without variance.

- Results for Elo rating on preference pairs improve by using non-uniform prior ratings of documents instead of uniform initial document ratings.

- Learning to rank results using preference pair probabilities for training are roughly equally as good as TREC qrel, whereas results produced form training on graded judgements become superior to TREC qrel as number of iterations increase.
• Quality of rank lists produced from nominal judgements as compared to preference judgements is not better or worse overall. We cannot infer from our results that one type of judgement method is better over the other in general.

7.1.3 Framework for Combining Assessments from Crowd Workers and Experts

In an ideal scenario we should use expensive experts only for difficult questions where crowd workers are more likely to make mistakes and use low cost crowd workers for easier tasks. This requires modeling the quality, cost, and benefit of various types of workers for various types of questions. We have proposed a probabilistic framework combines assessments from crowd workers and experts, and optimally selects questions to be asked of each type of assessor. We got following results for quality of rank list of assessments produced using different methods:

• Our Bayesian framework (that combines relevance assessments from experts and crowd workers) produces higher quality assessments as compared to assessments from only crowd workers, when evaluated at same budget.

• Our Bayesian framework produces higher quality assessments as compared to assessments from only experts when evaluated at same budget.

• Our Bayesian framework produces higher quality assessments as compared to a baseline method for combining assessments from crowd workers and experts when evaluated at same budget.

We got following results for correlation with TREC qrel for TREC 8 system evaluations using different methods of producing assessments (evaluated at same budget):

• Correlation is higher for qrel from our Bayesian framework as compared qrel from only crowd worker’s assessments.

• Correlation is higher for qrel from our Bayesian framework as compared qrel from only expert’s assessments.

• Correlation is higher for qrel from our Bayesian framework as compared qrel from baseline method for combining assessments from crowd workers and experts.

7.2 Evaluation

This section provides evaluation measures used for results presented in previous section.
7.2.1 Analysis of Crowd Worker’s Assessments

Results for analysis of crowd worker’s assessments presented in Section 7.1.1 are based on error rate, per topic. These error rates are the observed probability of a randomly-selected worker making an error of a certain type. We consider a relevance grade to be correct if a worker assigns a grade of 3 or 4 to a relevant document or a grade of 0, 1, or 2 to a non-relevant document, and a pair preference to be correct if a worker prefers a relevant document over a non-relevant document. For consistency reasons, we exclude preferences between two relevant or two non-relevant documents.

7.2.2 Combining Preferences to Get A Rank List

The quality of rank lists produced by combining preferences (using Elo Rating) was evaluated using accuracy (AUC) (defined as area under ROC curve) and Mean Average Precision (MAP). We have used LambdaMART algorithm for evaluation of utility of preferences for learning to rank. LambdaMART takes nominal grades as input and infers probabilities for each possible pair of documents. For preference judgements, we inferred probabilities for each document pair such that document A is more relevant than document B from Elo rating system. The input to Elo rating system is EM probabilities for each judged pair of documents. For graded judgements, we used distribution of grades produced from EM for each pair of documents, and calculated a convolution of probability distributions of two documents to infer the probability that document A has higher grade than document B. We modified LambdaMART algorithm to take pair probabilities as input instead of nominal grades. For comparison we used binary judgements from TREC qrel for training. Mean Average Precision (MAP) has been used for evaluation of results from learning to rank algorithm.

7.2.3 Framework for Combining Assessments from Crowd Workers and Experts

To measure directly the quality of rankings from assessments, we use mean Graded Average Precision (GAP) [113]. A high GAP indicates that the order is similar to the order given by the published TREC qrel grades.

For IR system evaluations, we need to convert the rank list of documents into a QREL (Query Relevance) with absolute grades; once we have a QREL we can use standard TREC measures to score participant TREC IR systems and compare with published scores. To obtain a baseline QREL, we round average grades from baselines into absolute integer grades. For the framework QREL, we use the posterior distribution of grades (at a given cost point) as follows: We calculate $R_i$, the number of
documents labelled with average grade \( i \) from sum of \( N \sum_d p_i \) (where \( p_i \) is current posterior distribution of relevance grade \( i \), and \( N \) is total number of documents for the topic) for all documents \( d \) at a given cost. We label top \( R_i \) documents in the rank list sorted from expected grades with grade \( i \) in descending order.

