Explicit Web Search Result Diversification

Rodrygo Luis Teodoro Santos

School of Computing Science

College of Science and Engineering

University of Glasgow

A thesis submitted for the degree of

Doctor of Philosophy

February, 2013

Abstract

Queries submitted to a web search engine are typically short and often ambiguous. With the enormous size of the Web, a misunderstanding of the information need underlying an ambiguous query can misguide the search engine, ultimately leading the user to abandon the originally submitted query. In order to overcome this problem, a sensible approach is to diversify the documents retrieved for the user's query. As a result, the likelihood that at least one of these documents will satisfy the user's actual information need is increased.

In this thesis, we argue that an ambiguous query should be seen as representing not one, but multiple information needs. Based upon this premise, we propose xQuAD—E**x**plicit **Qu**ery **A**spect **D**iversification, a novel probabilistic framework for search result diversification. In particular, the xQuAD framework naturally models several dimensions of the search result diversification problem in a principled yet practical manner. To this end, the framework represents the possible information needs underlying a query as a set of keyword-based *sub-queries*. Moreover, xQuAD accounts for the overall *coverage* of each retrieved document with respect to the identified sub-queries, so as to rank highly diverse documents first. In addition, it accounts for how well each sub-query is covered by the other retrieved documents, so as to promote *novelty*—and hence penalise redundancy—in the ranking. The framework also models the *importance* of each of the identified sub-queries, so as to appropriately cater for the interests of the user population when diversifying the retrieved documents. Finally, since not all queries are equally ambiguous, the xQuAD framework caters for the ambiguity level of different queries, so as to appropriately trade-off *relevance* for *diversity* on a per-query basis.

The xQuAD framework is general and can be used to instantiate several diversification models, including the most prominent models described in the literature. In particular, within xQuAD, each of the aforementioned dimensions of the search result diversification problem can be tackled in a variety of ways. In this thesis, as additional contributions besides the xQuAD framework, we introduce novel machine learning approaches for addressing each of these dimensions. These include a learning to rank approach for identifying effective sub-queries as query suggestions mined from a query log, an intent-aware approach for choosing the ranking models most likely to be effective for estimating the coverage and novelty of multiple documents with respect to a sub-query, and a selective approach for automatically predicting how much to diversify the documents retrieved for each individual query. In addition, we perform the first empirical analysis of the role of novelty as a diversification strategy for web search.

As demonstrated throughout this thesis, the principles underlying the xQuAD framework are general, sound, and effective. In particular, to validate the contributions of this thesis, we thoroughly assess the effectiveness of xQuAD under the standard experimentation paradigm provided by the diversity task of the TREC 2009, 2010, and 2011 Web tracks. The results of this investigation demonstrate the effectiveness of our proposed framework. Indeed, xQuAD attains consistent and significant improvements in comparison to the most effective diversification approaches in the literature, and across a range of experimental conditions, comprising multiple input rankings, multiple sub-query generation and coverage estimation mechanisms, as well as queries with multiple levels of ambiguity. Altogether, these results corroborate the state-of-the-art diversification performance of xQuAD.

Acknowledgements

I would like to express my deepest gratitude to several people for their immense support during the course of my PhD.

First of all, I am extremely grateful to my family for the continued and unconditional encouragement and for helping me manage my life in two different countries over the past four years.

A great deal of gratitude is due to my supervisor, Iadh Ounis. His ambitious and critical view helped shape not only an outstanding thesis, but also my path towards a promising research career. I am also grateful to Craig Macdonald, for co-supervising the development of this thesis, and for collaborating in various other research endeavours.

I must also thank my colleagues at the Terrier team for our collaboration over the past four years: Ben He, Jie Peng, Richard McCreadie, Nut Limsopatham, Dyaa Albakour, and Eugene Kharitonov.

I am also thankful to David Manlove for his courteous and punctual encouragement over the course of my PhD, to Mark Girolami for our fruitful discussions regarding the probabilistic interpretation of the xQuAD framework at an early stage of its development, to the several anonymous reviewers who refereed this work in its various publications, and to Charles Clarke and Leif Azzopardi for their thoughtful feedback during my PhD viva.

Last but foremost, I am specially grateful to my wife-to-be, for her incommensurable support at every single moment of this endeavour, and for shortening over 6,000 miles between us with immense love.

Contents

List of Figures

List of Tables

List of Symbols

Elements

u	A user
q	A query
s	A sub-query
ι	A search intent (e.g., informational, navigational)
d	A document
t	A term

Sets

$\mathcal{Q}$	A set of queries
$\mathcal{L}$	A query log
$\mathcal{S}_q$	A set of sub-queries produced for a query q
$\mathcal{I}_s$	A set of search intents for a sub-query s
$\mathcal{C}$	A corpus of documents
$\mathcal{V}$	A lexicon of terms
Θ	A set of search verticals
$\mathcal{G}_q$	A set of documents relevant for a query q
$\mathcal{R}_q$	A set of documents retrieved for a query q
$\mathcal{D}_q$	A set of documents diversified for a query q
$\mathcal{K}_q$	A set of documents clicked for a query q
$\mathcal{B}_d$	A set of documents linking to a document d
$\mathcal{F}_d$	A set of documents linked to by a document d

Operators

n	The total number of documents in the corpus
n_q	The number of documents retrieved for the query q
n_q^*	The number of documents relevant to the query q

n_t	The number of documents where the term t occurs
v	The number of unique terms in the lexicon $\mathcal{V}$
k	The number of aspects underlying a query
$tf_{t,d}$	The number of occurrences of the term t in the document d
idf_t	The inverse document frequency of the term t
l_ζ	The length (in tokens) of the text ζ
ς_ζ	The length (in bytes or characters) of the text ζ
$w_{t,d}$	The weight of the occurrence of the term t in the document d
f	A function (e.g., a ranking function)
r_d	The rank position of the document d
g_i	The relevance label of the i-th document
h	A learning hypothesis
x	A learning instance
y	A learning label
Δ	A loss function
κ	An evaluation cutoff

Parameters

λ	The diversification trade-off
τ	The diversification cutoff

Chapter 1

Introduction

Search engines have become the primary mechanism for information retrieval (IR) on the World Wide Web. In particular, the leading web search engine has recently reported to be answering a total of 100 billion queries each month, and to be tracking over 30 trillion unique URLs (Cutts, 2012). Nevertheless, the enormous scale at which content is produced and consumed on the Web is not the only challenge faced by current web search engines. An equally challenging task, which is of particular interest to this thesis, is understanding the information needs underlying the queries submitted by web search users (Spärck-Jones et al., 2007).

Queries submitted to a web search engine are typically short (Jansen et al., 2000) and often carry some degree of ambiguity (Song et al., 2009). On the one hand, at least 16% of all queries submitted to a web search engine are genuinely *ambiguous*, in that they allow for multiple *interpretations* of the user's underlying information need to be drawn (Song et al., 2009). For instance, a user issuing the query "*bond*" could mean the financial instrument for debt security, the classical crossover string quartet "Bond", or Ian Fleming's secret agent character "James Bond". On the other hand, even those queries with a single, clearly defined interpretation—and, arguably, every query to some extent—may still be *underspecified*, in that it is not clear which *aspect* of this interpretation the user is actually interested in (Clarke et al., 2008). For example, a user searching for "*james bond*" may be interested to learn about the actors that played the secret agent character in the various films of the series, or when the next film will be released, or simply where to buy the entire film collection.

1. Introduction

The most trivial approach to tackle the ambiguity of a query could be to simply ignore it. Alternatively, a search engine could focus the retrieval process on documents satisfying the most plausible (e.g., the most popular) aspect[1] of the query. In both cases, there is an inherent risk of leaving the user unsatisfied, if none of the retrieved documents matches the actual information need underlying the query. A more diligent approach could be then to ask the users for feedback on what they actually mean (Baeza-Yates et al., 2004). However, it is unreasonable to expect that a user will always be willing to provide such feedback (Hearst, 2009). When a (usually short) query is the only evidence of the user's information need available to the search engine, a more sensible approach is to diversify the documents retrieved for this query (Clarke et al., 2008). By doing so, the search engine can maximise the chance that the user will find at least one of these documents to be relevant to their information need (Chen & Karger, 2006).

Diversifying the search results usually involves a departure from the independent relevance assumption underlying the well-known probability ranking principle in IR (Cooper, 1971; Robertson, 1977). Indeed, it is arguable whether users will still find a given document relevant to their information need once other documents satisfying this need have been observed. Therefore, a search engine should consider not only the relevance of each document, but also how relevant the document is in light of the other retrieved documents (Goffman, 1964). By doing so, the retrieved documents should provide the maximum coverage and minimum redundancy with respect to the aspects underlying a query (Clarke et al., 2008). Ideally, the covered aspects should also reflect their relative importance, as perceived by the user population (Agrawal et al., 2009). In its general form, this is an NP-hard problem (Carterette, 2009). Most previous approaches to this problem deploy a greedy approximation, inspired by the notion of maximal marginal relevance (Carbonell & Goldstein, 1998). In common, they seek to promote diversity by comparing the documents retrieved for a given query to one another, in order to iteratively select those that are the most relevant to the query while being the most dissimilar to the documents already selected. Therefore, these approaches *implicitly* assume that similar documents cover similar aspects of the query, and should hence be demoted, in order to achieve a diversified ranking.

[1] Unless otherwise noted, we will refer to "aspects" and "interpretations" indistinctly.

Alternatively, the broad topic underlying an ambiguous or underspecified query can be usually decomposed into its constituent sub-topics. As a result, we can *explicitly* account for the different aspects of the query, in order to produce a diverse ranking of documents. In this thesis, we introduce a novel framework for search result diversification that exploits such an intuition. In particular, our **Ex**plicit **Qu**ery **A**spect **D**iversification (xQuAD) framework uncovers different aspects underlying the original query in the form of *sub-queries*, and estimates the relevance of the retrieved documents with respect to each identified sub-query. Hence, we can take into account both the variety of aspects covered by a single document, as well as the novelty of this document in face of the aspects already covered by the other retrieved documents. Moreover, the relative importance of each identified sub-query can be directly incorporated within our framework, so as to guide the diversification process towards more plausible aspects of the initial query. This thesis thoroughly evaluates the proposed framework as well as several strategies for instantiating its various components, both analytically as well as empirically. Results using data from the diversity task of the TREC 2009, 2010, and 2011 Web tracks (Clarke et al., 2009a, 2010, 2011b) attest the effectiveness of the proposed framework in contrast to the current state-of-the-art.

1.1 Thesis Statement

The statement of this thesis is that an effective diversification performance can be attained by explicitly representing the multiple possible information needs underlying a query as sub-queries. In particular, by inferring the relative importance of each sub-query, the retrieved documents can better cater for the needs of the user population. Moreover, by maximising the relevance of the retrieved documents with respect to multiple sub-queries, a high coverage of these sub-queries can be achieved. Furthermore, by estimating the relevance of the retrieved documents to already well covered sub-queries, a high novelty can also be attained. Finally, by inferring the level of ambiguity of different queries, a balance between promoting relevance or diversity can be effectively attained.

1.2 Thesis Contributions

The key contributions of this thesis can be summarised as follows:

1. We approach the diversification problem from a user-centric perspective, by explicitly attempting to identify the multiple information needs that may underlie an ambiguous query, based upon past reformulations of this query.

Traditional diversification approaches in the literature exploit intrinsic features of the retrieved documents (e.g., their constituent terms) as surrogates for these documents' coverage of the actual information needs underlying a query. In this thesis, we show that a representation that explicitly aims to model these information needs as sub-queries is more effective. To this end, we exploit query suggestions mined from the query logs of web search engines as sub-queries.

2. We introduce a novel probabilistic framework for search result diversification that is both principled, general, and effective.

The explicit representation of query aspects as sub-queries leads to several ranking criteria that intuitively capture the requirements of the diversification problem, namely, that the search results should have maximum coverage of the possible information needs underlying the query with minimum redundancy, that different information needs may be more or less probable given the query, and that different queries may require different amounts of diversification. We model all these requirements as components of a probabilistic framework, which lays the foundation for a general and effective approach to search result diversification.

3. We thoroughly evaluate all the components of the proposed framework and their impact on the performance of the framework as a whole.

Our thorough experiments validate the aforementioned contributions in comparison to state-of-the-art diversification approaches from the literature. Moreover, we meticulously investigate alternative instantiations for the various components of our proposed framework. As a result, we further contribute effective solutions to the related problem of identifying effective query aspects from a query log, as well as the problem of diversifying the search results in light of query aspects with different intents, or in light of queries with different levels of ambiguity.

1.3 Origins of the Material

Most of the material presented in this thesis has previously appeared in several journal and conference papers published in the course of this PhD programme:

- Chapter 3 describes a taxonomy for diversification approaches in the literature, as initially proposed by Santos et al. (ECIR, 2010e) and later extended by Santos et al. (IRJ, 2012b). A discussion of one of the approaches described in this chapter—orthogonal to the one introduced in this thesis and focusing on efficiency issues—previously appeared in the works by Gil-Costa, Santos, Macdonald & Ounis (SPIRE, 2011) and Gil-Costa, Santos, Macdonald & Ounis (JDA, 2013).

- Chapter 4 provides motivations for a user-centric diversification, as initially advocated by Santos & Ounis (DDR, 2011). In addition, this chapter also identifies the key requirements for an effective diversification performance, as first discussed by Santos et al. (ECIR, 2010e), and describes a probabilistic diversification framework that fulfils these requirements, as originally introduced by Santos et al. (WWW, 2010a).

- Chapter 5 markedly extends the empirical evaluation conducted by Santos et al. (WWW, 2010a), in order to validate the proposed framework in contrast to the current state-of-the-art.

- Chapter 6 extends the work by Santos et al. (IRJ, 2013) on identifying effective query suggestions for an ambiguous query as sub-queries.

- Chapter 7 extends the investigations by Santos et al. (SIGIR, 2011d) on effective estimations of document coverage and novelty.

- Chapter 8 builds upon the simulation analysis conducted by Santos et al. (IRJ, 2012b) on the role of novelty for search result diversification.

- Chapter 9 builds upon the work by Santos et al. (CIKM, 2010b) on diversifying the search results for queries with different levels of ambiguity.

- Chapter 10 includes future directions inspired by Santos et al. (ICTIR, 2011a) on search result diversification across multiple search verticals, such as news, images, and product search, as well as motivations for a unified machine learning approach to explicitly diversify web search results, based upon the findings reported by Santos et al. (SIGIR, 2011e).

During the course of this PhD programme, the approaches introduced in this thesis have also been evaluated in the context of the two major international forums for research on web search result diversification: the Text REtrieval Conference (TREC),[2] run by the US National Institute of Standards and Technology (NIST), and the Workshop on Evaluation of Information Access Technologies (NTCIR),[3] run by the Japanese National Institute of Informatics (NII). The former forum evaluates diversification approaches for English queries (Clarke et al., 2009a, 2010, 2011b, 2012), while the latter is concerned with diversification for the Chinese and Japanese Web (Song et al., 2011a). In addition to the aforementioned publications, some of the approaches introduced in this thesis have been described in the following TREC and NTCIR reports:

- McCreadie, Macdonald, Ounis, Peng & Santos (2009), Santos et al. (2010d), McCreadie, Macdonald, Santos & Ounis (2011), and Limsopatham, McCreadie, Albakour, Macdonald, Santos & Ounis (2012) describe our participations in the diversity task of the TREC 2009-2012 Web tracks.
- Santos et al. (2011f) describe our participation in the NTCIR-9 Intent task.

In our participations in the diversity task of the TREC Web track, the framework proposed in this thesis attained the top performance among the participant groups (best "category B" submission in TREC 2009 and TREC 2010, best overall submission in TREC 2011 and TREC 2012) (Clarke et al., 2009a, 2010, 2011b, 2012), attesting to its effective diversification performance. In our participation in the NTCIR-9 Intent task, our proposed framework ranked second among the participant groups (Song et al., 2011a), showing that the ideas underlying the framework are sound and can generalise effectively to non-English data.

[2] http://trec.nist.gov/
[3] http://research.nii.ac.jp/ntcir/

1.4 Thesis Outline

The remainder of this thesis is organised as follows:

- Chapter 2 describes background material on ranking for IR on the Web, from the basics of a web search engine, to classical approaches for query-dependent and query-independent ranking, to more recent ones that automatically learn an effective ranking model given a set of training queries. The chapter ends with a discussion about retrieval evaluation in IR, laying the foundations for the several experiments conducted in this thesis.

- Chapter 3 begins by describing search result diversification from a historical perspective, as a natural generalisation of relevance-oriented ranking. The diversification problem is then formalised as an optimisation problem, and its computational complexity is analysed. In addition, the chapter organises and describes related approaches to search result diversification. Lastly, the discussion about retrieval evaluation initiated in Chapter 2 is extended to encompass the evaluation of approaches that aim to promote diversity.

- Chapter 4 introduces the xQuAD framework, including its motivation from a user-centric perspective. The framework's optimisation objective is then formalised in probabilistic terms, as a mixture of the probabilities that a retrieved document is relevant to the query and that this document is diverse given the possible information needs underlying the query. The various components that naturally emerge in the formulation of these two probabilities are then described, and an example application of the framework is provided. Lastly, the commonalities and differences between the proposed framework and related approaches from the literature are discussed.

- Chapter 5 is the first of a series of chapters reporting on the experimental evaluation of the xQuAD framework. In this chapter, the experimental methodology that serves as the basis for the experiments in the subsequent chapters of the thesis is also described. The framework is then thoroughly validated in comparison to effective representatives of the various families of diversification approaches in the literature.

- Chapter 6 evaluates the sub-query generation and sub-query importance components of the xQuAD framework. In particular, the chapter introduces a novel machine learning approach for generating effective sub-queries from a limited sample of the query log of a commercial web search engine, in contrast to sub-queries generated by this search engine itself and by a state-of-the-art query suggestion mechanism from the literature.

- Chapter 7 evaluates the document coverage component of xQuAD. To this end, a novel machine learning approach is introduced to leverage the automatically detected intent of each sub-query in order to choose the most effective ranking model to be applied for this sub-query.

- Chapter 8 further evaluates the role played by novelty as a diversification strategy in comparison to and in combination with coverage. In particular, through a simulation analysis, we uncover the limitations of novelty and its role at differentiating between documents with similar coverage.

- Chapter 9 evaluates xQuAD's diversification trade-off component, in order to determine not only when to diversify the search results, but also by how much. To this end, the chapter introduces a supervised approach to automatically adapt the trade-off for queries with different levels of ambiguity.

- Chapter 10 closes this thesis by providing a summary of the contributions and the conclusions made throughout the chapters. Several future directions are then presented, regarding alternative approaches for estimating the several components of the framework, as well as modelling directions for extending the framework for other search scenarios.

Chapter 2

Web Information Retrieval

Information retrieval (IR) deals with the representation, storage, organisation of, and access to information items (Baeza-Yates & Ribeiro-Neto, 2011). The overall goal of an IR system can be stated as to provide items that are relevant to a user's *information need*. In the context of text retrieval, which is the focus of this thesis, information items typically correspond to unstructured or semi-structured *documents*, while information needs are represented as natural language *queries*.

The key challenge faced by an IR system is to determine the *relevance* of a document given a user's query (Goffman, 1964). Since relevance is a prerogative of the user, the IR system can at best estimate it. This task is further aggravated by the fact that both queries and documents are semantically ambiguous expressions of information in natural language. Such an inherent ambiguity precludes a precise match between information needs and items, as would be the case in a *data* retrieval system, such as a relational database (Codd, 1970). In order to be able to effectively answer a user's query, an IR system must be able to first understand the information need underlying this query. In turn, this information need may convey distinct user intents, from a general search for information about a topic, to a search for a particular website (Broder, 2002).

The primary application of interest for this thesis is web search. With this in mind, Section 2.1 describes the basic retrieval process of a web search engine and introduces the main components in this process. Section 2.2 further describes several approaches devoted to ranking documents in a web search setting. Lastly, Section 2.3 describes current approaches for web search evaluation.