We have used \( \tau_{AP} \) (APCorr) [15], Kendall’s \( \tau \) and Root Mean Squared Error (RMSE) [36] for evaluation of our QREL. These three measures were used by TREC 2013 Crowdsourcing track [36] for evaluation of participant runs. \( \tau_{AP} \) and Kendall’s \( \tau \) show ranking correlation between the ranking of participant runs (induced by the mean ERR@20 for all topics with the estimated QREL) against the published runs ranking.

### 7.3 Conclusions and Discussion

Our first study was done on analysis and comparison of nominal assessments and preferences from crowd workers. We have concluded from our results that crowd workers have biases toward certain features of documents. They judge documents with high frequency of query words especially in title of documents as more relevant as compared to documents with lower number of query terms. Lengthy documents look more relevant to crowd workers as compared to short documents. Precision of crowd worker’s assessments (nominal vs pairs) show that preferences are not always a better choice as compared to nominal assessments and one has to choose the type of relevance assessment very carefully based on the characteristics of the collection and nature of task. The results also show correlation between time spent on reading documents and worker accuracy. We have learned from this study that we can identify some sloppy workers if we use a minimum time limit on assessments.

Use of preference judgments has been limited due to the polynomial increase in the number of judgments that need to be collected. We have applied the Elo rating system to the problem of combining a linear number of preferences (to obtain either an ordered list of documents or document relevance scores). Although, the results of our experiments are encouraging but there were some limitations in the selection criteria of preference pairs. Due to these weaknesses preference judgements have not shown any significant improvement over nominal judgements. We have done some analysis of our experiments and identified some issues in design of matches for Elo rating system. The details of this analysis are presented in Section 5.5. Our analysis can be helpful for intelligent design of documents matches for Elo rating system.

The Information Retrieval community currently faces a dilemma caused by the
inability of test collections used for evaluation to keep up with the scale of the Internet. Expert relevance assessments are too costly to extensively collect for new, large corpora, and many have looked to crowdsourcing to cheaply replace these expert assessments. However, the poor quality of crowdsourced assessments has so far frustrated these hopes. We have proposed a probabilistic framework for optimally combining assessments from crowd workers and experts to get cost effective assessments. Our framework assess the difficulty of questions using some document features and asks easier questions to crowd workers and hard questions to experts. The results of our experiments were significantly better than baseline methods for combining assessments but there is still room for improvement. We have used only document length and query term frequency for assessing document difficulty for crowd workers but more features can be used for better training. Moreover, our framework does not incorporate any notion of individual worker quality. Modelling worker quality can help divert difficult questions to high quality crowd workers which can save cost of expensive experts.

The end goal of this research is both theoretical and applied, and the potential broader impacts of this research are numerous. Given the rise of Mechanical Turk and the use and utility of such services for solving numerous and varied “human intelligence tasks”, understanding how to optimally leverage disparate categories of workers on the cost-accuracy scale is interesting in its own right, with broad potential applicability and impact. From a more targeted standpoint, the inexpensive yet accurate test collections that would arise would be of great benefit to the IR community.

7.4 Future Work

We are interested in building a machine learning system that can predict the quality of a particular assessment given that it knows about certain features of the object being assessed and the assessor. Our hypothesis is that given large amounts of training data, a machine learning system can learn about the dependencies between various factors affecting the quality of relevance assessment. These factors include various document features (e.g., document length and number of query terms), topic difficulty, assessor demographics, time spent on the assessment, assessor quality on trap questions, and the cost of the assessment. Once we have such a system, it will be easier to identify which questions should be asked to which assessors. This system can be used with active learning to solve three open research problems in relevance assessment tasks: 1) What type of question (preference or nominal) should we ask for a particular topic and a particular document collec-
tion? 2) Which document or pair of documents should be selected for the next question? 3) Which type of assessor (expert or crowd worker) should be selected for the next question? The ability to leverage all types of assessors (expensive and inexpensive), and all types of assessments (preferences and nominal) for test collection construction will be of benefit for IR researchers as well as search engines.
Appendix A

Participation in TREC 2013 Crowdsourcing Track

The goal of the TREC 2013 Crowdsourcing Track [36] was to evaluate approaches to crowdsourcing high quality relevance judgments for web pages and search topics. This appendix describes our submission to Crowdsourcing track.

A.1 Introduction

We have participated in Crowdsourcing Track of TREC 2013 which required collecting relevance judgments for web pages and search topics. This year’s Crowdsourcing Track offered two entry levels for participation, basic and standard. Basic task required relevance assessment of approximately 3,500 documents (a subset of NIST pool, 10 topics), whereas standard task required relevance assessment of approximately 20,000 documents (entire NIST pool, 50 topics). We have participated in basic task of the track. Instead of the usual nominal judgments ‘‘how relevant is this web page?’’, our assessing system used preference judgments ‘‘which one of these two pages is more relevant?’’. Evaluation of adhoc documents works better with preferences for following reasons:

- **reliability.** Traditionally, assessors are asked to give absolute relevance grades to each document with respect to some topic. However, studies have shown that assessors can give more reliable judgments if they are asked which of a pair of documents they prefer, i.e. “is document A better than document B?” [8].

- **consistency.** A much larger agreement among assessors is observed from preference judgments, most likely because assessors do not have to guess or
interpret the given grade scale as they have to do on nominal judgments.

• **training.** Another advantage of using preferences is that many popular learning-to-rank algorithms such as RankBoost [108] and RankNet [109] are trained on preferences (we are not using here such learning algorithm). When preferences are not available, which is often the case, these algorithms need to infer preferences from the absolute relevance judgments collected from assessors; some information is lost during this process, leading to many ties between documents. This suggests that preferences can be used to improve the training of learning-to-rank algorithms that use such a pairwise approach.

The use of preference judgments on document pairs, as opposed to absolute judgments on documents, for IR evaluations creates new challenges. There are \( \binom{n}{2} \) unique pairs of documents for a list of \( n \) documents, which means that the number of judgments we need to collect increases to \( O(n^2) \). Since collecting judgments is costly (even with Mechanical Turks), we need a mechanism for collecting these preferences efficiently.

In the first stage we are approaching the assessing problem as a sorting-the-documents task. Our goal is to minimize the number of judgments needed to sort all documents and find true differences in the performance of retrieval systems. We have implement a QuickSort-like algorithm, using preference judgments (comparisons), so that web documents can be organized into grades following the preferences between each document and preselected special “pivot” documents. Such an algorithm reduces in general the number of judgments needed to fully order a list, as the rate of growth in the number of comparisons is \( O(n \lg n) \), considerably slower than the \( O(n^2) \) growth rate for all comparisons. Our algorithm uses constant number of pivots (given by the grade scale, typically 0-4 integers), so the total number of comparison is reduced to \( O(n) \) in number of documents assessed.

The rest of this appendix is organized as follows: Section A.2 describes our sorting algorithm and collection of preference judgments on document pairs. Section A.3 provides details on our interface design and experimental setup. Section A.4 describes the results of experiments. The final section, Section A.5, concludes this work.

**A.2 Methodology**

In this Crowdsourcing track, participants are actually free to use or not use crowdsourcing techniques however they wish. As discussed in Section 1.1.2, preference judgments are easier for assessors as compared to nominal grades. We have used
a modified version of QuickSort algorithm for sorting documents. This algorithm can be divided into following steps:

1. Select pivot documents such that each pivot belongs to a different relevance class and one pivot is selected from each relevance class.

2. Sort pivot documents

3. For each document, search for the correct position between the sorted pivots.

Each of the above steps is explained in the following sections. Only one assessor (graduate student) was used in these experiments.