2.1 Web Search Engines

Web search engines are arguably the most popular instantiation of an IR system. A recent report revealed that at least 100 billion searches are conducted on the leading commercial web search engine each month, amounting to over 3.3 billion searches each day (Cutts, 2012). Besides understanding the information needs of such a mass of users with varying interests and backgrounds, web search engines must also strive to understand the information available on the Web. In particular, the decentralised nature of content publishing on the Web has led to the formation of an unprecedentedly large repository of information, comprising over 30 trillion uniquely addressable documents (Cutts, 2012). While the lack of a central control is key for the democratisation of the Web, it also results in a substantial heterogeneity of the produced content, from its language and writing style, to its authoritativeness and trustworthiness (Arasu et al., 2001).

Another distinctive characteristic of the Web compared to traditional information repositories is its interconnected nature. Indeed, not only do web authors publish massive amounts of information, but they also create links (also known as *hyperlinks*) between the published information (Berners-Lee, 1989). As a result, the Web can be viewed as a directed graph, with documents represented as nodes, and hyperlinks between documents represented as directed edges (Kleinberg et al., 1999). Understanding the web graph is crucial for understanding the structure and dynamics of the Web itself, but it also plays a fundamental role in designing effective and efficient web search engines (Broder et al., 2000).

The massive-scale, heterogeneous, and interconnected nature of the Web makes it a particularly challenging environment for search (Arasu et al., 2001). To cope with this challenge, web search engines are typically designed with three core components: crawler, indexer, and query processor. Figure 2.1 provides a schematic view of these components. In particular, a *crawler* browses the Web in order to collect documents into a local corpus. This corpus is processed by an *indexer*, which produces data structures for efficient access to the contents of the corpus. The resulting structures are then used by the *query processor*, in order to produce a ranking of documents that are likely to be relevant to a user's query. In the remainder of this section, we briefly describe each of these components.

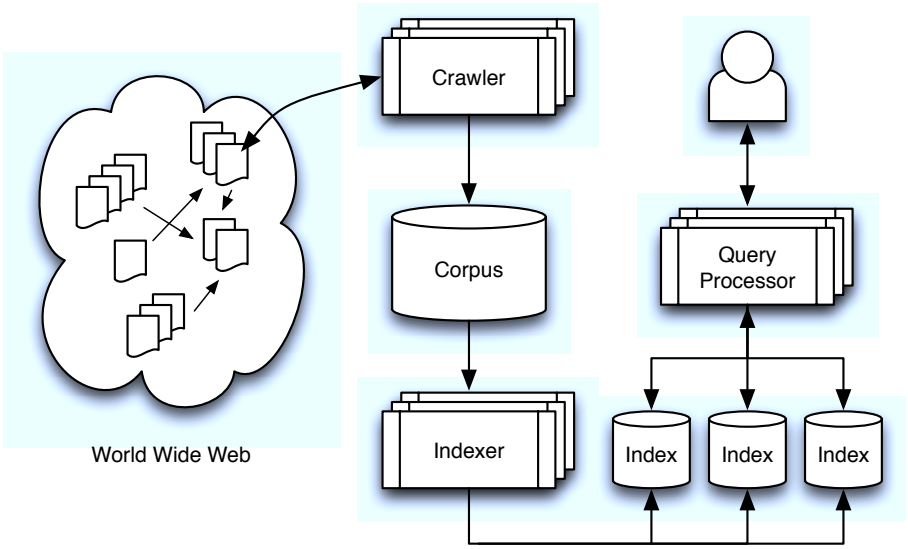


Figure 2.1: Schematic view of a web search engine.

2.1.1 Crawling

Crawling is the process by which search engines collect documents from the Web into a local corpus. Such a corpus can be then processed by the search engine in order to allow users to efficiently locate information. The overall goal of crawling is to build a corpus as comprehensive as possible, in as little time as possible (Pant et al., 2004). To this end, a web crawler must maximise its crawling rate, while making efficient use of its own resources (Castillo, 2004), as well as the resources of the servers that host the desired documents (Thelwall & Stuart, 2006).

Crawling the Web can be seen as a graph traversal problem (Broder et al., 2000). As shown in Figure 2.2, at all times, the crawler maintains a list of URLs to be visited, the so-called *crawling frontier*, which is initially filled with a few seed URLs. While the frontier is not empty, the next URL to be visited is removed from it and downloaded by a *fetcher* module, after a *DNS resolver* translates the URL domain into an IP address. The fetched document is processed by the crawl *controller* and the extracted contents are stored locally for indexing, as will be discussed in Section 2.1.2. The URLs extracted from this document—and the document's own URL, for continuous crawls—are inserted back into the frontier, so that they can be visited by the crawler at a later time (Manning et al., 2008).

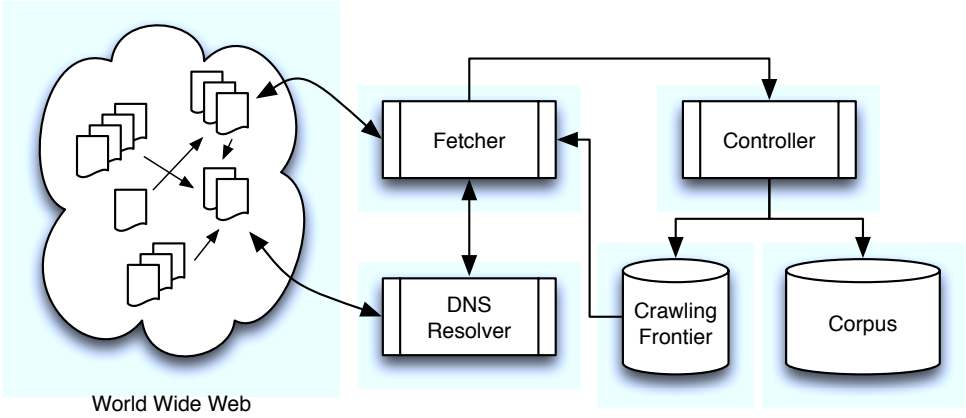


Figure 2.2: Schematic view of a crawler.

Not all content on the Web can be crawled directly. On the one hand, the *surface* Web comprises content that is reachable by following hyperlinks between documents in the web graph. On the other hand, the *deep* Web comprises content that is generated dynamically, typically in response to a user action (e.g., after submitting information through a form, or entering a password-protected area). As a result, the deep Web is orders of magnitude larger than the surface Web (Bergman, 2001),[1] and can only be sampled with special-purpose crawlers (Raghavan & Garcia-Molina, 2000). Nevertheless, the surface Web is itself massive (Cutts, 2012), making crawling a challenging task.

While new documents are created and existing ones are modified at a massive scale, the resources available for crawling—notably, storage and bandwidth—are limited. To make crawling scalable, web crawlers must consider carefully which URLs to visit, and how often to revisit each URL (Castillo, 2004). The decision of which URLs to visit depends on the predicted usefulness of each URL regardless of any particular query. Such a decision could be based on the global importance of the document referred to by the URL or its perceived quality, as will be discussed in Section 2.2.2. However, in practice, it has been shown that a simple breadth-first search is an effective traversal strategy, as it identifies important pages early in the crawling process (Cho et al., 1998; Najork & Wiener, 2001).

[1]Strictly speaking, the deep Web can be infinitely large, as some web applications can generate content indefinitely (e.g., a calendar with "previous" and "next" hyperlinks).

The decision of how often to revisit a particular URL can be even more involved. With the dynamic nature of the Web, by the time a web crawler has finished crawling its frontier, many events could have happened. These events can include the creation, update, or deletion of documents. Moreover, different documents evolve at different rates (Fetterly et al., 2004). For instance, documents related to news, sports, and personal pages tend to change more frequently than those hosted in educational or governmental domains (Adar et al., 2009). At the extreme, recent years have witnessed the emergence of social media, which encourage real-time publishing on collaborative projects, blogs, microblogs, social networking sites, and virtual game worlds (Kaplan & Haenlein, 2010). To provide access to the wealth of information on the Web, a crawler must be able to adapt itself to the publishing patterns of such heterogeneous outlets, e.g., by crawling more often those pages that change more often (Edwards et al., 2001; Ntoulas et al., 2004). As will be discussed in the next section, these considerations are also important for deciding how to efficiently index the crawled content.

2.1.2 Indexing

The overall goal of indexing is to create a representation of the documents in the local corpus suitable for automatic processing by a search engine (Baeza-Yates & Ribeiro-Neto, 2011). The devised document representations are then stored in appropriate data structures for efficient access by the query processor.

Given a corpus of documents (e.g., crawled from the Web), each document is indexed following the general process illustrated in Figure 2.3. Initially, a *parser* extracts the textual content from each document. The extracted content is then processed by a *tokeniser*, which splits the raw text into individual tokens. An *analyser* performs multiple text operations on individual tokens and records their occurrences in each document. In this process, two main data structures are created, which are at the core of modern indexing architectures (Dean, 2009). The first of these is a *lexicon*, which stores information for all unique terms in the corpus, such as their total number of occurrences and the number of documents where they occur. The second structure is an *inverted file*, which stores, for each term in the lexicon, a posting list, comprising information on the occurrence

of the term in different documents, such as the frequency of the term in each document. To enable efficient storage and retrieval, both structures are typically compressed (Witten et al., 1999). Indexing may be performed in a single batch, in which case the whole corpus must be re-indexed when there is an update, or incrementally, through small atomic operations (Peng & Dabek, 2010).

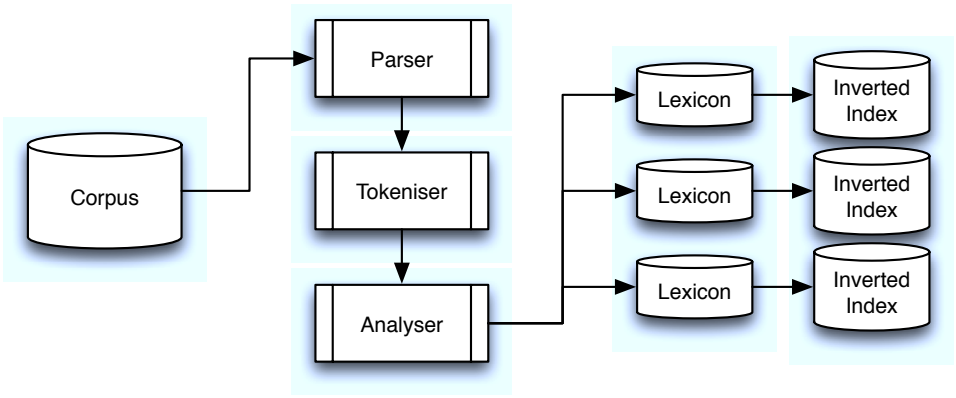


Figure 2.3: Schematic view of an indexer.

Parsing web documents can be a complex task. With the global and democratic nature of the Web, web documents can have a variety of content types and character encodings, which may not be immediately identifiable from the document itself (in an HTML header) or from its provider (in an HTTP response header) (Croft et al., 2009). Even pure textual content may contain noise. Indeed, web documents typically comprise irrelevant content besides their core topic, such as advertisements, client-side scripting code, and frequently a whole HTML template structure. Such a noisy content can hurt not only the effectiveness of a search engine, but also its efficiency, since more content needs to be stored and processed. In order to remove noise and extract cleaner content for indexing, "boilerplate removal" algorithms can be applied (e.g., Vieira et al., 2006; Chakrabarti et al., 2007; Evert, 2008; Kohlschütter et al., 2010).

Tokenisation is a relatively trivial task for most western languages, in which tokens can be separated by a whitespace or a punctuation character. On the other hand, languages such as German do not separate compound words. In the extreme, East Asian languages such as Chinese, Japanese, and Korean have no word boundaries at all. A similar problem, common to all languages, is the segmenta-

tion of queries and URLs (Risvik et al., 2003; Tan & Peng, 2008). An effective approach to this problem is word segmentation based on prior knowledge, by deploying machine-learned sequence models, such as hidden Markov models (Zhang et al., 2003). For East Asian languages, a simple yet effective alternative is to split the textual stream into fixed-length character sequences (typically, two characters long), which can capture the semantics of most individual syllables without having to rely on lexical resources (Manning et al., 2008).

Not all identified tokens are directly useful for search. For this reason, each token can be analysed and transformed through a series of text operations before being indexed. For instance, a search engine can choose not to index too common terms. Such terms, known as *stopwords*, possess little discriminative power for deciding which documents should be retrieved in response to a query. In addition, their presence can also impact efficiency, since their posting lists can be almost as long as the number of documents in the corpus. Besides stopword removal, another common text operation is *stemming*, a process that reduces multiple words to their common grammatical root, so as to increase the chance of retrieving documents that contain a different variant of the query terms (Porter, 1980). For instance, after stemming, the terms "retrieval", "retriever", and "retrieving" can be all reduced to their common root, "retriev". Alternatively, the search engine may choose to index all the identified tokens in their original form, in which case text operations are delayed until the query processing stage. As will be discussed in Section 2.1.3, this choice is more flexible, as it allows for text operations to be deployed only when they are predicted to be helpful (Peng et al., 2007a).

Different information about terms, documents, and the occurrence of terms in documents can be indexed. The most basic information, which is one of the pillars for query-dependent ranking, as will be discussed in Section 2.2.1, is the *frequency* of a term in a document (Luhn, 1957). Recording the *position* where each term occurs in each document can also help improve the effectiveness of a search engine (Zobel & Moffat, 2006). For instance, the terms "information" and "retrieval" appearing next to each other can be a strong indicator of the relevance of a document for the query "information retrieval". In addition, term frequency and positional information can be recorded for different *fields* of a document, such as its title, URL, or body (Zaragoza et al., 2004). Another valuable source

of evidence, which conveys how a document is described by the rest of the Web, is the anchor text of the incoming hyperlinks to this document (Craswell et al., 2001). Finally, several other *features* that can help infer the prior relevance of a document regardless of any query can be computed and stored at indexing time (Das & Jain, 2012). Various such features will be discussed in Section 2.2.2.

2.1.3 Query Processing

Query processing is the component responsible for answering users' queries (Arasu et al., 2001). As illustrated in Figure 2.1, when a user poses a query, the search engine examines its index structures to locate the most relevant documents for this query. Given the size of the Web (Alpert & Hajaj, 2008) and the short length of typical web search queries (Jansen et al., 2000), there may be billions of matching documents for a single query. In order to be effective, a search engine must be able to rank the returned documents, so that the most relevant documents are presented ahead of less relevant ones (Baeza-Yates & Ribeiro-Neto, 2011).

Query processing consists of three basic operations, as illustrated in Figure 2.4. Initially, the search engine receives a query, as a typically short and often under-specified representation of the user's information need (Song et al., 2009). This query may go through a series of *query understanding* operations, aimed to overcome the gap between the user's information need and the ill-defined representation of this need in the form of a query (Li, 2010). This stage is important, since misinterpreting the user's information need implies that relevant documents may never be returned, regardless of how sophisticated the subsequent retrieval is. Once a suitable representation of the user's query has been created, a *matching* process retrieves the indexed documents that contain the query terms. Lastly, to ensure that the user is presented with the most likely relevant documents for the query, the retrieved documents are scored and sorted by a *ranking* process.

Query understanding aims to derive a representation of the user's query that is better suited for a search engine (Li, 2010). Typical query understanding operations include refinements of the original query (Huang & Efthimiadis, 2009), such as spelling correction (Ahmad & Kondrak, 2005; Li et al., 2006), acronym expansion (Jain et al., 2007), stemming (Porter, 1980; Peng et al., 2007a), term

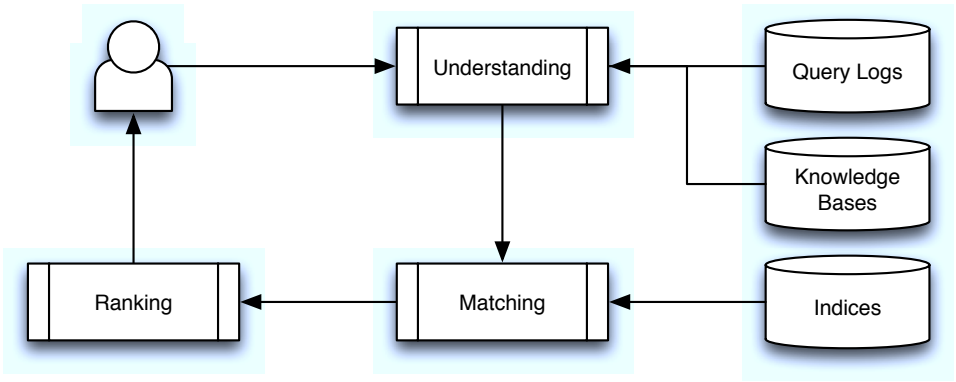


Figure 2.4: Schematic view of a query processor.

deletion (Kumaran & Allan, 2008; Kumaran & Carvalho, 2009), query segmentation (Risvik et al., 2003; Bergsma & Wang, 2007), and named entity recognition (Guo et al., 2009). Other common query understanding operations are query topic classification, aimed to restrict the scope of the retrieved documents (Beitzel et al., 2005; Shen et al., 2006), and query expansion, aimed to enhance the query representation with useful terms from the local corpus (Rocchio, 1971; Lavrenko & Croft, 2001; Zhai & Lafferty, 2001; Carpineto & Romano, 2012), or from external resources, such as a query log (Cui et al., 2002) or a knowledge base such as Wikipedia (He & Ounis, 2007; Li et al., 2007; Xu et al., 2009).

Users typically expect instant responses from a web search engine (Silverstein et al., 1999). This makes it inefficient to fully score all documents matching the query terms. Hence, scoring is typically performed as a multi-layer process (Cambazoglu et al., 2010). In the first layer, matching documents from the entire corpus are returned as an unordered set using a standard boolean retrieval approach (Gudivada et al., 1997). The second layer deploys an unsupervised query-dependent ranking approach, such as those described in Section 2.2.1, in order to provide an overall ordering of the initially matched documents at a low cost. This cost can be made even lower by deploying efficient matching techniques, so as to short-circuit the examination of the posting lists of documents that will not make the final ranked list (e.g., Turtle & Flood, 1995; Macdonald et al., 2012c). Finally, in the third layer, machine-learned ranking can be deployed to integrate ranking evidence from multiple features, as will be discussed in Section 2.2.3.

2.2 Web Search Ranking

The enormous size of the Web most often results in an amount of documents matching a user's query that by far exceeds the very few top ranking positions that the user is normally willing to inspect for relevance (Silverstein et al., 1999). While users may have high expectations regarding the quality of the documents returned by a search engine, they often provide the search engine with a very limited representation of their information need, in the form of a short query (Jansen et al., 2000). In such a challenging environment, effectively ranking the returned documents becomes of utmost importance for satisfying the needs of search users.

Ranking is normally applied on the subset of the indexed documents that matches the user's query, according to, for instance, a boolean retrieval approach, as discussed in Section 2.1.3. A ranking function $f(q, d)$ takes as input a query q, as a representation of the user's need, and a document d, initially matched for this query. As an output, it returns a list $\mathcal{R}_q$ of documents in decreasing order of their estimated relevance to q. Different ranking functions can be thought of as different *features* (or signals) of the estimated relevance of a document to a query. In particular, depending on the evidence it leverages from the query q and the document d, a ranking feature can be categorised into one of three classes:

- *query-dependent document features* score a document according to its estimated relevance to the query;
- *query-independent document features* score the relevance of a document a priori, regardless of any particular query;
- *query features* depend solely on the query, and can be used to adaptively score the relevance of all documents for each individual query.

Query features are addressed in the specific contexts of Chapters 6, 7, and 9. In the remainder of this chapter, Sections 2.2.1 and 2.2.2 introduce several approaches for query-dependent and query-independent ranking, respectively, which are used as document ranking features in various experiments throughout this thesis. In Section 2.2.3, we introduce a machine learning framework for automatically constructing ranking functions that leverage multiple features. The evaluation of the effectiveness of different ranking approaches is discussed in Section 2.3.

2.2.1 Query-dependent Ranking

A standard boolean retrieval is typically insufficient in a web search scenario, and its use is often restricted to producing an initial set of documents that match the query (Cambazoglu et al., 2010; Li & Xu, 2012). From this set, more sophisticated approaches can be deployed to produce a ranking of documents likely to be relevant to the user's information need. To this end, it is of utmost importance that the deployed ranking function be able to appropriately score the occurrences of the query terms in each document. From this perspective, ranking can be seen as the problem of appropriately counting frequencies (Salton & Buckley, 1988).

There are two fundamental frequencies of interest for ranking documents: term frequency and document frequency. The *term frequency* ($tf_{t,d}$) represents the number of occurrences of a term t in a document d, and denotes the importance of the term in the document. The intuition is that a document with more occurrences of a query term is more likely to be relevant to the query (Luhn, 1957). The *document frequency* (n_t) represents the number of documents where the term t occurs in the corpus. This quantity is related to the ability of the term to discriminate between documents. Intuitively, a document that contains a rare query term is more likely to be relevant than a document that contains a common query term (Spärck Jones, 1972). This notion leads to the so-called *inverse document frequency* (idf_t). In its simplest form, given the document frequency, n_t, and the total number of documents in the corpus, n, it can be defined as:

$$idf_t = \log \frac{n}{n_t}. \tag{2.1}$$

A third important quantity for ranking is the *document length* (l_d). This quantity denotes the likelihood that a document will match any query term, regardless of its relevance to the query. As a result, if two documents contain the same number of occurrences of a term, the shorter document should be preferred. Different definitions of document length can be considered (Singhal et al., 1996). A basic working definition is the following:

$$l_d = \sum_{t \in d} tf_{t,d}\,. \tag{2.2}$$

Term frequency, inverse document frequency, and document length are at the heart of the most prominent query-dependent ranking approaches in the literature. These approaches can be broadly categorised as either *algebraic* or *probabilistic*, depending on their underlying mathematical basis (Baeza-Yates & Ribeiro-Neto, 2011). Algebraic approaches represent both the query q and each document d as vectors in the space of all unique terms $t_i \in \mathcal{V}$, such that:

$$\mathbf{q} = (w_{t_1,q}, w_{t_1,q}, \ldots, w_{t_v,q}) \quad \text{and} \quad \mathbf{d} = (w_{t_1,d}, w_{t_1,d}, \ldots, w_{t_v,d}), \tag{2.3}$$

where $w_{t,\bullet}$ is the weight of t in either the query q or the document d, as assigned by a term weighting model, and $v = |\mathcal{V}|$ is the number of unique terms in the lexicon $\mathcal{V}$. The most prominent approach in this family is the vector space model (VSM; Salton et al., 1975), which scores a document vector $\mathbf{d}$ by its similarity to the query vector $\mathbf{q}$, as given by the cosine between $\mathbf{q}$ and $\mathbf{d}$, according to:

$$f_{\text{VSM}}(q,d) = \cos(\mathbf{q},\mathbf{d}) = \frac{\mathbf{q}\cdot\mathbf{d}}{\|\mathbf{q}\|\,\|\mathbf{d}\|} = \frac{\sum_{i=1}^{v} w_{t_i,q}\, w_{t_i,d}}{\sqrt{\sum_{i=1}^{v} w_{t_i,q}^2}\,\sqrt{\sum_{i=1}^{v} w_{t_i,d}^2}}. \tag{2.4}$$

In a classical formulation, the VSM adopts *tf-idf* weights, such that $w_{t,\bullet} = tf_{t,\bullet}\, idf_t$ for both queries and documents (Salton et al., 1975). A simple document length normalisation is automatically performed by dividing the dot product between the query and document vectors by the product of their norms. Alternative formulations have been further investigated by Salton & Buckley (1988). In particular, an unnormalised version of Equation (2.4) with binary weights $w_{t,\bullet}$ leads to the simple yet effective coordination level matching (CLM):

$$f_{\text{CLM}}(q,d) = \mathbf{q}\cdot\mathbf{d} = \sum_{i=1}^{v} w_{t_i,q}\, w_{t_i,d}\,. \tag{2.5}$$

Different from algebraic approaches, probabilistic approaches leverage probability theory to model the relationship between queries and documents. In the following, we describe approaches from the three major families of probabilistic ranking in the literature: probabilistic relevance modelling (Section 2.2.1.1), language modelling (Section 2.2.1.2), and divergence from randomness (Section 2.2.1.3).

2.2.1.1 Probabilistic Relevance Modelling

The literature on probabilistic ranking dates back to 1960, with the seminal work by Maron & Kuhns (1960) on probabilistic indexing and retrieval in a library setting. The field experienced intensive development in the 1970s and 1980s (Cooper, 1971; Harter, 1975a,b; Robertson & Spärck Jones, 1976; Robertson, 1977; Robertson et al., 1981), culminating in some of the most effective ranking functions used by current IR systems (Robertson et al., 1994, 2004; Zaragoza et al., 2004).

Probabilistic relevance modelling explicitly accounts for relevance as an integral part of the ranking process. Although relevance is an unknown variable to a retrieval system, properties of the query and the document may provide probabilistic evidence of the relevance of the document to the information need expressed by the query. The probability of relevance of a given document to a given query is central in the formalisation of the well-known probability ranking principle (PRP) in IR (Cooper, 1971; Robertson, 1977):

> *"If a reference retrieval system's response to each request is a ranking of the documents in the collection in order of decreasing probability of relevance to the user who submitted the request, where the probabilities are estimated as accurately as possible on the basis of whatever data have been made available to the system for this purpose, the overall effectiveness of the system to its user will be the best that is obtainable on the basis of those data."*

The PRP provides a general framework for ranking functions:

$$\begin{aligned} f_{\text{PRP}}(q,d) &= p(\mathcal{G}_q|q,d) \\ &\approx \sum_{t\in q} w_{t,d}, \end{aligned} \tag{2.6}$$

where $\mathcal{G}_q$ is the set of documents relevant to the query q, in which case $p(\mathcal{G}_q|q,d)$ denotes the probability of relevance given the query q and the document d. As an abstract principle, the PRP does not prescribe how the probability of relevance should be estimated. Nonetheless, after a series of order-preserving transformations, its general formulation is typically stated as a summation over individual

term weights $w_{t,d}$ (Robertson & Zaragoza, 2009). This simplification makes the estimation of the probability of relevance tractable, by assuming independence among the query terms conditioned on the observation of relevant (and non-relevant) documents. To estimate the individual term weights, there have been two major directions, depending on whether the presence or the actual frequency of terms in a document is considered. The resulting models, respectively, binary independence and best matching, are discussed next.

Binary Independence Model One of the first instantiations of the PRP was the binary independence model (BIM; Robertson & Spärck Jones, 1976). This model assumes a presence-absence scenario, where absence is the complementary event to presence. Under this assumption, $tf_{t,d}$ is a binary variable, denoting whether or not the term t occurs in the document d. It is further assumed that $tf_{t,d}$ provides evidence of the relevance of the document d for the term t, independently of other terms. The general formulation of the PRP under these particular assumptions leads to the following definition:

$$w_{t,d}^{\text{BIM}} = \log \frac{p(tf_{t,d}\,|\mathcal{G}_q)(1 - p(tf_{t,d}\,|\bar{\mathcal{G}}_q))}{(1 - p(tf_{t,d}\,|\mathcal{G}_q))\, p(tf_{t,d}\,|\bar{\mathcal{G}}_q)}, \tag{2.7}$$

where $\mathcal{G}_q$ is the relevance set for q and $tf_{t,d}$ is either 0 or 1. In the presence of actual relevance data (e.g., from the user's feedback), replacing the probabilities in Equation (2.7) with their equivalent proportions leads to the well-known Robertson / Spärck Jones (RSJ) formula (Robertson & Spärck Jones, 1976):

$$w_{t,d}^{\text{RSJ}} = \log \frac{(n_t^* + 0.5)(n - n^* - n_t + n_t^* + 0.5)}{(n_t - n_t^* + 0.5)(n^* - n_t^* + 0.5)}, \tag{2.8}$$

where n_t is the total number of documents in the corpus that contain the term t, n_t^* is the number of such documents that were judged relevant, and n^* is the total number of documents judged relevant. The introduced factor of 0.5 makes the resulting estimation more robust compared to using a simple ratio (Robertson & Spärck Jones, 1976). In a usual scenario, in the absence of relevance data, $n^* = n_t^* = 0$, in which case the individual term weights in Equation (2.8) closely approximate the *idf* formulation in Equation (2.1).

Best-Matching Models The binary independence model estimates the usefulness of different terms at identifying relevant documents for a given query. Such estimates can be iteratively refined with relevance feedback from the users, resulting in an overall improved model. On the other hand, this model cannot differentiate between documents that contain the same query terms, regardless of the extent to which these documents are about these terms.

To overcome the deficiencies of the BIM, Robertson et al. (1981) introduced a non-binary term frequency component to the framework of probabilistic relevance modelling. In order to adequately model term frequency distributions, Robertson et al. (1981) built upon the notion of *eliteness* proposed by Harter (1975a,b). As conceived by Harter, for each term, there exists a set of documents, known as the elite set, which is assumed to be somehow relevant to the term.[2] As a result, the frequency of a term can be described as a mixture of two Poisson distributions (Poisson, 1837): the first distribution describes the frequency of the term in the elite set, whereas the second describes the term frequency in the non-elite set, comprised by the rest of the documents in the corpus.

These distributional assumptions are at the core of Harter's 2-Poisson model for estimating the probability that a document is relevant to a single term (Harter, 1975a,b). In order to extend Harter's idea of eliteness to multi-term queries, Robertson et al. (1981) initially proposed to model the relationship of the elite sets associated with individual query terms and the relevance set associated with the query. Estimating the various parameters that emerge from this formulation turned out to be intractable, since there was no directly useful evidence for performing this task, primarily because eliteness is a hidden variable.

As an alternative, Robertson et al. (1993) proposed a simple yet effective approximation of the 2-Poisson model, by investigating the model's qualitative behaviour as a function of $tf_{t,d}$, i.e., $w_{t,d}(tf_{t,d})$. In particular, they noted that this function had the following properties (Robertson & Walker, 1994):

$$\text{(a) } w_{t,d}(0) = 0 \qquad \text{(b) } w_{t,d}(tf_{t,d}) \propto tf_{t,d} \qquad \text{(c) } \lim_{tf_{t,d} \to \infty} w_{t,d}(tf_{t,d}) = w_{t,d}^{\text{BIM}} \,. \tag{2.9}$$

[2] Strictly speaking, term frequency is assumed to be dependent on eliteness, which is in turn assumed to be dependent on relevance (Robertson & Zaragoza, 2009, page 352).

The first property follows by design. The second property emphasises the monotonically increasing behaviour of $w_{t,d}$ as a function of $tf_{t,d}$. The last property was denoted *saturation*, and reflects the observation that the contribution of a term to a document cannot exceed an asymptotic limit. This limit corresponds to the weight given by the BIM, as defined in Equation (2.7). A simple parametric function that satisfies all these properties is the following:

$$w_{t,d}^{\text{SATU}} = \frac{tf_{t,d}}{k + tf_{t,d}}, \tag{2.10}$$

where $k > 0$ is the saturation parameter. For high k values, increments in $tf_{t,d}$ continue to contribute to the overall weight, whereas for low k values, this contribution tails off quickly (Robertson & Walker, 1994).

Harter's 2-Poisson model relies on the assumption that all documents have the same (constant) length. While this assumption was arguably plausible in the scenario originally addressed by Harter (1975a,b), where abstracts rather than the full text of documents were considered, it is unlikely to hold in a general text retrieval setting, particularly on the Web (Fetterly et al., 2004). To cope with documents of different lengths, Robertson et al. (1993) proposed the following parametrised length normalisation scheme:

$$w_{t,d}^{\text{NORM}} = (1 - b) + b(l_d \,/\bar{l}), \tag{2.11}$$

where l_d and $\bar{l}$ are the length of document d and the average length of all documents in the corpus, respectively, with the parameter b, $0 \leq b \leq 1$, controlling the strength of the normalisation. In particular, $b = 0$ results in no normalisation, whereas $b = 1$ results in a full length normalisation. A carefully chosen setting can help balance the normalisation, so as to penalise long documents that are verbose without harming those that genuinely include extra relevant content (Robertson & Zaragoza, 2009). Applying this scheme to normalise the *tf* component of Equation (2.10) results in the following saturation function:

$$w_{t,d}^{\text{nSATU}} = \frac{tf_{t,d}}{k\, w_{t,d}^{\text{NORM}} + tf_{t,d}}. \tag{2.12}$$

Finally, by combining the normalised term frequency saturation function of Equation (2.12) with the asymptotic maximum of Equation (2.9), which can be approximated by Equation (2.8), we arrive at the definition of the well-known BM25 ranking function (Robertson et al., 1994):

$$f_{\text{BM25}}(q,d) = \sum_{t \in q} w_{t,d}^{\text{nSATU}} \; w_{t,d}^{\text{RSJ}} \; . \tag{2.13}$$

BM25 is the latest of the original family of best-matching (BM) probabilistic models proposed by Robertson et al. (1993, 1994). Variants of the model, including different correction factors for document length normalisation, as well as with parameters for controlling the term frequency saturation in the query itself, are discussed by Robertson & Zaragoza (2009, page 361).

2.2.1.2 Language Modelling

Language modelling is the task of predicting the next term given a previously observed sequence of terms. This task has been extensively investigated in contexts such as automatic word completion, speech, handwriting and optical character recognition (OCR), spelling correction, and statistical machine translation (Manning & Schütze, 1999). Early developments date back to Markov's work on modelling character sequences in Russian literature (Markov, 1913), as well as Shannon's work on modelling sequences of symbols, which helped lay out some of the basic elements of modern information theory (Shannon, 1948).

A *language model* is a probability distribution over sequences of terms (Manning & Schütze, 1999). Formally, let ζ represent some sample text (e.g., a query, a document, a set of documents). A language model θ_ζ is a function that assigns a probability to a sequence of terms $t_1, \cdots, t_v$ given ζ, such that:

$$\theta_\zeta = p(t_1, \cdots, t_v | \zeta) = \prod_{i=1}^{v} p(t_i | t_1, \cdots, t_{i-1}, \zeta), \tag{2.14}$$

where the right-hand expansion follows from the chain rule.

A language model permits generating sequences of terms following the model's distribution, or estimating the probability that a given sequence is generated by

the model. As apparent from Equation (2.14), language modelling aims to predict the next term given the previously observed terms. However, conditioning this prediction on the entire history of observed terms is often infeasible, as evidence of the occurrence of longer sequences is sparser than that of shorter ones (e.g., the observation of a sequence of terms implies the observation of its subsequences, but the opposite is not necessarily true). To counteract sparsity problems, a typical solution is to limit the history of considered terms to the previous $o-1$. This simplification leads to an ngram language model of order o, i.e., $\theta_\zeta^{(o)}$:

$$\theta_\zeta^{(o)} \approx \prod_{i=1}^{v} p(t_i|t_{i-(o-1)}, \cdots, t_{i-1}, \zeta). \tag{2.15}$$

An ngram language model of order o corresponds to a Markov model (Markov, 1954) of order $o-1$, where future observations (i.e., the next term) depend solely on the present state (i.e., the immediately preceding $o-1$ terms). Typical ngram language models are the unigram ($o = 1$) and bigram ($o = 2$) models, which instantiate Equation (2.15) respectively as follows:

$$\theta_\zeta^{(1)} \approx \prod_{i=1}^{v} p(t_i|\zeta), \tag{2.16}$$

$$\theta_\zeta^{(2)} \approx \prod_{i=1}^{v} p(t_i|t_{i-1}, \zeta). \tag{2.17}$$

Despite its prominent usage in other fields, it was only in the late 1990s that language modelling was introduced as a ranking approach for IR (Ponte & Croft, 1998; Hiemstra, 1998; Berger & Lafferty, 1999; Miller et al., 1999). While probabilistically equivalent to the classical models described in Section 2.2.1.1, the language modelling approaches are fundamentally different from a statistical perspective. In particular, probabilistic relevance modelling constructs a model for relevant (and non-relevant) documents given a query, while language modelling constructs a model for relevant queries given a document (Zhai, 2008). The latter choice allows language modelling approaches to estimate effective ranking models without having to make parametric assumptions regarding the distribution of terms in predefined relevance classes (Ponte & Croft, 1998).

Query Likelihood Departing from an explicit account of relevance, the most basic language modelling approach attempts to model the query generation process (Ponte & Croft, 1998). Starting from the probability $p(d|q)$ of observing a document d given the query q and applying Bayes' rule, we have:

$$p(d|q) = \frac{p(q|d)\, p(d)}{p(q)} \propto p(q|d)\, p(d), \tag{2.18}$$

where the latter expression is obtained by ignoring $p(q)$, which is the same for every document d. The document prior $p(d)$ can be estimated in order to emphasise distinctive characteristics of different documents, such as their authority or quality, as will be discussed in Section 2.2.2. Alternatively, this probability is commonly assumed to be uniformly distributed across all documents, in which case it can also be ignored. After these simplifications, ranking is reduced to the task of estimating the probability $p(q|d)$ of observing the query q given the document d. This model, denoted the *query likelihood model* (QLM), estimates the probability that the query q is generated by the document language model θ_d. Under a unigram assumption, it can be stated as follows:

$$f^{(1)}_{\text{QLM}}(q, d) = \prod_{t \in q} p(t|\theta_d)^{tf_{t,q}}, \tag{2.19}$$

where $p(t|\theta_d)$ denotes the probability of observing the term t given the language model θ_d, and $tf_{t,q}$ denotes the frequency of this term in the query q.

Higher-order ngram language models have been deployed with some success in the literature, as a means to reward the occurrence of the query terms in close proximity. For instance, Song & Croft (1999) proposed to interpolate unigram and bigram language models. Srikanth & Srihari (2002) relaxed the sequential nature of bigrams and exploited unordered term pairs. Gao et al. (2004) extended unigram models to cater for term dependence in both the query and the retrieved documents using identified syntactic structures. Alternatively, Cao et al. (2005) leveraged term relationships derived from a thesaurus. Recently, Lv & Zhai (2009) proposed to build multiple language models for different positions within each document, while Zhao & Yun (2009) proposed to refine the estimation of unigram models based upon the centrality of each query term in a document.

A particularly effective approach to exploit term dependence was proposed by Metzler & Croft (2005). Their approach models term dependence in the language modelling framework via Markov random fields (MRF), an undirected graph structure commonly used to model joint distributions. Within this framework, they proposed to model two types of dependence: *sequential dependence*, capturing relationships between pairs of neighbouring query terms, and *full dependence*, capturing relationships between all pairs of query terms. These two models were linearly interpolated with a unigram model, according to:

$$\begin{aligned} f_{\mathrm{MRF}}(q,d) = \alpha_u & \sum_{t_i \in q} \log p(t_i|\theta_d) \\ + \alpha_s & \sum_{t_i \in q} \sum_{\substack{t_j \in q \\ j=i+1}} \log p(\langle t_i, t_j \rangle_\omega | \theta_d) \\ + \alpha_f & \sum_{t_i \in q} \sum_{\substack{t_j \in q \\ j \neq i}} \log p(\langle t_i, t_j \rangle_\omega | \theta_d), \end{aligned} \tag{2.20}$$

where the parameters α_u, α_s, and α_f control the weights of the unigram, sequential, and full dependence models in the linear combination, respectively, and the parameter ω defines the length (in tokens) of the sliding window for counting occurrences of the pair $\langle t_i, t_j \rangle$ in the document d.