A.2.1 Pivot Selection

For pivot selection, we want to sample a subset of documents such that it has at least one document from all possible classes of relevance for a particular topic. Our goal is to minimize the number of documents we need to examine in order to select documents from all possible relevance classes. We sampled documents for pivot selection in the following manner. First, we calculated a prior relevance score for each document using BM25. This produced an initial ranking of the documents for each topic. We sampled every fifth document from this rank list for pivot selection. In addition to every fifth document, we also sampled top 10 documents from the rank list. The motivation behind this strategy is that BM25 rank list has higher ratio of relevant documents at top of list as compared to bottom of list. For pivot selection, our goal is not to select the most relevant documents, but to select at least one document from all relevance classes for a particular topic. (If we only sample documents from top of the list, then there is a high chance that our sample has no document from some relevance class whose documents had very few topic keywords and were ranked very low in the BM25 rank list). Selecting every fifth document from BM25 rank list decreases chances of missing an entire relevance class.

These sampled documents were shown to an assessor who examined all these sampled documents. The assessor selected all the documents from this sample that were even slightly relevant to the topic. These selected documents were then re-examined for selection of small number of pivots. These pivots are selected in a manner such that each pivot is from a different relevance class, for a given topic. The selection of pivots took 1 hour of human effort for each topic.

A.2.2 Linear Search for Each Document

Once pivot documents are selected for a topic, they are sorted based on their relevance to the topic. Each of the remaining documents is compared with pivot docu-
ments to find its “correct spot”. We start by comparing each document to the least
relevant pivot document. If the document is assessed as more relevant than the
pivot document then it is compared to the next least relevant pivot document. If
the document is assessed as less relevant than a pivot document, we stop compar-
isons for that document. One would think binary search is the most effective search
among sorted pivots, but in our case a weighted binary search essentially reduces
a linear search: our prior assumption about relevance of Information Retrieval text
collections is that the ratio of non-relevant documents, then little-relevance, and
so on, is high enough compared to the next grade, such that the distribution over
grades dictates a linear search as the most efficient. All the non-relevant documents
will be compared only once to the least relevant pivot. Since the number of pivots
is fixed, the overall assessment takes $O(n)$ comparisons. Each preference compa-
risons with pivot documents took approximately 30 seconds of human effort, so on
average this phase took 6 hours of human effort for each topic.

A.2.3 Grades from Preference

After comparing all documents with pivot documents, we sorted the documents
using preference judgements. The documents are partitioned into $n + 1$ relevance
classes for $n$ pivots. Each topic-document pair needs to be judged on a six-point
scale for this Crowdsourcing Track. Each of the pivots were manually examined by
an assessor for assigning a relevance grade.
A.3 Experiments

In this section we describe our experimental design for the collection of preference judgements.

A.3.1 Interface Design

The interface we created to collect preference judgements had the following design. After accepting the assignment, assessor was shown instructions. In these instructions, we explained that documents should be preferred strictly based on whether they provide information about the query, description, and narrative for a particular topic: that a well-written discussion of a related topic should not be preferred to a poorly-written document which is exactly on topic.

After dismissing the instructions, the assessor is shown the interface presented in Figure A.1. The “Title” field of a TREC topic is displayed on top, along with its description and narrative. This information describes in detail what constitutes a relevant document for this query. Below the query information is a series of buttons, which allow assessor to record their preferences. Two documents are displayed side-by-side below the buttons. The leftmost and rightmost buttons are labeled “This One,” with an arrow pointing to the left or right document, respectively. These buttons allow assessors to choose a winning document. Between these buttons are two buttons for recording ties, labeled “They’re Equally Good” and “They’re Equally Bad”.

A.3.2 Data

In the basic task of Crowdsourcing track, there were a total of 3,470 documents from ClueWeb12 dataset. ClueWeb12 dataset consists of 733,019,372 English web pages, collected between February 10, 2012 and May 10, 2012 using web crawlers. 10 topics were provided by NIST assessors for this task. Participants of the Crowdsourcing Track were required to simulate the role of NIST assessors for the 10 topics.