Document Likelihood By modelling the language of documents rather than the query language, traditional language modelling approaches are able to leverage more data for inferring the relevance of a document to a given query. On the other hand, it is unclear how to enhance the query representation for improved retrieval, since the query is assumed to be a random sample of the document language model (Zhai & Lafferty, 2001). To overcome this limitation, one could instantiate the language modelling framework to produce a ranking function orthogonal to the query likelihood model. Analogously to Equation (2.19), under a unigram assumption, we can define a *document likelihood model* (DLM) as:

$$f^{(1)}_{\mathrm{DLM}}(q,d) = \prod_{t \in d} p(t|\theta_q)^{tf_{t,d}}. \tag{2.21}$$

Directly deploying this ranking function would likely be ineffective, given the sparse evidence available for estimating θ_q from the query q alone. Nevertheless, the query language model can be enhanced by leveraging feedback information, either directly from users, in the form of relevance judgements, or automatically, by assuming that the top retrieved documents for the query are relevant. The latter scenario is denoted *pseudo*-relevance feedback (Rocchio, 1971). Effective alternatives for constructing improved query language models include the relevance-based language modelling approach of Lavrenko & Croft (2001), as well as the model-based feedback approach of Zhai & Lafferty (2001).

Unified Likelihood While there exist effective approaches for modelling both the query and the document generation processes, an even more effective approach is to combine both query and document language models in a unified formulation. In particular, Lafferty & Zhai (2001) proposed a risk minimisation approach for document ranking within the language modelling framework. In their approach, the risk of returning documents with a language that does not fit the query language is quantified by the Kullback-Leibler (KL) divergence between the query and document language models, θ_q and θ_d, respectively, according to:

$$\begin{aligned} f_{\mathrm{KL}}(q,d) &= -\mathrm{KL}(\theta_q \| \theta_d) \\ &= -\sum_{t \in q} p(t|\theta_q) \log \frac{p(t|\theta_q)}{p(t|\theta_d)}, \end{aligned} \tag{2.22}$$

where $p(t|\theta_q)$ and $p(t|\theta_d)$ denote the probability of observing the term t given the query and document language models, respectively. This formulation has been shown to be effective across many ranking scenarios, and represents the current state-of-the-art in language modelling for IR (Zhai, 2008).

Language Model Estimation A key issue for the effectiveness of language modelling approaches is the estimation of a language model (Zhai & Lafferty, 2004). Given some text ζ (a query or a document), one of the most simple and widely used mechanisms to estimate the language model $\theta_\zeta = p(t|\zeta)$ is the maximum likelihood estimation (MLE; Fisher, 1922), defined as:

$$p_{\text{MLE}}(t|\zeta) = \frac{tf_{t,\zeta}}{l_\zeta}, \tag{2.23}$$

where $tf_{t,\zeta}$ denotes the raw frequency of the term t in the sample of text ζ, whereas l_ζ denotes the length of this text, measured in tokens.

A central problem when estimating language models is that the majority of the terms in a lexicon typically appear very sparsely in limited text samples such as queries and documents. For example, in a query likelihood scenario, some query terms may not appear at all in a document. If the document language model is estimated as in Equation (2.23), the document will be assigned a zero probability of generating the query, unless it contains all query terms. In addition, even when a query term is present in the document, its associated generation probability tends to be overestimated via maximum likelihood (Manning et al., 2008). To overcome these limitations, an effective approach is to *smooth* the probabilities when estimating a language model (Zhai & Lafferty, 2004).

A simple smoothing approach consists in interpolating a query or document-specific language model with the language model of a large background corpus. Typically, the target document corpus $\mathcal{C}$ is used for this purpose. The resulting model, $p_\alpha(t|\zeta)$, is referred to as a linear interpolation language model (or a language model with Jelinek-Mercer smoothing), and is estimated as follows:

$$p_\alpha(t|\zeta) = \alpha\, p_{\text{MLE}}(t|\zeta) + (1-\alpha)\, p_{\text{MLE}}(t|\mathcal{C}), \tag{2.24}$$

where $0 \leq \alpha \leq 1$ is the interpolation parameter. A particularly effective alternative to linear interpolation is Bayesian smoothing with a Dirichlet prior with parameter μ (Mackay & Peto, 1994), defined according to:

$$p_\mu(t|\zeta) = \frac{tf_{t,\zeta} + \mu\, p_{\text{MLE}}(t|\mathcal{C})}{l_\zeta + \mu}. \tag{2.25}$$

It can be shown that Equation (2.25) is a special case of Equation (2.24), with a length-dependent interpolation parameter, i.e., $\alpha = \mu/(l_\zeta + \mu)$. This observation explains the state-of-the-art performance of Dirichlet smoothing (Zhai, 2008).

2.2.1.3 Divergence from Randomness

A different probabilistic approach to query-dependent ranking is based on the notion of *divergence from randomness* (DFR; Amati, 2003). DFR models build upon the intuition that the more the content of a document diverges from a random distribution, the more informative the document is. Similarly to the best-matching approaches discussed in Section 2.2.1.1, DFR models are inspired by Harter's 2-Poisson model (Harter, 1975a,b), which assumes that the informativeness of a term in a corpus can be inferred by analysing its distribution in different subsets of the corpus. Nonetheless, different from best-matching and other probabilistic relevance models, DFR models have no explicit account of relevance. Instead, these models exploit the statistical distribution of terms in documents, in which they resemble the language modelling approaches described in Section 2.2.1.2. However, different from language models, DFR models are an example of frequentist rather than Bayesian inference models (Amati, 2006).

The relationship between the informativeness of a term and its distribution in a corpus of documents has been recognised early (Damerau, 1965; Bookstein & Swanson, 1974; Harter, 1975a,b). As discussed in Section 2.2.1.1, non-informative terms tend to be randomly distributed over the document corpus, whereas informative terms appear more densely in a few *elite* documents. In particular, the frequency of a non-informative term can be modelled by a Poisson distribution with a mean proportional to the average frequency of the term in the corpus. Under this assumption, inferring the informativeness of a term reduces to measuring the deviation of the term's frequency distribution from a random distribution. Harter's 2-Poisson model and the family of best-matching models derived from it perform this inference by parametrising the occurrence of informative terms as a second Poisson distribution (Harter, 1975a,b). As discussed in Section 2.2.1.1, estimating the parameter of this distribution for each query term is problematic, since eliteness is a hidden variable (Robertson & Zaragoza, 2009).

To overcome this limitation, DFR models assume that the elite set of a term is simply the set of documents that contain the term (Amati & van Rijsbergen, 2002). In particular, the basic hypothesis underlying DFR models is that *"the informative content of a term can be measured by examining how much the term*

frequency distribution departs from a 'benchmark' distribution, that is, the distribution described by a random process" (Amati, 2003). To quantify this hypothesis, a prototypical DFR model can be defined as follows:

$$f_{\text{DFR}}(q, d) = \sum_{t \in q} w_{t,q}\, w_{t,d}, \tag{2.26}$$

where $w_{t,q}$ and $w_{t,d}$ represent the weight of each term t in the query q and in the document d, respectively. The former weight is typically computed as the normalised frequency of t in q, according to:

$$w_{t,q} = \frac{tf_{t,q}}{\max_{t_i \in q} tf_{t_i,q}}. \tag{2.27}$$

In turn, the weight $w_{t,d}$ is computed as:

$$w_{t,d} = inf_1\, inf_2, \tag{2.28}$$

where $inf_1 = -\log_2 p_1(t|\mathcal{C})$ and $inf_2 = 1 - p_2(t|d)$ define the informativeness of the term t in the corpus $\mathcal{C}$ and in a document d that contains t, respectively. As a result, the weight $w_{t,d}$ of each query term t in a document d is a decreasing function of both probabilities $p_1(t|\mathcal{C})$ and $p_2(t|d)$. In particular, the probability $p_1(t|\mathcal{C})$ defines a *basic randomness model* of the distribution of t in the corpus $\mathcal{C}$, whereas $p_2(t|d)$ defines the *information gain* of observing the term t in the document d. As the amount of information in a document is directly proportional to its length, a third component is introduced to perform a *term frequency normalisation*. Different distributional assumptions for estimating the basic randomness model and the information gain conveyed by the occurrence of a term in a document, as well as different term frequency normalisation schemes, lead to a variety of effective DFR models (Amati, 2003). In the following, we describe examples of models that are used in the experimental part of this thesis. These include both parametric and non-parametric models that assume term independence, as well as an extended non-parametric model that exploits term dependence, in order to promote documents where the query terms occur in close proximity.

Parametric Models Several effective ranking functions can be derived by combining different models of randomness, information gain, and term frequency normalisation (Amati, 2003). While DFR was originally conceived as a framework of non-parametric models (Amati & van Rijsbergen, 2002), subsequent studies have shown that the effectiveness of these models could be further improved by parametrising the term frequency normalisation component for the characteristics of different corpora or for different query lengths (He & Ounis, 2003).

One of the most prominent parametric models in the DFR framework is PL2 (Amati, 2003). This model deploys the Poisson distribution (Poisson, 1837) and Laplace's law of succession (Laplace, 1814) as models of randomness and information gain, respectively. In particular, the Poisson distribution is a limiting case of a binomial process, expressing the probability $p_1(t|\mathcal{C})$ of observing $tf_{t,d}$ occurrences of a term t in a randomly selected document d from the corpus $\mathcal{C}$. After $tf_{t,d}$ occurrences have been observed, the probability $p_2(t|d)$ of observing a further occurrence of t in d—the so-called *aftereffect* of future sampling (Feller, 1968)—is proportional to the number of already observed occurrences, according to Laplace's law of succession. Intuitively, while an informative term may be relatively rare in the corpus, the frequency of this term tends to be high in the documents where it occurs. PL2 instantiates Equation (2.28) as:

$$w_{t,d}^{\mathrm{PL2}} = \frac{1}{tf_{t,d}^{(2)}+1}\left(tf_{t,d}^{(2)} \log_2 \frac{n\, tf_{t,d}^{(2)}}{tf_{t,\mathcal{C}}} + \left(\frac{tf_{t,\mathcal{C}}}{n} - tf_{t,d}^{(2)} \right) \log_2 e + 0.5 \log_2(2\pi\, tf_{t,d}^{(2)}) \right), \tag{2.29}$$

where n is the number of documents in $\mathcal{C}$, $tf_{t,\mathcal{C}}$ is the frequency of the term t in the corpus, and $tf_{t,d}^{(2)}$ is given by the so-called *normalisation 2*, according to:

$$tf_{t,d}^{(2)} = tf_{t,d} \log_2 \left(1 + \gamma \frac{\bar{l}}{l_d} \right), \tag{2.30}$$

where $tf_{t,d}$ is the raw term frequency in d, l_d and $\bar{l}$ are the length of d and the average length of all documents in the corpus, respectively, and γ a is a parameter controlling the amount of normalisation. This model has been shown to be particularly effective for web search (Plachouras & Ounis, 2004).

Non-Parametric Models Although provably effective, PL2 and several other models derived from the DFR framework require tuning the parameter γ in Equation (2.30) (Amati, 2003). Parameter tuning also plays an important role for term frequency normalisation in probabilistic relevance models and for smoothing in language models, as discussed in Sections 2.2.1.1 and 2.2.1.2, respectively. To alleviate the need for extensive tuning while attaining an effective retrieval performance for corpora and queries with different characteristics, Amati (2006) introduced a series of non-parametric DFR models. Such models deploy a hypergeometric distribution (Feller, 1968) as the basic randomness model. Similarly to the binomial distribution (or its previously discussed Poisson approximation), the hypergeometric distribution expresses the probability $p_1(t|\mathcal{C})$ of observing $tf_{t,d}$ occurrences of a term t in a corpus $\mathcal{C}$. Unlike the binomial, the hypergeometric distribution assumes that samples are drawn without replacement, i.e., in a non-independent fashion. As a practical consequence, this randomness model naturally incorporates an inherent non-parametric term frequency normalisation mechanism, hence precluding any need for further parameter tuning.

Of the family of non-parametric DFR models, DPH (Amati et al., 2007) has been shown to perform effectively across a variety of web search tasks (McCreadie et al., 2009; Santos et al., 2010d; McCreadie et al., 2011). Moreover, as it requires no parameter tuning, it is also efficient from a deployment perspective. Besides using a hypergeometric randomness model, DPH estimates the information gain of observing a term inspired by the notion of informative content of a theory introduced by Popper (1934) and extensively studied by Hintikka & Suppes (1970). The weighting scheme of DPH is formulated as:

$$w_{t,d}^{\text{DPH}} = \frac{tf_{t,d}\left(1-\frac{tf_{t,d}}{l_d}\right)^2}{tf_{t,d}+1}\log_2\left(\frac{tf_{t,d}\,\bar{l}n}{l_d\,tf_{t,\mathcal{C}}}\right) + 0.5\log_2\left(2\pi\,tf_{t,d}\left(1-\frac{tf_{t,d}}{l_d}\right)\right). \tag{2.31}$$

Once again, as normalisation is inherent in the model, DPH provides an effective and efficient alternative to other models. For these reasons, it will be used extensively in the experimental part of this thesis, both as a baseline ranking on its own as well as a strong basis for building additional baseline rankings.

Extended Models All previously described DFR models assume that the query terms occur in a document independently of one another. To relax this assumption, Peng et al. (2007b) introduced the pBiL DFR model to exploit higher-order term dependence for ranking documents. Similarly to the MRF model of Metzler & Croft (2005), described in Section 2.2.1.2, pBiL can model different modes of term dependence, such as sequential and full dependence. As Peng et al. (2007b) have shown, full dependence generally outperforms sequential dependence, and is hence the mode used in our experiments. Assuming a full dependence mode, the pBiL weighting scheme can be defined as:

$$
\begin{aligned}
w_{t,d}^{\text{pBiL}} &= \alpha_u \, w_{t,d} \\
&\quad + \alpha_f \sum_{\substack{t_i \in q \\ t_i \neq t}} \frac{1}{tf_{\langle t,t_i \rangle,d} + 1} \Big(- \log_2 (l_d - 1)! + \log_2 tf_{\langle t,t_i \rangle,d}! \\
&\qquad\qquad + \log_2 (l_d - 1 - tf_{\langle t,t_i \rangle,d})! \\
&\qquad\qquad - tf_{\langle t,t_i \rangle,d} \log_2 (1/(l_d - 1)) \\
&\qquad\qquad - (l_d - 1 - tf_{\langle t,t_i \rangle,d}) \log_2 ((l_d - 2)/(l_d - 1)) \Big),
\end{aligned} \tag{2.32}
$$

where the parameters α_u and α_f control the linear interpolation between the unigram and full dependence weights, respectively. The unigram weight, $w_{t,d}$, can be computed using any of the aforementioned DFR models, such as PL2 (Equation (2.29)) or DPH (Equation (2.31)). The term dependence weight combines the binomial randomness model with the Laplace model of information gain to measure the informativeness of occurrences of pairs $\langle t, t_i \rangle$ of query terms in each document d. The resulting factorials in Equation (2.32) can be efficiently computed using Lanczos' approximation of the Gamma function (Lanczos, 1964).

Different from other probability distributions, such as the Poisson and hypergeometric distributions used by PL2 and DPH, respectively, the binomial distribution does not consider the total frequency of each pair $\langle t, t_i \rangle$ in a corpus, which would be computationally expensive to estimate given the combinatorial number of possible pairs. Instead, the informativeness of the pair in the document d is solely dependent on the frequency $tf_{\langle t,t_i \rangle,d}$ of the pair in the document and on the length l_d of the document. As a result, pBiL is also an efficient approach for exploiting term dependence (Peng et al., 2007b; Macdonald & Ounis, 2010).

2.2.2 Query-independent Ranking

The previous section described query-dependent ranking approaches, which infer the extent to which a document is about the topic of the user's query. While topicality is essential for inferring the relevance of a document (Boyce, 1982), there may be too many documents with relatively similar topicality scores for the same query. In addition, some queries may be better answered by sources that fulfil a specific quality criterion, such as authoritativeness, credibility, or trustworthiness, particularly when the user is searching for a specific information provider (Kraaij et al., 2002; Bendersky et al., 2011). To distinguish between documents with similar topicality, and also to address queries that explicitly seek for quality content, several query-independent ranking approaches have been proposed in the literature. In this section, we describe two broad classes of such approaches, which are used in the experimental part of this thesis. In particular, Section 2.2.2.1 describes approaches that infer the a priori quality of a document based upon evidence in the document itself, whereas Section 2.2.2.2 focuses on approaches that infer quality from sources external to the document.

2.2.2.1 On-Document Evidence

A typical assumption underlying query-dependent ranking approaches is that all documents in a corpus are equally relevant a priori (Kraaij et al., 2002). While this assumption may hold when retrieving from curated corpora such as newswire documents, it may be unrealistic in an environment such as the Web (Bendersky et al., 2011). In particular, web documents are produced independently by authors with various motives and backgrounds, leading to a vast heterogeneity in content quality, ranging from high quality sources, such as online encyclopedias, to adversarial content, such as spam (Castillo & Davison, 2011).

Kraaij et al. (2002) were among the first to analyse the usefulness of the a priori evidence of the quality of documents for web search. To this end, they investigated the effectiveness of several features for estimating the document prior $p(d)$ in a query likelihood model, as described in Section 2.2.1.2. Among these, URL-based features were shown to be particularly effective for identifying homepages, a classical web search task (Broder, 2002). For instance, the URL *type* feature

was introduced to distinguish between URLs containing different components, such as a domain name (a "root" URL), a domain followed by a subdirectory (a "subroot" URL), a deeper directory (a "path" URL), or a filename (a "file" URL). In particular, homepages tend to be mainly of type "root" (Kraaij et al., 2002). Given some relevance data $\mathcal{G}$, this feature can be quantified as:

$$f_{\mathrm{UT}}(q,d) = \frac{|\{d_i \in \mathcal{G} \mid \mathit{type}(u_d) = \mathit{type}(u_{d_i})\}|}{|\{d_i \in \mathcal{C} \mid \mathit{type}(u_d) = \mathit{type}(u_{d_i})\}|}, \tag{2.33}$$

where $\mathit{type}(u_d)$ defines the type of the URL u_d of document d. A simpler feature, capturing the intuition that shorter URLs are preferred is the URL *depth* (UD), which counts the number of components in the document's URL:

$$f_{\mathrm{UD}}(q,d) = |\,\mathit{parts}(u_d, \text{`/'})|, \tag{2.34}$$

where $\mathit{parts}(u_d, \text{`/'})$ denotes the set of forward slash-separated substrings of u_d, excluding its protocol (e.g., "http://"). Yet another similar feature counts the number of characters ς_{u_d} in the URL, and is denoted URL *length* (UL):

$$f_{\mathrm{UL}}(q,d) = \varsigma_{u_d}\,. \tag{2.35}$$

Another class of query-independent ranking features used in the experimental part of this thesis exploits the textual content of each document, in order to measure its overall readability. The underlying intuition is that documents that are easier to read are more likely to be perceived as relevant by search users. For instance, Kanungo & Orr (2009) investigated a series of features for the task of generating readable document summaries to be displayed in response to a query (Tombros & Sanderson, 1998). Of these, we use the average term length (ATL) in a document as a simple measure of readability, according to:

$$f_{\mathrm{ATL}}(q,d) = \frac{1}{l_d} \sum_{t \in d} tf_{t,d}\, \varsigma_t\,. \tag{2.36}$$

where ς_t denotes the length in characters of the term t. The intuition here is that longer terms would reflect a more thoughtful, and hence readable writing style.

Additional readability features have been recently proposed by Bendersky et al. (2011). For instance, they proposed to use the entropy $\mathrm{H}(\theta_d)$ of a document's language model θ_d as a measure of topic cohesiveness (TC), according to:

$$f_{\mathrm{TC}}(q,d) = \mathrm{H}(\theta_d) = -\sum_{t \in d} p(t|d) \log p(t|d), \tag{2.37}$$

where $p(t|d)$ was computed using a maximum likelihood estimation, as described in Equation (2.23). Other readability features proposed by Bendersky et al. (2011) include the document's fraction (SF) and coverage (SC) of stopwords, computed as the ratio of terms in the document that are stopwords and the ratio of all stopwords that are covered in the document, respectively, according to:

$$f_{\mathrm{SF}}(q,d) = \frac{|\{t_i \in d\} \cap \mathcal{V}_s|}{|\{t_i \in d\}|}, \tag{2.38}$$

$$f_{\mathrm{SC}}(q,d) = \frac{|\{t_i \in d\} \cap \mathcal{V}_s|}{|\mathcal{V}_s|}, \tag{2.39}$$

where $\mathcal{V}_s$ is a list of stopwords. Both SF and SC are intended as simple estimators of the divergence between the document and the corpus language models, and are positively correlated with the document informativeness (Zhou & Croft, 2005). Another readability feature used in the experimental part of this thesis is the fraction of terms in the document that appear in tables. The underlying intuition here is that documents comprising mostly tabular content are less readable. Let $\mathcal{T}_d$ comprise the textual content appearing within tables in the document d. The table text (TT) feature (Bendersky et al., 2011) can be estimated according to:

$$f_{\mathrm{TT}}(q,d) = \frac{\sum_{t \in d} tf_{t,\mathcal{T}_d}}{l_d}. \tag{2.40}$$

At the lower end of the quality spectrum, the Web is severely affected by *spam*. Spam documents typically include automatically generated content targeting popular search queries, or even human-generated content plagiarised from legitimate sources, so as to deceive search engines and attract larger audiences, which can ultimately result in increased advertisement revenue for the spammer (Castillo & Davison, 2011). In particular, spam documents typically have

abnormally long titles, a total length that deviates from the average length of non-spam documents, a proportionally higher ratio of raw text per HTML markup, and a high redundancy, which is typically a sign of automatic "keyword stuffing". Inspired by the latter observation, Ntoulas et al. (2006) proposed a simple feature for spam detection, denoted compression ratio (CR), and defined as:

$$f_{\mathrm{CR}}(q,d) = \frac{\varsigma_{z(d)}}{\varsigma_d}, \tag{2.41}$$

where $z(d)$ denotes a compressed representation of document d, produced by any standard data compression algorithm (Salomon, 2007), whereas $\varsigma_{z(d)}$ and ς_d are the size (in bytes) of the compressed and uncompressed representations of d. The higher the compressed size $\varsigma_{z(d)}$ and consequently the compression ratio $f_{\mathrm{CR}}(q,d)$, the less redundant, and hence the less likely the document d is to be spam.

A more sophisticated spam detection feature was devised by Cormack et al. (2011). In particular, using a gradient-descent logistic regression classifier (Goodman & tau Yih, 2006) with training data combining manually labelled documents, as well as documents highly ranked for "honey pot" queries (popular queries that are commonly targeted by spammers), they estimated the probability that a document contains harmful or malicious content. Taking the complement event, the probability that a document d is not spam can be used to compute a *ham*[3] likelihood (HL) score as a log-odds estimate, according to:

$$f_{\mathrm{HL}}(q,d) = \log \frac{p(\eta|d)}{p(\bar{\eta}|d)}, \tag{2.42}$$

where η and $\bar{\eta}$ denote the observation of ham and spam content, respectively.

2.2.2.2 Off-Document Evidence

The analysis of the content of a document provides valuable evidence about the quality of this document. On the other hand, such evidence is prone to manipulation by the document author. Indeed, as previously discussed, much of the content produced on the Web is intended to maliciously deceive search engines in order to increase revenue for spammers. While analyses based on off-document

[3] In the jargon of the spam detection community, "ham" is an antonym of "spam".

evidence, such as hyperlinks (Kleinberg et al., 1999) or clicks (Joachims, 2002), are certainly not immune from spammers (Castillo et al., 2007), they provide an arguably more unbiased assessment of the quality of a document, by relying not on the document author, but on other web authors or on web search users.

The view of the Web as a graph of hyperlinked documents brings various opportunities for improving web search. A particularly prominent use of the web graph is for inferring the global importance—or, in graph theorists' terms, the *centrality* (Newman, 2003)—of each document in the graph. In the context of web search, the centrality of a document in the web graph is considered as a measure of authority, as perceived by the entire Web, which has been extensively used for improving the quality of document rankings (e.g., Kleinberg, 1998; Page et al., 1999; Plachouras et al., 2005). A simple measure of the centrality of a document d is its *indegree* (ID), defined as the cardinality of the set $\mathcal{B}_d$ of documents linking to d (i.e., the document's *backlinks*) in the web graph, according to:

$$f_{\text{ID}}(q, d) = |\mathcal{B}_d|. \tag{2.43}$$

An analogous measure to the indegree of a document is its outdegree. Different from the indegree, however, the outdegree of a document is not considered as a measure of global authority. On the contrary, a document with abnormally high outdegree often serves malicious purposes, by inflating the indegree of other documents, a spamming technique known as a *link farm* (Castillo & Davison, 2011). The outdegree (OD) of a document d is defined as the cardinality of the set $\mathcal{F}_d$ of documents linked to from d (i.e., the document's *forward links*):

$$f_{\text{OD}}(q, d) = |\mathcal{F}_d|. \tag{2.44}$$

One of the most well-known link analysis algorithms—and one that is used in our experiments—is PageRank (Page et al., 1999). The PageRank algorithm estimates the global importance of a document based on the number of other documents that link to it and also on the importance of these documents. To this end, the algorithm iteratively performs a random walk on the web graph, so that the score assigned to a document when the algorithm converges can be seen

as the probability of that document being visited by the random walker. The PageRank (PR) of a document d in a graph with n documents is given by:

$$f_{\text{PR}}^{(i+1)}(q,d) = \frac{1-\gamma}{n} + \gamma \sum_{d_j \in \mathcal{B}_d} \frac{f_{\text{PR}}^{(i)}(q,d_j)}{f_{\text{OD}}(q,d_j)}, \tag{2.45}$$

where $f_{\text{PR}}^{(i)}(q,d)$ is the PageRank of d at the i-th iteration, with $f_{\text{PR}}^{(1)}(q,d) = (1/n)$ for all d, $\mathcal{B}_d$ is the set of documents linking to d, $f_{\text{OD}}(q,d_j)$ is the outdegree of $d_j \in \mathcal{B}_d$, given by Equation (2.44), and γ is a damping factor, which can be interpreted as the probability that a random walker will stop following the chain of hyperlinks and "jump" to a randomly selected document. The algorithm iterates until the computed PageRank scores stabilise within a given threshold or until a predefined number of iterations is performed (Brin & Page, 1998).

An alternative, rich source of off-document ranking evidence is based on the quality of a document as perceived by web search users rather than other web authors. In particular, a web search engine can record in a query log a variety of signals describing the interaction of search users during their search tasks. One class of such signals is click evidence. While not all searches lead to clicks—for both positive and negative reasons (Li et al., 2009; Stamou & Efthimiadis, 2010)—a click on a document ranked in response to a query can be seen as an implicit judgement of the relevance of this document, of the non-relevance of the documents ranked ahead of it that were skipped or, more generally, of the user's preference for the clicked document over the skipped ones (Joachims, 2002).

A simple query-independent feature can also be derived by leveraging click evidence. In particular, given the sets of documents displayed ($\mathcal{R}_{q_i}$) and clicked ($\mathcal{K}_{q_i}$) for each query q_i in a query log $\mathcal{L}$, the click likelihood (CL) of a document d models the probability that d will receive a click regardless of any particular query (Richardson et al., 2007), according to:

$$f_{\text{CL}}(q,d) = \frac{\sum_{q_i \in \mathcal{L}} \mathbf{1}_{\mathcal{K}_{q_i}}(d)}{\sum_{q_i \in \mathcal{L}} \mathbf{1}_{\mathcal{R}_{q_i}}(d)}, \tag{2.46}$$

where the indicator functions $\mathbf{1}_{\mathcal{K}_{q_i}}(d)$ and $\mathbf{1}_{\mathcal{R}_{q_i}}(d)$ determine whether the document d belongs to each of the aforementioned sets for each query q_i in the log.

2.2.3 Machine-learned Ranking

The previous sections have introduced several approaches for ranking documents in response to a query. Regardless of these approaches' relative effectiveness when compared to one another, it is extremely unlikely that any single one of them will be effective in all search scenarios (Zhai, 2011). This is particularly true for web search, given the massive size and heterogeneity of the Web and the increasingly complex information needs of web search users (Liu, 2009). On the other hand, each of these approaches can potentially capture a different dimension of the relevance of a document for the user's query. As a result, combining these approaches as multiple features of a unified ranking function emerges as a promising direction for effectively searching the Web (Fuhr, 1989). The automatic construction of such functions is the goal of a branch of machine learning denoted *learning to rank*, which is the focus of this section. In particular, Section 2.2.3.1 introduces the general framework of learning to rank, whereas Section 2.2.3.2 describes the three main families of approaches that adhere to this framework, including the approaches that will be used in the experiments in this thesis.

2.2.3.1 Discriminative Learning Framework

A learning to rank process can be specified within the general framework of discriminative learning (Liu, 2009). In particular, the ultimate goal of learning to rank is to automatically construct a ranking function:

$$f_{\text{LTR}}(q, d) \equiv h : \mathcal{X} \rightarrow \mathcal{Y}, \tag{2.47}$$

where $\mathcal{X}$ and $\mathcal{Y}$ represent the input and output space of learning, respectively. The *input space* $\mathcal{X}$ comprises learning instances, typically represented as feature vectors $\mathbf{x} = \mathbf{\Phi}(q, d)$, where $\mathbf{\Phi}$ is a feature extractor. Each dimension $\phi(q, d)$ of the feature vector could correspond, for instance, to one of the various ranking functions described in the previous sections. The *output space* $\mathcal{Y}$ defines the target of the learning task, which could be either a continuous or a discrete distribution over the learning instances, or simply an overall ordering of these instances. The class of functions h that map from the input to the output space is denoted the

hypothesis space $\mathcal{H}$. Lastly, the loss incurred by the predicted output for the input learning instances compared to these instances' expected output is quantified by a *loss function* Δ, which is used to guide the learning process towards improved ranking functions, for instance, by iteratively minimising the observed loss.

As a supervised or semi-supervised learning task, learning to rank requires some form of *training* (Macdonald, Santos & Ounis, 2013). As illustrated in Figure 2.5, the training data comprises a *sample* $\{(\mathbf{x}_{ij}, y_{ij})\}_{j=1}^{n_{q_i}}$ for each training query q_i, including a feature vector representation $\mathbf{x}_{ij}$ and an output label y_{ij} for each of the top n_{q_i} documents retrieved for q_i, typically by using one of the query-dependent ranking approaches described in Section 2.2.1. The training samples are used by a *learner* module to produce a ranking function h with optimal effectiveness on the training queries, as measured by the loss function Δ. To reduce the possibility that the learned function is overfitted to the training data, and hence generalises poorly to unseen queries, separate *validation* samples may be used to guide the learner. Finally, given a test query q with a sample $\{(\mathbf{x}_j, ?)\}_{j=1}^{n_q}$ sharing the same feature space with the training and validation samples, a *ranker* module applies the learned function h in order to produce an ideally more effective permutation of the documents in the initial sample.

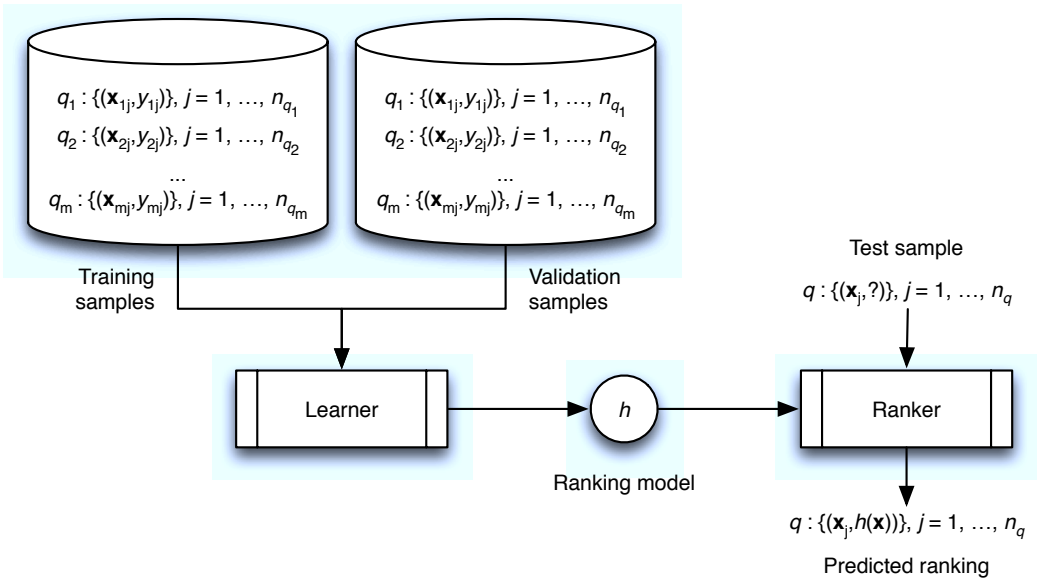


Figure 2.5: Discriminative learning framework.

2.2.3.2 Learning to Rank Approaches

Depending on their choice for implementing each of the input space, output space, hypothesis space, and loss function components, learning to rank approaches can be classified as pointwise, pairwise, or listwise (Liu, 2009). *Pointwise* approaches consider an input space comprising feature vectors built for individual documents, and an output space comprising a single numeric score for each document vector. In this case, learning to rank is reduced to a standard regression task, namely, that of predicting the relevance score of each query-document pair. As a result, a range of existing regression approaches—and classification approaches, for discretised scores—can be directly leveraged for learning to rank (Witten et al., 1999).

Different from pointwise approaches, *pairwise* approaches have an input space comprising pairs of document vectors and an output space covering binary values $\{-1, 1\}$, which denote a preference for one of the two documents in the pair over the other. Accordingly, the hypothesis space covers bivariate functions $h(\mathbf{x}_1, \mathbf{x}_2)$, which can be transformed using a scoring function $f(\mathbf{x})$ for simplicity, i.e., $h(\mathbf{x}_1, \mathbf{x}_2) = 2\ [\mathbf{1}(f(\mathbf{x}_1) > f(\mathbf{x}_2))] - 1$. As their loss function, pairwise approaches minimise the average number of swaps in the ranking (Li, 2011).

A limitation of both pointwise and pairwise approaches is that they ignore the fact that some (pairs of) documents are related to the same query. To overcome this limitation, *listwise* approaches extend their input space to include the entire sample for each query. Accordingly, their output space comprises either a full permutation of the sample, or numeric scores for all documents in the sample. In the latter case, a scoring function $f(\mathbf{x})$ can be used to produce the output, by serving as a sorting criterion, i.e., $h(\{\mathbf{x}_j\}) = sort_{f(\mathbf{x})}\{\mathbf{x}_j\}$. The output space also determines the choice of a loss function. In particular, if the output is a permutation, the prediction loss can be estimated as the difference between the ground-truth and the predicted permutations. Otherwise, with ground-truth labels for all documents, a standard metric for retrieval evaluation can be used to estimate the loss. The latter option has the additional benefit of directly accounting for the actual effectiveness of the ranking—as measured by any standard metric for retrieval evaluation, as will be described in Section 2.3—instead of resorting to an intermediate function as a proxy for retrieval effectiveness (Liu, 2009).

In the experimental chapters of this thesis, we use two learning to rank algorithms: AFS (Metzler, 2007) and LambdaMART (Wu et al., 2008). AFS is a listwise learning to rank algorithm that incrementally builds a hypothesis $h(\{\mathbf{x}_j\})$ as a linear combination of single-feature hypotheses $\hat{h}$, selected iteratively, in a greedy fashion (Metzler, 2007). In particular, at the i-th iteration, AFS selects the single-feature hypothesis $\hat{h}^{(i)}$ that most improves the current hypothesis $h^{(i-1)}$, according to a loss function Δ. The selected single-feature hypothesis $\hat{h}^{(i)}$ is then weighted proportionally to the improvement it brings, with the resulting weight $w^{(i)}$ used to combine it with the current hypothesis $h^{(i-1)}$, according to:

$$h^{(i)}(\{\mathbf{x}_j\}) = h^{(i-1)}(\{\mathbf{x}_j\}) + w^{(i)}\,\hat{h}^{(i)}(\{\mathbf{x}_j\}), \tag{2.48}$$

where $h^{(i)}$ is the resulting hypothesis at the i-th iteration. Metzler (2007) has shown that the greedy learning strategy deployed by AFS suffices for most practical cases, with little benefits observed when retraining all individual weights $w^{(i)}$ after each iteration. Indeed, despite its simplicity, AFS has been shown to perform effectively in a web search setting (Santos et al., 2011d).

Besides AFS, we use LambdaMART (Wu et al., 2008), which represents the current state-of-the-art in learning to rank (Chapelle & Chang, 2011). LambdaMART is a listwise learning to rank algorithm that falls into the general framework of boosting (Kearns, 1988; Schapire, 1990). A boosting algorithm aims to iteratively build a *strong* hypothesis by combining multiple *weak* hypotheses. In particular, given an input sample $\{\mathbf{x}_j\}$, a strong hypothesis (or an *ensemble*) $h(\{\mathbf{x}_j\})$ of weak hypotheses $\hat{h}(\{\mathbf{x}_j\})$ can be iteratively built according to Equation (2.48), where $h^{(i)}(\{\mathbf{x}_j\})$ now represents the resulting ensemble at the i-th iteration, whereas $\hat{h}^{(i)}(\{\mathbf{x}_j\})$ and $w^{(i)}$ represent the learned weak hypothesis and its associated weight at the same iteration, respectively. Different from AFS, LambdaMART models $\hat{h}^{(i)}(\{\mathbf{x}_j\})$ as a multi-feature regression tree, with leaves representing possible prediction outcomes and inner nodes representing decision points that lead to a particular outcome, depending on the conjunction of feature values in the chosen path. An example of such a tree is illustrated in Figure 2.6, with UL (Equation (2.35)), HL (Equation (2.42)), PR (Equation (2.45)), DPH (Equation (2.31)), and pBiL (Equation (2.32)) serving as decision points.

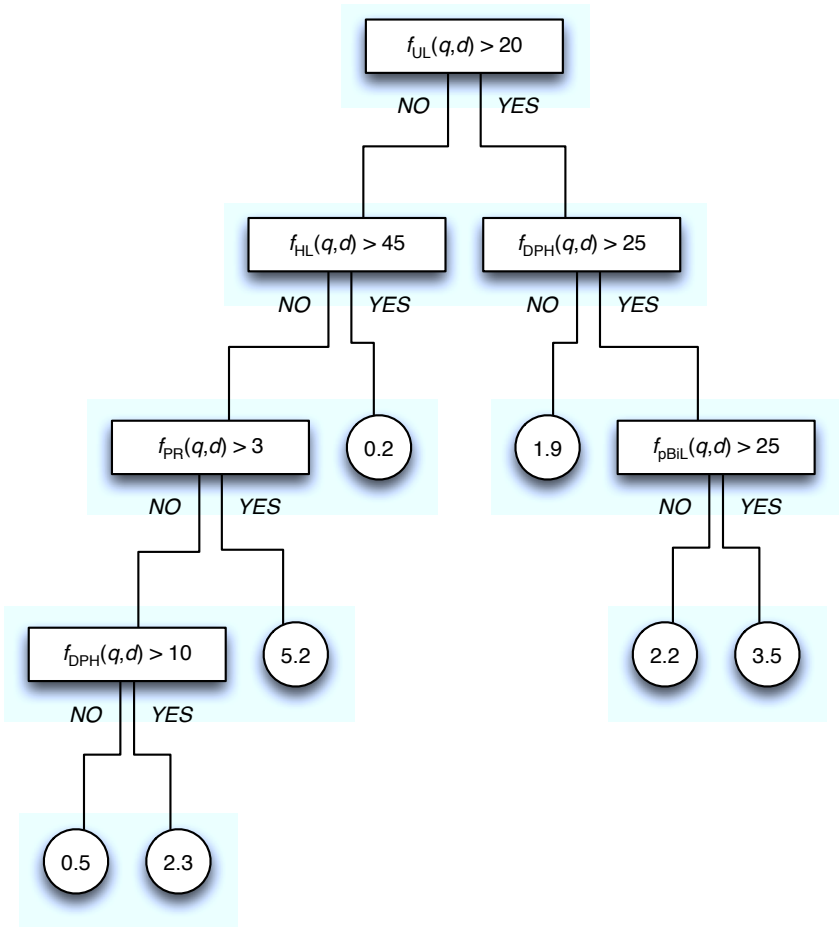


Figure 2.6: Example regression tree with query-independent (URL length (UL), ham likelihood (HL), and PageRank (PR)) and query-dependent (DPH and pBiL) features.