A.4 Results

Four groups submitted a total of 11 runs to the Crowdsourcing track this year. Graded judgments of all runs were evaluated against the NIST qrels, using the GAP (Robertson et al [113]) measure. NIST did not judge as much of the pool as anticipated, and thus there are fewer documents judged than expected. A ranking of TREC 2013 Web track adhoc runs (34 adhoc runs) was induced using each of the submitted runs (crowd.qrels). Each crowd.qrel.ranking was compared to
Table A.1: This table shows per-topic statistics and overall averages for the run NEU-Pivot1 and median score for 11 runs submitted to crowdsourcing track. The metrics GAP, ERR@20, AP-correlation and RMSE are listed for each topic. Note that for row all, (i) GAP is the mean gap over all 10 topics, (ii) APCorr and RMSE depend on the ranking of runs induced by the mean ERR20 for all the 10 topics.

<table>
<thead>
<tr>
<th>Topic</th>
<th># Docs</th>
<th>GAP</th>
<th>Median of TREC Runs</th>
<th>τAP</th>
<th>APCorr</th>
<th>Median of TREC Runs</th>
<th>RMSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>202</td>
<td>231</td>
<td>0.007</td>
<td>0.035714</td>
<td>-0.045</td>
<td>-0.01868796</td>
<td>0.157</td>
<td>0.3667390</td>
</tr>
<tr>
<td>214</td>
<td>305</td>
<td>0.797</td>
<td>0.629760</td>
<td>0.325</td>
<td>0.16270939</td>
<td>0.210</td>
<td>0.2100571</td>
</tr>
<tr>
<td>216</td>
<td>387</td>
<td>0.730</td>
<td>0.588345</td>
<td>0.449</td>
<td>0.14264343</td>
<td>0.090</td>
<td>0.2284822</td>
</tr>
<tr>
<td>221</td>
<td>368</td>
<td>0.708</td>
<td>0.535430</td>
<td>0.430</td>
<td>0.08049942</td>
<td>0.183</td>
<td>0.1972541</td>
</tr>
<tr>
<td>227</td>
<td>246</td>
<td>0.606</td>
<td>0.113211</td>
<td>0.642</td>
<td>-0.23982220</td>
<td>0.107</td>
<td>0.4655352</td>
</tr>
<tr>
<td>230</td>
<td>172</td>
<td>0.678</td>
<td>0.272572</td>
<td>0.569</td>
<td>-0.21109738</td>
<td>0.169</td>
<td>0.3311034</td>
</tr>
<tr>
<td>234</td>
<td>298</td>
<td>0.814</td>
<td>0.602812</td>
<td>0.231</td>
<td>0.12199660</td>
<td>0.320</td>
<td>0.2912807</td>
</tr>
<tr>
<td>243</td>
<td>342</td>
<td>0.598</td>
<td>0.254524</td>
<td>0.732</td>
<td>0.38503298</td>
<td>0.158</td>
<td>0.3668166</td>
</tr>
<tr>
<td>246</td>
<td>202</td>
<td>0.428</td>
<td>0.373818</td>
<td>0.048</td>
<td>0.22663554</td>
<td>0.210</td>
<td>0.3438483</td>
</tr>
<tr>
<td>250</td>
<td>207</td>
<td>0.474</td>
<td>0.093525</td>
<td>0.366</td>
<td>-0.18280615</td>
<td>0.107</td>
<td>0.2342819</td>
</tr>
<tr>
<td>All</td>
<td>2758</td>
<td>0.584</td>
<td>NA</td>
<td>0.461</td>
<td>NA</td>
<td>0.085</td>
<td>NA</td>
</tr>
</tbody>
</table>

A.5 Conclusion

In this appendix we have described our work based on preference judgments for obtaining high quality multiple graded relevance judgements. We have used modified version of QuickSort for sorting documents using linear number of preference judgments. The choice of good pivots is essential, to good performance of our sort-
NEUPivot1−vs−qrels.basic (apcorr = 0.4611)

Figure A.2: NEUPivot1-basic-ERR@20 vs qrels.basic-ERR@20. qrels.basic is the TREC 2013 web track qrels reduced to topics 202, 214, 216, 221, 227, 230, 234, 243, 246, and 250.

The indexing algorithm so pivots should be selected by expert assessors. Once we select good pivots, the task of placing documents in right position among pivots is not complex and can be assigned to crowd workers.
Bibliography


115


120