Both AFS and LambdaMART optimise an information retrieval evaluation metric, such as the several metrics introduced in Section 2.3.3, as their loss function Δ. Nevertheless, most such metrics are non-continuous and non-differentiable and hence cannot be optimised directly (Burges et al., 2006). In order to overcome this limitation, AFS leverages an evaluation metric indirectly, as a criterion for selecting the best performing feature at each iteration. LambdaMART, on the other hand, uses the gradient of an evaluation metric (Burges et al., 2006)—as opposed to the metric itself—as a loss function. In particular, in order to learn both a regression tree $\hat{h}^{(i)}(\{\mathbf{x}_j\})$ and its weight $w^{(i)}$ at each iteration, LambdaMART performs a gradient descent optimisation (Friedman, 2001).

2.3 Retrieval Evaluation

Retrieval evaluation is crucial for assessing and improving search technologies. In particular, both the *effectiveness* and the *efficiency* of a search engine can be evaluated. While effectiveness concerns the ability of the search engine to retrieve and rank documents that are relevant to the users' information needs, efficiency is concerned with the speed with which such a ranking is produced. As this thesis is primarily concerned with improving the satisfaction of the users' information needs, in this section, we focus on the evaluation of retrieval effectiveness. In Section 2.3.1, we overview the most prominent methodologies for web search evaluation. In Section 2.3.2, we discuss the particular methodology that underlies all experiments in this thesis. Lastly, in Section 2.3.3, we introduce some of the most prominent metrics for assessing the effectiveness of ranking approaches.

2.3.1 Evaluation Methodologies

Evaluating the effectiveness of web search ranking is an open challenge. Not only is relevance an ill-understood concept per se (Mizzaro, 1997), but it can also span multiple dimensions (Borlund, 2003), particularly in light of the complex information needs of web search users (Broder, 2002; Rose & Levinson, 2004). Alternative evaluation methodologies have been proposed and tested throughout the years, based upon both implicit and explicit user feedback on the relevance of the documents ranked in response to a query (Sanderson, 2010).

Implicit feedback approaches typically rely on the observation of web search users' interactions with the ranking, such as the documents they click on or the time they spend examining a clicked document (Kelly & Teevan, 2003). Treating implicit feedback as an absolute judgement of relevance has important limitations though. On the one hand, clicks are significantly biased by the presentation order of the ranked documents (Craswell et al., 2008). On the other hand, the absence of a click does not necessarily reflect a poor ranking. For instance, the user may leave the search page without clicking on any document, simply because relevant information appears in the snippet of some document (Li et al., 2009; Stamou & Efthimiadis, 2010). A more sensible approach in this situation is to treat the user's feedback as a preference judgement between pairs of documents (Joachims,

2002; Joachims et al., 2005), or even between entire rankings produced by different approaches, presented either side-by-side (Thomas & Hawking, 2006) or interleaved (Radlinski et al., 2008b; Chapelle et al., 2012).

A different evaluation methodology relies on the users' explicit feedback on the effectiveness of ranking approaches. This can be achieved, for instance, by observing real users interacting with the ranking (Saracevic, 1995; Borlund & Ingwersen, 1997). Such a methodology enables the assessment of relevance in context (Ingwersen & Järvelin, 2005), which can contribute to understanding its multiple dimensions (Borlund, 2003). However, this methodology is often costly and therefore limited to small-scale studies. An alternative methodology recently introduced to enable gathering users' feedback at a larger scale is *crowdsourcing* (Alonso et al., 2008). In particular, crowdsourcing platforms, such as Amazon's Mechanical Turk,[4] provide a marketplace where researchers can hire a large number of human judges for a relatively small cost. Nevertheless, the limited knowledge of these judges' background and motivations makes it a challenging task to assure the quality of the evaluation (Carvalho et al., 2011). Another alternative methodology relies on expert judges to produce a benchmark against which multiple ranking approaches can be tested, as we discuss next.

2.3.2 Evaluation Benchmarks

One of the most established retrieval evaluation methodologies abstracts away from the specificities of individual users, instead relying on the relevance assessment of expert judges to produce an evaluation *benchmark* (Voorhees, 2007). Such a methodology was pioneered by Cleverdon (1967) at the College of Aeronautics, Cranfield, UK, in their experiments to assess the effectiveness of multiple indexing approaches. While the so-called *Cranfield paradigm* may limit the assessment of relevance in context (Teevan et al., 2007), it dramatically improves the reproducibility of the resulting evaluation, by allowing multiple ranking approaches to be tested on a common benchmark (Voorhees & Harman, 2005). Moreover, it is estimated that such a methodology has fostered around one third of all improvement in web search ranking from 1999 to 2009 (Rowe et al., 2010).

[4] http://www.mturk.com

The Text REtrieval Conference (TREC), one of the major forums for research in information retrieval (Voorhees & Harman, 2005; Voorhees, 2007) can be seen as a modern instantiation of the Cranfield paradigm. In particular, TREC was introduced in 1992 in a co-sponsorship between the National Institute of Standards and Technology (NIST) and the Defense Advanced Research Projects Agency (DARPA), both U.S. government agencies. Since its inception, the conference has witnessed a substantial increase in the number of participant groups working on several different search scenarios (known as *tracks* in the TREC jargon).

The overall aim of TREC is to support information retrieval research by providing the necessary infrastructure for the evaluation of retrieval techniques on a common benchmark, known as a *test collection*. A test collection comprises three components: a corpus of documents, a set of stated information needs (called *topics*), and a set of relevance assessments, which function as a mapping between each topic and the documents deemed as relevant for this topic. A prototypical TREC track works as follows (Voorhees, 2007). Firstly, a document corpus is built so as to serve as a common testbed for experimentation in the particular search scenario addressed by the track, such as web search. Secondly, NIST provides the participants with a set of topics representing realistic information needs for the search scenario under consideration. Thirdly, in order to build a ground-truth for evaluating the participants' approaches as to the extent to which they are able to retrieve the relevant documents in the corpus for the devised topics ahead of irrelevant ones, a process called *pooling* (Spärck Jones & van Rijsbergen, 1975) is usually employed. This process consists of building a pool of documents for each of the considered topics as the union of the top documents retrieved for that topic by all the participant systems. These document pools are then sampled and submitted to manual relevance assessment. Finally, the participant groups submit the document rankings (known as *runs*) generated by their different retrieval approaches for each of the considered topics. These document rankings are then scored based on the produced relevance assessments according to several standard evaluation metrics, such as those discussed in Section 2.3.3. By evaluating the participants' approaches using this common benchmark, TREC allows for the direct comparison of their deployed ranking techniques, hence identifying which techniques work best for the retrieval scenario under consideration.

2.3.3 Evaluation Metrics

Several metrics have been proposed in the literature to evaluate the effectiveness of ranking approaches using a benchmark test collection (Sanderson, 2010). Given a query q and a cutoff κ, the goal of an evaluation metric is to quantify how well the top κ documents from a ranking $\mathcal{R}_q$, produced by some ranking approach, cover the documents $\mathcal{G}_q$ judged relevant for q. Since different queries may have different numbers of relevant documents, the evaluation score for a given query is typically *normalised* by the maximum attainable score for this query, which is equivalent to the score assigned by the metric to an ideal ranking.

Perhaps the most basic metrics associated with retrieval effectiveness are *precision* (P) and *recall* (R) (Cleverdon & Keen, 1966). While precision measures the fraction of retrieved documents that are relevant, recall measures the fraction of relevant documents that are retrieved. These metrics are defined as:

$$\mathrm{P}(q,\kappa) = \frac{|\mathcal{G}_q \cap \mathcal{R}_q^{(\kappa)}|}{|\mathcal{R}_q^{(\kappa)}|} \qquad \text{and} \qquad \mathrm{R}(q,\kappa) = \frac{|\mathcal{G}_q \cap \mathcal{R}_q^{(\kappa)}|}{|\mathcal{G}_q|}, \tag{2.49}$$

where $\mathcal{R}_q^{(\kappa)}$ is the set of top κ documents retrieved for q and $\mathcal{G}_q$ is the set of documents relevant to this query. As observed by Cleverdon & Keen (1962), precision and recall often have an inverse relationship, as illustrated in Figure 2.7. For instance, precision-improving approaches, such as term dependence weighting (Metzler & Croft, 2005; Peng et al., 2007b), typically lead to reduced recall, as relevant documents that do not contain the query terms in close proximity are demoted. Conversely, recall-improving techniques, such as query expansion (Rocchio, 1971; Lavrenko & Croft, 2001; Zhai & Lafferty, 2001), typically incur some topic drift, potentially promoting non-relevant documents.

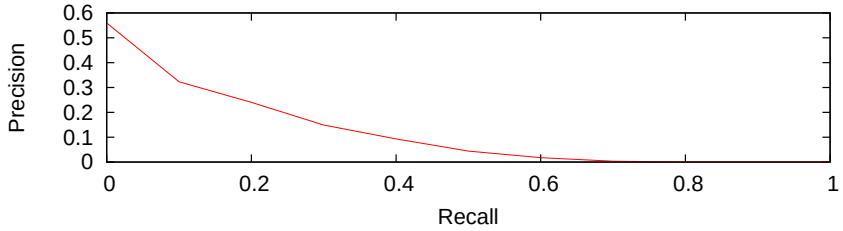


Figure 2.7: Example precision vs. recall graph.

A clear limitation of set-based metrics such as precision and recall is that they are insensitive to ranking swaps above the rank cutoff where these metrics are computed (Robertson, 2008). For instance, two approaches retrieving the same amount of relevant documents in the top 10 positions will receive exactly the same precision and recall scores at cutoff 10, regardless of how well each approach ranks these documents. One of the first metrics to address the limitation was *average precision* (AP; Harman, 1993). It is defined according to:

$$\mathrm{AP}(q, \kappa) = \frac{\sum_{i=1}^{\kappa} \mathrm{P}(q, i)\, g_i}{|\mathcal{G}_q|}, \tag{2.50}$$

where $\mathrm{P}(q, i)$ denotes the ranking precision at the i-th position, according to Equation (2.49), whereas g_i denotes whether or not the i-th document in the ranking $\mathcal{R}_q$ is relevant. Indeed, as originally conceived, average precision assumes that relevance is a binary quantity, an assumption that also underlies the probabilistic relevance modelling approaches described in Section 2.6.

While a binary assumption simplifies the processes of both assessing and inferring the relevance of documents, such an assumption is arguably limiting. Indeed, in a large and heterogeneous corpus such as the Web, different documents are likely to be relevant to the same query to different extents (Teevan et al., 2007). To account for a non-dichotomous notion of relevance, Järvelin & Kekäläinen (2002) considered a scenario where the relevance of a document is assessed using a graded scale, from less relevant to more relevant. In addition, they proposed to use a log-based discount factor to model the fact that relevant documents ranked high are preferred over lower ranked ones. The resulting metric, denoted *discounted cumulative gain* (DCG), is defined as:

$$\mathrm{DCG}(q, \kappa) = \sum_{i=1}^{\kappa} \frac{2^{g_i} - 1}{\log_2(i + 1)}, \tag{2.51}$$

where g_i now denotes a non-binary relevance grade associated with the document ranked at the i-th position. In a typical web search scenario, five relevance grades are used (Burges et al., 2005). In addition, different logarithm bases can be used to simulate smaller or larger discounts (Järvelin & Kekäläinen, 2002).

The formulation of DCG assumes that the probability of a user inspecting a particular document depends only on the position of this document in the ranking. Moffat & Zobel (2008) argued that different users may not have the same willingness to inspect documents at lower ranks. To cater for such a varying user behaviour, they introduced the *rank-biased precision* (RBP) metric, a graded relevance metric with a parameter p denoting the (fixed) probability that a user will inspect a further document. The higher the value of p, the more persistent the user. The RBP metric can be defined by incorporating this probability into a geometric discount function, according to:

$$\mathrm{RBP}(q, \kappa) = (1 - p) \sum_{i=1}^{\kappa} g_i \, p^{i-1}, \tag{2.52}$$

where g_i is as defined for the DCG metric in Equation (2.51).

Although having different discount factors, both DCG and RBP assume that the probability that the user will inspect a given document does not depend on the documents previously inspected. In practice, such an independence assumption does not fit well the users' observed click behaviour. In particular, Craswell et al. (2008) observed that the probability that a user will click on a given document diminishes as higher ranked documents are clicked. Intuitively, according to this *cascade browsing model*, once a user has found the desired information, the need for inspecting further documents is reduced. As a result, this model tends to favour rankings that contain novel information, as will be discussed in Chapter 3. Chapelle et al. (2009) quantified the effectiveness of a ranking according to this model into the expected reciprocal rank (ERR) metric, defined as:

$$\mathrm{ERR}(q, \kappa) = \sum_{i=1}^{\kappa} \frac{1}{i} \prod_{j=1}^{i-1} (1 - p_j) \, p_i, \tag{2.53}$$

where p_i denotes the probability that the i-th document is relevant to the query, in which case $\prod_{j=1}^{i-1}(1 - p_j)$ denotes the probability that none of the documents ranked higher than the i-th document is relevant. In practice, p_i is defined as a function of the relevance grade g_i of the i-th document, i.e., $p_i = (2^{g_i} - 1)/2^{g_{\max}-1}$, where $g_{\max}$ is the maximum grade considered.

2.4 Summary

In order to lay out the foundations for the work contributed in this thesis, this chapter provided a comprehensive and up-to-date background on web information retrieval in general, and on web search ranking in particular.

Starting with an overview of the typical operation of a web search engine, in Section 2.1, we described the processes of crawling, indexing, and query processing. Within the scope of the latter, in Section 2.2, we provided a contextualised background on over 50 years' worth of literature on ranking in information retrieval. This encompassed classical approaches to query-dependent ranking in Section 2.2.1, including the three main families of probabilistic ranking approaches. In addition, in Section 2.2.2, we described several query-independent ranking approaches, which emerged with the advent of the Web. The framework of learning to rank was introduced in Section 2.2.3 as a sound mechanism for integrating multiple ranking approaches as individual features of a strong ranking model. Lastly, in Section 2.3, we reviewed different methodologies for retrieval evaluation, with a further look into the most established metrics for assessing the adhoc retrieval effectiveness of a ranking approach.

In common, all ranking approaches described in this chapter assume that a query submitted to a web search engine represents a single, well-defined information need. In the next chapter, we will discuss the limitations of this assumption in a complex search environment such as the Web, and the new ranking problem that results from abandoning such an assumption.

Chapter 3

Search Result Diversification

Ranking in IR has been traditionally approached as a pursuit of relevant information, under the assumption that the users' information needs are unambiguously conveyed by their submitted queries (Spärck-Jones et al., 2007; Sanderson, 2008). While such an assumption may have arguably held in the library setting where the early studies of relevance-oriented ranking were conducted (Maron & Kuhns, 1960; Cooper, 1971; Harter, 1975a,b; Robertson, 1977), it does not hold in general (Gordon & Lenk, 1992), and it is unlikely to hold for web search in particular.

Web search queries are typically short, ranging from two to three terms on average (Jansen et al., 2000). While short queries are more likely to be ambiguous, even longer queries can show some degree of ambiguity (Song et al., 2009), which in turn can substantially affect the effectiveness of web search engines (Sanderson, 2008). In order to identify relevant information under the uncertainty posed by query ambiguity, an effective approach is to diversify the search results. By doing so, the search engine can minimise the chance of wrongly guessing the users' needs, which might cause the users to abandon their queries (Chen & Karger, 2006).

In this chapter, we describe the search result diversification problem. In particular, Section 3.1 discusses how query ambiguity manifests in web search, as a motivation for diversifying the search results. Section 3.2 starts with a historical perspective on the diversification problem, before providing a formal definition and an analysis of the complexity of the problem. Section 3.3 describes several related approaches for diversifying the search results. Lastly, Section 3.4 extends the discussion initiated in Section 2.3 with an emphasis on diversity evaluation.

3.1 Query Ambiguity

As an inherently limited representation of a more complex information need, every query can be arguably considered ambiguous to some extent (Cronen-Townsend & Croft, 2002). Nevertheless, in the query understanding literature, query ambiguity is typically classified into three broad classes (Clarke et al., 2008; Song et al., 2009). At one extreme of the ambiguity spectrum, genuinely *ambiguous queries* can have multiple *interpretations*. For instance, it is generally unclear whether the query *"bond"* refers to a debt security certificate or to Ian Fleming's fictional secret agent character.[1] Next, *underspecified queries* have a clearly defined interpretation, but it may be still unclear which particular *aspect* of this interpretation the user is interested in. For instance, while the query *"james bond"* arguably has a clearly defined interpretation (i.e., the secret agent character), it is unclear whether the user's underlying information need is for books, films, games, etc. Finally, at the other extreme, *clear queries* have a generally well understood interpretation. An example such query is *"james bond books"*.

Sanderson (2008) investigated the impact of query ambiguity on web search. In particular, he analysed queries from a 2006 query log of a commercial web search engine that exactly matched a Wikipedia disambiguation page[2] or a WordNet[3] entry. Ambiguous queries from Wikipedia showed a larger number of senses on average than those from WordNet (7.39 vs. 2.96), with the number of senses per ambiguous query following a power law in both cases. The average length of an ambiguous query was also similar across the two sources, with the predominance of single-word queries. In contrast to previous works, which assumed that multi-word queries were relatively unaffected by ambiguity, he found that ambiguous queries with more than one term were also numerous. Importantly, he observed that ambiguous queries comprised over 16% of all queries sampled from the query log, with Wikipedia queries being more frequent than WordNet ones, particularly among popular queries. Finally, through a simulation, he showed that current search systems underperform for ambiguous queries.

[1] As a matter of fact, Wikipedia's disambiguation page for *"bond"* lists over 100 possible meanings for this particular entry: http://en.wikipedia.org/wiki/Bond.

[2] http://en.wikipedia.org/wiki/Wikipedia:Disambiguation

[3] http://wordnet.princeton.edu

Song et al. (2009) analysed the ambiguity of web search queries through a user study. In their study, five assessors manually classified 60 queries sampled from the log of a commercial search engine from August 2006 as either ambiguous, underspecified, or clear queries. While a high assessor agreement (90%) was observed for judging whether a given query was ambiguous or not, distinguishing between underspecified and clear queries turned out to be substantially more difficult. Nonetheless, based on the demonstrated feasibility of the former case, they proposed a binary classification approach to automatically identify ambiguous queries. Based on the learned classification model, they estimated that 16% of the queries in their entire query log sample were ambiguous.

Another log analysis of query ambiguity was performed by Clough et al. (2009). In their analysis, a total of 14,909 unique queries that satisfied minimum frequency criteria were selected from a one-month sample of the query log of a commercial search engine from 2006. Of the sample queries, 18% had a high click entropy, which quantified the spread of each query's clicked documents. Such queries were mostly informational, whereas queries with a low entropy were predominantly navigational (Broder, 2002). Analysing the subset of queries with an exact match among Wikipedia disambiguation pages, they found no significant correlation between click entropy and the number of suggested interpretations on Wikipedia. However, they observed that queries with a dominant interpretation on Wikipedia had a higher entropy. Such queries tended to be underspecified, with clicks covering a range of aspects of the dominant interpretation. In particular, they found a significant correlation between the entropy of these queries and the total length of the corresponding articles on Wikipedia, suggesting that they indeed covered broad topics. Finally, considering both queries with high entropy and those with at least one reformulation in the query log, they estimated that from 9.5% to 16.2% of all queries in their sample were ambiguous.

The aforementioned studies characterised query ambiguity from different perspectives. In common, all studies reached the surprisingly consensual figure that around 16% of all user queries are ambiguous, while many more can be underspecified to some degree. In the next section, we will discuss how query ambiguity can pose challenges to traditional ranking approaches, and how search result diversification can be deployed to address such challenges.

3.2 Ranking under Uncertainty

Throughout the years, the probability ranking principle (PRP; Cooper, 1971; Robertson, 1977), discussed in Section 2.2.1.1, has served as a general policy for ranking in IR (Gordon & Lenk, 1991). However, the development of probabilistic ranking has been permeated by simplifying modelling assumptions that are often inconsistent with the underlying data (Gordon & Lenk, 1992; Cooper, 1995).

Gordon & Lenk (1991, 1992) analysed the optimality of the PRP under the light of both decision and utility theories (von Neumann & Morgenstern, 1944). In the context of document ranking, while decision theory assigns a cost to retrieving each document independently of other documents, utility theory considers the overall benefit of retrieving a set of documents. Besides the definitional assumption that probabilities are well-calibrated,[4] Gordon & Lenk (1991) discussed two key assumptions underlying probabilistic ranking approaches in IR:

A1. The probability of relevance is estimated with *certainty*, and is provided as a single point estimate, with no associated measure of risk.

A2. The probability of relevance is estimated for a query-document pair *independently* of the estimated probability of relevance of the other documents.

As Gordon & Lenk (1991) demonstrated, the PRP attains the greatest expected utility compared to any other ranking policy under the aforementioned assumptions. However, when at least one of these assumptions fails to hold, the principle is suboptimal (Gordon & Lenk, 1992). In general, neither A1 nor A2 are realistic assumptions. Regarding A1, uncertainty arises naturally from the fact that the probability of relevance is estimated based upon limited representations of both information needs and information items (Turtle & Croft, 1996). The former is particularly the case in complex search environments such as the Web, where queries are often ambiguous, as discussed in Section 3.1.

Regarding A2, the limitation of assuming that documents are conditionally independent given the query was early recognised. In his note on relevance as a measurable quantity, Goffman (1964) pointed out that *"the relationship between*

[4] According to the definition of Gordon & Lenk (1991), a well-calibrated IR system is one that predicts an accurate probability of relevance for each document.

a document and a query is necessary but not sufficient to determine relevance." Intuitively, once a document satisfying the user's information need has been observed, it is arguable whether other documents satisfying the same need would be deemed relevant. This intuition has been empirically corroborated in recent years with the analysis of users' browsing behaviour from click logs. Indeed, as discussed in Section 2.3.3, users' clicks on the ranked documents are better explained by a cascade model (Craswell et al., 2008), in which the probability of clicking on a given document diminishes as higher ranked documents are clicked.

3.2.1 The Search Result Diversification Problem

The aforementioned assumptions, A1 and A2, generally do not hold in a realistic search scenario, such as web search. While A1 is challenged by *ambiguity* in the user's query, A2 is challenged by *redundancy* in the ranking. In order to overcome these limitations, *search result diversification* has been proposed as a generalisation of the standard ranking problem, where ambiguity and redundancy are no longer ruled out by simplifying assumptions (Bennett et al., 2008).

Departing from these assumptions requires viewing an ambiguous query as representing not one, but multiple information needs (Spärck-Jones et al., 2007). Under this view, query ambiguity can be tackled by ensuring a high *coverage*[5] of the possible information needs underlying the query. In turn, redundancy can be tackled by ensuring a high *novelty* with respect to the covered needs. Analogously to the traditional single-need ranking problem, coverage and novelty can be seen as a generalisation of recall and precision, respectively, as introduced in Section 2.3.3. Just as it happens with recall and precision, coverage and novelty can also be conflicting goals (Gollapudi & Sharma, 2009). Indeed, a ranking with maximum coverage may not attain maximum novelty (e.g., although covering all information needs, the ranking may place all documents covering a particular need ahead of documents covering other needs). Conversely, a ranking with maximum novelty may not attain maximum coverage (e.g., although covering each need as early as possible in the ranking, not all possible needs may be covered).

[5] Clarke et al. (2008) refer to this concept as *"diversity"*. We call it *"coverage"* to emphasise the fact that it is one component of the broader search result diversification problem.

Coverage and novelty can be combined to define the search result diversification problem. Informally, the problem can be stated as that of producing a ranking with maximum coverage and maximum novelty with respect to the possible information needs underlying a query, as illustrated in Figure 3.1. The figure also contrasts a diversity-oriented ranking from a traditional relevance-oriented ranking, which assumes that a single information need underlies the query.

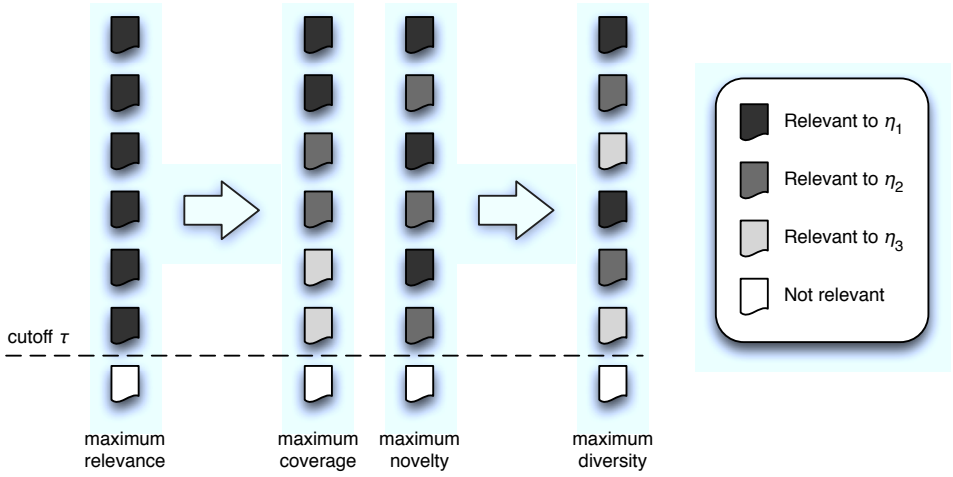


Figure 3.1: Relevance-oriented ranking and the often conflicting goals of diversity-oriented ranking, namely, to attain maximum coverage and maximum novelty.

Formally, let $\mathcal{R}_q$ denote the ranking produced for the query q by a relevance-oriented ranking approach, such as those described in Section 2.2. Moreover, let $\mathcal{N}_q$ and $\mathcal{N}_d$ denote the sets of information needs for which the query q and each document $d \in \mathcal{R}_q$ are relevant, respectively. The goal of the search result diversification problem is to find a subset $\mathcal{D}_q \in 2^{\mathcal{R}_q}$, such that:

$$\mathcal{D}_q = \arg\max_{\mathcal{D}'_q \in 2^{\mathcal{R}_q}} \left| \cup_{d \in \mathcal{D}'_q} \mathcal{N}_q \cap \mathcal{N}_d \right|, \text{ s.t. } |\mathcal{D}'_q| \leq \tau, \tag{3.1}$$

where $\tau > 0$ is the *diversification cutoff*, denoting the number of top documents from $\mathcal{R}_q$ to be diversified, and $2^{\mathcal{R}_q}$ is the power set of $\mathcal{R}_q$, comprising all subsets $\mathcal{D}'_q$ of $\mathcal{R}_q$, with $0 < |\mathcal{D}'_q| \leq \tau$, to be considered as candidate permutations of $\mathcal{R}_q$. The permutation with the maximum number of covered information needs up to the cutoff τ is chosen as the optimal diversified ranking $\mathcal{D}_q$.

3.2.2 Complexity Analysis

The search result diversification problem, as formalised in Equation (3.1), is an instance of the maximum coverage problem, a classical NP-hard problem in computational complexity theory (Hochbaum, 1997), which can be stated as:

> Given a universe of elements $\mathcal{U}$, a collection of potentially overlapping subsets $\mathcal{W} \in 2^{\mathcal{U}}$, and an integer τ, select a set of subsets $\mathcal{M} \subseteq \mathcal{W}$, with $|\mathcal{M}| \leq \tau$, with maximum coverage of the elements from $\mathcal{U}$.

To show that the diversification problem is also NP-hard, we can reduce the maximum coverage problem to it (Agrawal et al., 2009). In particular, we map the universe of elements $\mathcal{U}$ to the possible information needs $\mathcal{N}_q$ underlying the query q. Likewise, we map the collection of candidate subsets $\mathcal{W}$ to the documents in $\mathcal{R}_q$, initially retrieved for q, in which case each document $d \in \mathcal{R}_q$ can be seen as a subset of the information needs $\eta \in \mathcal{N}_q$ for which this document is relevant. As a result, it can be easily verified that a set of subsets $\mathcal{M} \subseteq \mathcal{W}$, $|\mathcal{M}| \leq \tau$, has maximum coverage of the elements in $\mathcal{U}$ if and only if a permutation $\mathcal{D}_q \subseteq \mathcal{R}_q$, $|\mathcal{D}_q| \leq \tau$, has maximum diversity with respect to the information needs in $\mathcal{N}_q$.

Since the diversification problem is NP-hard, we must look for a polynomial-time approximate solution. An important observation to this end is that the maximisation objective in Equation (3.1) shows a *submodular* structure (Vohra & Hall, 1993). In particular, given arbitrary sets $\Gamma_1, \Gamma_2 \subseteq \mathcal{U}$, with $\Gamma_1 \subseteq \Gamma_2$, and an element $\gamma \in \mathcal{U} \setminus \Gamma_2$, a set function $f : 2^{\mathcal{U}} \rightarrow \mathbb{R}$ is called submodular if and only if $f(\Gamma_1 \cup \{\gamma\}) - f(\Gamma_1) \geq f(\Gamma_2 \cup \{\gamma\}) - f(\Gamma_2)$. In other words, adding a new element γ to Γ_1 causes an equal or higher increment in f compared to adding γ to Γ_1's superset Γ_2. Intuitively, a submodular function captures the notion of *decreasing marginal utility*, a fundamental principle in economics (Samuelson & Nordhaus, 2001). In the context of search result diversification, the marginal utility of selecting a further document relevant to an information need diminishes the more this need is satisfied by the documents already selected.

A greedy algorithm can be used to solve the submodular function optimisation in Equation (3.1). As described in Algorithm 3.1, this greedy approach takes as input a query q, the initial ranking $\mathcal{R}_q$, with $|\mathcal{R}_q| = n_q$, and the diversification cutoff $\tau \leq n_q$. As its output, the algorithm produces a permutation $\mathcal{D}_q \subseteq \mathcal{R}_q$,

with $|\mathcal{D}_q| = \tau$. Such a permutation is initialised as an empty set in line 1 and iteratively constructed in lines 2–6 of Algorithm 3.1. In line 3, the submodular objective function $f(q, d, \mathcal{D}_q)$ scores each yet unselected document $d \in \mathcal{R}_q \setminus \mathcal{D}_q$ in light of the query q and the documents already in $\mathcal{D}_q$, selected in the previous iterations of the algorithm. The highest scored document, d^*, is then removed from $\mathcal{R}_q$ and added to $\mathcal{D}_q$ in lines 4 and 5, respectively. Finally, in line 7, the produced diverse permutation $\mathcal{D}_q$ of the initial ranking $\mathcal{R}_q$ is returned.

Diversify$(q, \mathcal{R}_q, \tau)$
1 $\mathcal{D}_q \leftarrow \emptyset$
2 **while** $|\mathcal{D}_q| < \tau$ **do**
3 $\quad d^* \leftarrow \arg\max_{d \in \mathcal{R}_q \setminus \mathcal{D}_q} f(q, d, \mathcal{D}_q)$
4 $\quad \mathcal{R}_q \leftarrow \mathcal{R}_q \setminus \{d^*\}$
5 $\quad \mathcal{D}_q \leftarrow \mathcal{D}_q \cup \{d^*\}$
6 **end while**
7 **return** $\mathcal{D}_q$

Algorithm 3.1: Greedy search result diversification.

The asymptotic cost of Algorithm 3.1 is the product of two factors: the cost ϖ_i of evaluating the function f in line 3 at the i-th iteration, and the number Λ_τ of such evaluations required by the algorithm to identify the τ most diverse documents. The unitary cost ϖ_i varies for different approaches, as will be discussed in Section 3.3. For approaches adhering to the greedy strategy in Algorithm 3.1, the number of evaluations Λ_τ performed up to (and including) the i-th iteration can be modelled as a recurrence relation. In particular, at the first iteration (i.e., $i = 1$), the most diverse document is trivially selected as the one with the highest estimated relevance to the query, independently of the other documents, since $\mathcal{D}_q = \emptyset$ at this point. At the i-th iteration, with $i > 1$, the function f is evaluated for each document $d \in \mathcal{R}_q \setminus \mathcal{D}_q$, which amounts to a total of $n_q - (i - 1)$ documents. These two observations can be modelled as the base and recursion steps of a first-order linear recurrence, respectively, according to:

$$\Lambda_1 = 0, \tag{3.2}$$

$$\Lambda_i = n_q - i + 1 + \Lambda_{i-1}. \tag{3.3}$$

To obtain the total number of evaluations Λ_τ required to select the τ most diverse documents from $\mathcal{R}_q$, we can iteratively expand the recursion step (Equation (3.3)) through telescoping (Cormen et al., 2001), until we finally arrive at the base step (Equation (3.3)), according to:

$$\Lambda_\tau = n_q - \tau + 1 + \Lambda_{\tau-1}, \tag{3.4}$$

$$\Lambda_{\tau-1} = n_q - \tau + 2 + \Lambda_{\tau-2}, \tag{3.5}$$

$$\dots$$

$$\Lambda_2 = n_q - \tau + (\tau - 1) + \Lambda_1. \tag{3.6}$$

Replacing Equation (3.2) into (3.6), and back-replacing Equations (3.5)-(3.6) up into Equation (3.4), we can derive a closed form for Λ_τ, as follows:

$$\begin{aligned}
\Lambda_\tau &= \sum_{i=2}^{\tau}(n_q - \tau + i) + \Lambda_1 \\
&= \sum_{i=2}^{\tau}(n_q - \tau) + \sum_{i=2}^{\tau} i + 0 \\
&= \frac{1}{2}(2\tau n_q - \tau^2 - 2n_q + \tau).
\end{aligned} \tag{3.7}$$

With $\tau \leq n_q$, it follows from Equation (3.7) that $\Lambda_\tau = \mathcal{O}(\tau n_q)$. As $\tau \to n_q$, we have $\Lambda_\tau = \mathcal{O}(n_q^2)$. An important non-approximability result is known for this polynomial-time algorithm, which stems from the submodular structure of the objective function f. In particular, Nemhauser et al. (1978) have shown that such a greedy algorithm achieves an approximation factor of $(1 - 1/e) \approx 0.632$ of the optimal solution to the maximum coverage problem. Feige (1998) has further demonstrated that, for any $\epsilon > 0$, the optimal solution cannot be approximated within a ratio of $(1 - 1/e) + \epsilon$, unless P = NP. This result was independently confirmed under a weaker assumption by Khuller et al. (1999), who proved that no approximation algorithm with ratio better than $(1-1/e)$ exists for the maximum coverage problem, unless NP $\subseteq$ DTIME($n_q^{\mathcal{O}(\log \log n_q)}$). Given the approximation guarantee offered by Algorithm 3.1, this algorithm underlies most diversification approaches in the literature, as will be described in Section 3.3, as well as the framework introduced in this thesis, which we will describe in Chapter 4.

3.3 Related Approaches

Most diversification approaches in the literature differ by how they implement the objective function $f(q, d, \mathcal{D}_q)$ in Algorithm 3.1. In this thesis, we propose to organise these approaches according to two complementary dimensions, as described in Table 3.1: aspect representation and diversification strategy (Santos et al., 2010e, 2012b). An *aspect representation* determines how the information needs underlying a query are represented as multiple *aspects* of this query.[6][7] In particular, an *implicit* representation relies on features intrinsic to each document in order to model different aspects, such as the terms contained in the document (Carbonell & Goldstein, 1998), or those derived from different language models (Zhai et al., 2003), topic models (Carterette & Chandar, 2009), or clusters (He et al., 2011) built from the initial ranking. In turn, an *explicit* representation seeks to directly approximate the possible information needs underlying a query, by relying on features derived from the query itself, such as its associated clicks (Radlinski et al., 2008a), reformulations (Santos et al., 2010a), or categories (Agrawal et al., 2009).

Given a particular aspect representation, a *diversification strategy* determines how to achieve the goal of satisfying the different query aspects. *Coverage*-based approaches achieve this goal by directly estimating how well each document covers each aspect of the query, regardless of the other retrieved documents. Depending on the underlying aspect representation, coverage can be estimated in terms of classification confidence (Agrawal et al., 2009), topicality (Carterette & Chandar, 2009), and relevance (Santos et al., 2010a,e). In contrast, *novelty*-based approaches directly compare the retrieved documents to one another, regardless of their covered aspects, in order to promote novel information. For instance, documents can be compared in terms of content dissimilarity (Carbonell & Goldstein, 1998), divergence (Zhai et al., 2003), or relevance score correlation (Rafiei et al., 2010; Wang & Zhu, 2009). Finally, the advantages of both coverage and novelty can be combined into a *hybrid* diversification strategy (Santos et al., 2012b).

[6] Unless otherwise noted, we will refer to query interpretations and aspects indistinctly.

[7] While both queries and aspects are *representations* of information needs, we find the following distinction helpful: a query is a potentially ambiguous representation of an information need in the classical "single-need" view of ranking, whereas an aspect is an unambiguous representation of one need when multiple needs are considered, as discussed in Section 3.2.1.

Table 3.1: Representative diversification approaches in the literature, organised into two complementary dimensions: diversification strategy and aspect representation.

Diversification strategy	Aspect representation	
	Implicit	Explicit
Novelty	Carbonell & Goldstein (1998) Zhai et al. (2003) Chen & Karger (2006) Zhu et al. (2007) Wang & Zhu (2009) Rafiei et al. (2010) Zuccon & Azzopardi (2010) Gil-Costa et al. (2011, 2013)	Santos et al. (2012b)
Coverage	Carterette & Chandar (2009) He et al. (2011)	Radlinski & Dumais (2006) Radlinski et al. (2008a) Capannini et al. (2011) Santos et al. (2012b)
Hybrid	Yue & Joachims (2008) Santos et al. (2010e) Raman et al. (2012)	Agrawal et al. (2009) Santos et al. (2010a) Slivkins et al. (2010)

3.3.1 Novelty-based Approaches

Novelty-based approaches have the longest history in the search result diversification literature, stemming from research on identifying novel sentences for text summarisation (Carbonell & Goldstein, 1998). The definitional characteristic of such approaches is their account for dependences between the ranked documents, and consequently their strict adherence to the formulation in Algorithm 3.1.

The novelty-based diversification approaches in the literature typically differ according to their estimation of document dependence. As highlighted in Table 3.1, the vast majority of these approaches adopts an implicit aspect representation, typically comprising the space of unique terms in a document corpus.[8] For such approaches, at the i-th iteration, an evaluation of the objective function $f(q, d, \mathcal{D}_q)$ would have a cost $\varpi_i \propto v(i-1)$, where v is the number of unique terms

[8]To enable the assessment of the effectiveness of novelty as a diversification strategy in isolation from the aspect representation dimension, in Chapter 8, we introduce the first explicit novelty-based diversification approaches in the literature.

in the lexicon. Nonetheless, in reality, the function f must only be evaluated with respect to the last document added to $\mathcal{D}_q$ (as opposed to the entire set $\mathcal{D}_q$), since the yet unselected documents in $\mathcal{R}_q \setminus \mathcal{D}_q$ would have already been compared to the documents added to $\mathcal{D}_q$ in the previous iterations. As a result, complementing the explanation in Section 3.2.2, the total cost incurred by a novelty-based diversification approach can be expressed as $\sum_{i=1}^{\Lambda_\tau} \varpi_i = \sum_{i=1}^{\mathcal{O}(\tau n_q)} v = \mathcal{O}(v\tau n_q)$.

The first novelty-based diversification approach in the literature was introduced by Carbonell & Goldstein (1998), with applications to text retrieval and summarisation. In particular, their maximal marginal relevance (MMR) method scored a candidate document $d \in \mathcal{R}_q \setminus \mathcal{D}_q$ as the document's estimated relevance with respect to the query q, discounted by the document's maximum similarity with respect to the already selected documents in $\mathcal{D}_q$, according to:

$$f_{\text{MMR}}(q, d, \mathcal{D}_q) = \lambda f_1(q, d) - (1 - \lambda) \max_{d_j \in \mathcal{D}_q} f_2(d, d_j), \tag{3.8}$$

where $f_1(q, d)$ and $f_2(d, d_j)$ estimate the relevance of d to the query q and its similarity to the documents already in $\mathcal{D}_q$, respectively. A balance between relevance (i.e., f_1) and redundancy (i.e., $\max f_2$, the opposite of novelty) is achieved through an appropriate setting of the linear combination parameter λ.

Inspired by the formulation of MMR, Zhai et al. (2003) proposed a novelty-based diversification approach within a risk minimisation (RM) framework for language modelling (Zhai & Lafferty, 2006). In particular, given a query q and a candidate document d, their approach estimated the score of the document model θ_d with respect to the query model θ_q, as well as a reference model $\theta_{\mathcal{D}_q}$, comprising the documents already selected, according to:

$$f_{\text{RM}}(q, d, \mathcal{D}_q) = f_1(\theta_q, \theta_d)(1 - \lambda - f_2(\theta_d, \theta_{\mathcal{D}_q})), \tag{3.9}$$

where $f_1(\theta_q, \theta_d)$ was estimated using the KL ranking function, as described in Equation (2.22). Six methods were proposed in order to estimate $f_2(\theta_d, \theta_{\mathcal{D}_q})$, based on either the divergence between θ_d and $\theta_{\mathcal{D}_q}$ or a mixture of the reference model $\theta_{\mathcal{D}_q}$ and an English background model. Similarly to Equation (3.8), the parameter λ controls the penalisation of redundancy.

A related risk-aware approach was proposed by Chen & Karger (2006). In particular, they argued that maximising the probability of relevance could lead to a complete retrieval failure when ranking under uncertainty. Instead, they proposed to maximise the chance of retrieving at least one relevant document in the ranking. To this end, they instantiated the objective function in Algorithm 3.1 to estimate the conditional relevance (CR) of a document d, under the assumption that none of the already selected documents $\mathcal{D}_q$ were relevant, according to:

$$f_{\text{CR}}(q, d, \mathcal{D}_q) = p(g_{r(d)} \mid \bar{g}_1, \cdots, \bar{g}_{|\mathcal{D}_q|}, d_1, \cdots, d_{|\mathcal{D}_q|}, d), \tag{3.10}$$

where $r(d)$ denotes the ranking position of document d, and g_i and $\bar{g}_i$ denote the events in which the document at the i-th position is relevant and non-relevant, respectively. Intuitively, this formulation promotes novelty by considering the already selected documents as a form of negative relevance feedback.

Wang & Zhu (2009) introduced a diversification approach[9] inspired by the portfolio theory in finance (Markowitz, 1952). In particular, the selection of documents for a ranking involves a fundamental risk, namely, that of overestimating the relevance of individual documents, analogously to the risk involved in selecting financial assets (e.g., stocks) for an investment portfolio. In both the finance and the retrieval scenarios, diversifying the selected items can maximise the expected return (mean) while minimising the involved risk (variance) of a particular selection. Wang & Zhu (2009) proposed to deploy such a mean-variance analysis (MVA) as a diversification objective, according to:

$$f_{\text{MVA}}(q, d, \mathcal{D}_q) = \mu_d - b\, w_i\, \sigma_d^2 - 2\, b\, \sigma_d \sum_{d_j \in \mathcal{D}_q} w_j\, \sigma_{d_j}\, \rho_{d,d_j}, \tag{3.11}$$

where μ_d and σ_d^2 are the mean and variance of the relevance estimates associated with document d, respectively, with the summation component estimating the redundancy of d in light of the documents in $\mathcal{D}_q$. Documents are compared in terms of the Pearson's correlation ρ_{d,d_j} of their relevance estimates. The weight w_i assigns a discount to the document at the i-th ranking position. A balance between relevance, variance, and redundancy is achieved with the parameter b.

[9] A very similar approach was proposed independently by Rafiei et al. (2010).

Building upon the formalism of quantum mechanics (Dirac, 1930), Zuccon & Azzopardi (2010) proposed the quantum probability ranking principle (QPRP). In contrast to the classic PRP (Cooper, 1971; Robertson, 1977), introduced in Section 2.2.1.1, the QPRP prescribes that not only the estimated relevance of each document should be considered as a ranking criterion, but also how it interferes with the estimated relevance of the other documents. In particular, in the quantum formalism, interference refers to the effect of an observation on subsequent observations. This notion was quantified into the following objective:

$$f_{\text{QPRP}}(q, d, \mathcal{D}_q) = p(\mathcal{G}_q|q, d) + \sum_{d_j \in \mathcal{D}_q} \varrho_{d,d_j}, \tag{3.12}$$

where $p(\mathcal{G}_q|q, d)$ denotes the probability of observing the relevant set $\mathcal{G}_q$, given the query q and the document d, which corresponds to the classic formulation of the PRP in Equation (2.6). The estimation of the interference ϱ_{d,d_j} between d and each document $d_j \in \mathcal{D}_q$ involves operations with complex numbers. In practice, it can be approximated as $\varrho_{d,d_j} \approx -2\sqrt{p(\mathcal{G}_q|q, d)}\sqrt{p(\mathcal{G}_q|q, d_j)}\, f(d, d_j)$, where $f(d, d_j)$ can be any function measuring the similarity between the two documents.

Zhu et al. (2007) approached the diversification problem as an absorbing random walk (ARW) with transition probabilities $p_{ij} = (1 - \lambda)\, p(d_j|q) + \lambda\, p(d_j|d_i)$, where $p(d_j|q)$ and $p(d_j|d_i)$ denoted the estimated relevance of d_j and its similarity to d_i, respectively, with the parameter λ balancing between the two scores. An absorbing random walk is a Markov chain with reachable absorbing states i, such that $p_{ij} = 1$ if $i = j$, and 0 otherwise (Kemeny & Snell, 1960). In their formulation, each already selected document $d_j \in \mathcal{D}_q$ was represented as an absorbing state, in which case candidate documents were scored according to:

$$f_{\text{ARW}}(q, d, \mathcal{D}_q) = \vartheta(d, \mathcal{D}_q), \tag{3.13}$$

where $\vartheta(d, \mathcal{D}_q)$ denotes the expected number of visits to document d before absorption by the states in $\mathcal{D}_q$. While this computation would incur an inversion of the underlying transition matrix at every iteration, in practice, such an inversion can be computed only once and reused subsequently to update the portion of the matrix corresponding to the states in $\mathcal{R}_q \setminus \mathcal{D}_q$ (Woodbury, 1950).

Gil-Costa, Santos, Macdonald & Ounis (2011, 2013) explored the properties of the metric space induced from the ranking produced for a query in order to identify novel documents. To this end, they deployed different techniques to partition the initial ranking $\mathcal{R}_q$ into zones $\mathcal{Z}_q$, with each zone comprising documents similar to each other and dissimilar from documents in the other zones. Since $|\mathcal{Z}_q| \ll |\mathcal{D}_q|$, they were able to drastically reduce the number of document comparisons required to promote novelty, by comparing each candidate document $d \in \mathcal{R}_q \setminus \mathcal{D}_q$ to each identified zone centre $z \in \mathcal{Z}_q$, instead of all previously selected documents $d_j \in \mathcal{D}_q$. While such centres could be directly returned as a diverse selection of documents, Gil-Costa et al. (2011) introduced a scoring function to perform what they called a sparse spatial selection diversification (SSSD):

$$f_{\text{SSSD}}(q, d, \mathcal{D}_q) = (1 - \lambda)\, f_1(q, d) + \lambda\Big(1 - \max_{z \in \mathcal{Z}_q} f_2(d, z)\Big), \tag{3.14}$$

where $f_1(q, d)$ and $f_2(d, z)$ estimate the relevance of d to the query q and its similarity—as given by a metric distance—to each zone centre z, with the parameter λ controlling the trade-off between the two scores.

3.3.2 Coverage-based Approaches

Different from novelty-based approaches, coverage-based approaches do not account for dependences between the ranked documents. Instead, they attempt to maximise the coverage of multiple query aspects by each independently selected document, regardless of the aspects covered by the other documents. As a result, these approaches do not adhere to the greedy formulation in Algorithm 3.1.

Although such an independence assumption breaks the effectiveness guarantees offered by the greedy approximation, it improves the efficiency of the resulting diversification. In particular, while novelty-based approaches evaluate the objective function $f(q, d, \mathcal{D}_q)$ a total of $\mathcal{O}(\tau n_q)$ times, only $\mathcal{O}(n_q)$ evaluations are required by coverage-based approaches. The cost of a single evaluation, in turn, depends on the total number of represented aspects k, i.e., $\varpi_i = \mathcal{O}(k)$. Similarly to the analysis conducted for novelty-based approaches, we can express the total cost incurred by coverage-based approaches as $\sum_{i=1}^{\Lambda_\tau} \varpi_i = \sum_{i=1}^{\mathcal{O}(n_q)} k = \mathcal{O}(k n_q)$.

Carterette & Chandar (2009) proposed a probabilistic approach for maximising the coverage of multiple "facets", representing different aspects of a query. Such facets were generated by constructing either relevance models (Lavrenko & Croft, 2001) or topic models (Blei et al., 2003) from the top retrieved documents for the query. Three strategies were proposed to re-rank the initially retrieved documents $\mathcal{R}_q$ with respect to their coverage of the identified facets. In particular, the best performing of these strategies selected the highest scored document d for each facet $z \in \mathcal{Z}_q$ in a round-robin fashion. Such a facet modelling (FM) approach can be formalised into the following objective function:

$$f_{\text{FM}}(q, d, \mathcal{D}_q) = \begin{cases} p(d|q) & \text{if } \exists z_i \in \mathcal{Z}_q \mid p(d|z_i) > 0 \wedge i = |\mathcal{D}_q| \bmod |\mathcal{Z}_q|, \\ 0 & \text{otherwise,} \end{cases} \tag{3.15}$$

where $\mathcal{Z}_q$ is the set of facets identified for the query q, $p(d|z_i)$ denotes the likelihood of observing each document d given the facet $z_i \in \mathcal{Z}_q$. The modulus operation ensures a round-robin selection from a total of $|\mathcal{Z}_q|$ facets. Since the probabilities $p(d|z_i)$ are not comparable across facets, the documents selected in the round-robin process are ultimately ordered by their likelihood given q.

A similar approach was investigated by He et al. (2011), by partitioning the documents initially retrieved for a query into non-overlapping clusters using topic modelling (Blei et al., 2003). In their approach, each cluster $c \in \mathcal{Z}_q$ received a score $p(c|q)$, given by the cluster's likelihood of generating the query q. As a result, the diversification problem was reduced to the task of selecting documents with a high coverage of highly scored clusters. Of the selection strategies investigated, a weighted round-robin selection (WRR) performed the best. This selection strategy can be formalised according to:

$$f_{\text{WRR}}(q, d, \mathcal{D}_q) = \begin{cases} p(d|q) & \text{if } \exists c_i \in \mathcal{Z}_q \mid d \in c_i \wedge i = |\mathcal{D}_q| \bmod |\mathcal{Z}_q| \\ & \text{s.t. } p(c_1|q) \geq p(c_2|q) \geq \cdots \geq p(c_{\mathcal{Z}_q}|q), \\ 0 & \text{otherwise,} \end{cases} \tag{3.16}$$

where the probability $p(c_i|q)$ imposes a total ordering over the clusters $c_i \in \mathcal{Z}_q$, essentially biasing the round-robin selection towards highly scored clusters.

Radlinski & Dumais (2006) proposed to diversify the documents retrieved for a query according to these documents' coverage of multiple reformulations of the query, mined from a query log. In particular, given a query q, they selected the k queries most likely to follow q across multiple sessions in a query log as a set $\mathcal{S}_q$ of query reformulations. In order to select the τ most diverse documents from the ranking $\mathcal{R}_q$, they enforced a proportional coverage of the identified reformulations. According to this proportional coverage (PC) policy, each reformulation $s \in \mathcal{S}_q$ could be represented by at most τ/k documents, which essentially filtered out documents covering already well covered reformulations, according to:

$$f_{\text{PC}}(q, d, \mathcal{D}_q) = \begin{cases} f(q,d) & \text{if } \exists s \in \mathcal{S}_q \mid d \in \mathcal{R}_s \wedge |\mathcal{R}_s \cap \mathcal{D}_q| < \tau/k, \\ 0 & \text{otherwise,} \end{cases} \tag{3.17}$$

where $\mathcal{R}_s$ is the set of documents that match the reformulation s. Despite ensuring a proportional coverage of different reformulations, the selected documents are still ranked by their estimated relevance to the initial query, $f(q, d)$.

In a similar vein, Capannini et al. (2011) proposed to mine query specialisations (i.e., queries with a more specific representation of the user's information need compared to the initial query (Boldi et al., 2009b)) from a query log in order to guide the diversification process. In particular, they selected the τ most diverse documents from $\mathcal{R}_q$ according to each document's weighted proportional coverage (WPC) of the identified specialisations $s \in \mathcal{S}_q$. More precisely, their approach can be formalised into the following objective function:

$$f_{\text{WPC}}(q, d, \mathcal{D}_q) = \begin{cases} f(q,d) & \text{if } \exists s \in \mathcal{S}_q \mid d \in \mathcal{R}_s \wedge |\mathcal{R}_s \cap \mathcal{D}_q| < p(s|q)\,\tau, \\ 0 & \text{otherwise,} \end{cases} \tag{3.18}$$

where $p(s|q)\,\tau$ is the proportion of the final ranking dedicated to documents matching each specialisation $s \in \mathcal{S}_q$, given each specialisation's likelihood $p(s|q)$. For documents matching a not well represented specialisation s, $f(q, d)$ denotes each document's utility, such that $f(q, d) \propto \sum_{s \in \mathcal{S}_q} p(s|q) \sum_{d_j \in \mathcal{R}_s} \frac{1 - f(d, d_j)}{r(d_j, \mathcal{R}_s)}$, where $\mathcal{R}_s$ is a ranking produced for each specialisation s and $f(d, d_j)$ measures the similarity between d and each document $d_j \in \mathcal{R}_s$, ranked at position $r(d_j, \mathcal{R}_s)$.

Radlinski et al. (2008a) proposed an online learning approach to maximise the coverage of clicks for a given query. Their intuition was that users with different information needs would click on different documents for the same query. In their formulation, the choice of the next document to be selected for a query was seen as a multi-armed bandit (MAB) problem (Berry & Fristedt, 1985). A MAB models the process of selecting one of many possible strategies or "arms", trading off the exploitation of existing knowledge and the acquisition (or exploration) of new knowledge. In the context of the diversification problem, each candidate document $d \in \mathcal{R}_q \setminus \mathcal{D}_q$ for the next ranking position of a query q was considered as an "arm", with existing knowledge $\mu_d^{(i)}$ at time i denoting the likelihood of the document being clicked when ranked at that position for the query. Precisely, their ranked-armed bandits (RAB) objective can be described as:

$$f_{\text{RAB}}(q, d, \mathcal{D}_q) = \begin{cases} 1 & \text{if } d = \text{MAB}_j(\mathcal{R}_q, \mu_\bullet^{(i)}), \\ 0 & \text{otherwise,} \end{cases} \tag{3.19}$$

where $\text{MAB}_j(\mathcal{R}_q, \mu_\bullet^{(i)})$ is a MAB instance specifically trained to select a document $d^* \in \mathcal{R}_q$ for the j-th ranking position, with $j = |\mathcal{D}_q| + 1$, balancing exploration and the exploitation of the expected reward $\mu_{d^*}^{(i)}$ at time i.

3.3.3 Hybrid Approaches

Hybrid search result diversification approaches combine the benefits of both coverage and novelty-based approaches. On the one hand, they try to certify that multiple aspects of the initial query are covered in the ranking. On the other hand, they strive to ensure that the covered aspects are novel with respect to the aspects covered by the other documents. As an inherited characteristic of novelty-based approaches, hybrid approaches also account for dependences between the ranked documents. As a result, hybrid approaches also strictly adhere to the greedy formulation in Algorithm (3.1). In addition, the account of document dependences gives hybrid approaches a total cost of $\sum_{i=1}^{\Lambda_\tau} \varpi_i = \sum_{i=1}^{\mathcal{O}(\tau n_q)} k = \mathcal{O}(k\tau n_q)$. Compared to the $\mathcal{O}(v\tau n_q)$ cost incurred by pure novelty-based approaches, hybrid approaches are more efficient, since typically $k \ll v$.

Yue & Joachims (2008) proposed a hybrid diversification approach within the framework of supervised machine learning. As training data, they considered a pair $(\mathcal{R}_{q_i}, \mathcal{N}_{q_i})$ for each query q_i, where $\mathcal{R}_{q_i}$ and $\mathcal{N}_{q_i}$ denoted the initially ranked documents and the manually labelled information needs possibly underlying q_i, respectively. Since the actual needs $\mathcal{N}_{q_i}$ are unknown in a real scenario, these were implicitly represented using the words covered by each document. In order to learn a function f to identify a set $\mathcal{D}_{q_i} \subseteq \mathcal{R}_{q_i}$ with maximum coverage of $\mathcal{N}_{q_i}$, they employed structural support vector machines (SVMs; Tsochantaridis et al., 2005). In particular, their weighted word coverage (WWC) approach considered linear functions f, parametrised by a weight vector $\mathbf{w}$, according to:

$$f_{\text{WWC}}(q, d, \mathcal{D}_q) = \mathbf{w}^T \mathbf{\Phi}(\mathcal{R}_q, \mathcal{D}_q \cup \{d\}), \tag{3.20}$$

where the feature extractor $\mathbf{\Phi}(\mathcal{R}_q, \mathcal{D}_q \cup \{d\})$ measures the extent to which the words in $\mathcal{R}_q$ are covered by each candidate selection $\mathcal{D}_q \cup \{d\}$.

A supervised learning approach similar to the one of Yue & Joachims (2008) was introduced by Raman et al. (2012), but within an online learning setting. In particular, at a given time i, their approach presented the user with a diverse ranking $\mathcal{D}_q$, produced by the following objective:

$$f_{\text{DP}}(q, d, \mathcal{D}_q) = \mathbf{w}_i^T \mathbf{\Phi}(\mathcal{R}_q, \mathcal{D}_q \cup \{d\}), \tag{3.21}$$

where $\mathbf{w}_i$ denotes the weight vector learned by a diversification perceptron (DP), based upon the evidence accumulated up to time i, and $\mathbf{\Phi}(\mathcal{R}_q, \mathcal{D}_q \cup \{d\})$ is defined in terms of word coverage, similarly to Equation (3.20). To update the vector $\mathbf{w}_i$, the feedback received from the user in the form of pairwise preferences is used to produce an improved (in expectation) ranking $\hat{\mathcal{D}}_q$. In particular, the updated vector is defined as $\mathbf{w}_{i+1} = \mathbf{w}_i + \mathbf{\Phi}(\mathcal{R}_q, \hat{\mathcal{D}}_q) - \mathbf{\Phi}(\mathcal{R}_q, \mathcal{D}_q)$.

Hybrid approaches based on explicit aspect representations have also been proposed. For instance, Slivkins et al. (2010) introduced a hybrid diversification approach within the multi-armed bandits (MAB) framework. In particular, they extended the click coverage maximisation approach of Radlinski et al. (2008a), described in Section 3.3.2, to account for the context in which clicks are observed.

To this end, they proposed to condition the expected reward $\mu^{(i)}_{d|\mathcal{D}_q}$ of each document d at time i on the documents $\mathcal{D}_q$ selected ahead of d. This can be formalised into the following objective function, denoted ranked context bandits (RCB):

$$f_{\text{RCB}}(q, d, \mathcal{D}_q) = \begin{cases} 1 & \text{if } d = \text{MAB}_j(\mathcal{R}_q, \mu^{(i)}_{\bullet|\mathcal{D}_q}), \\ 0 & \text{otherwise,} \end{cases} \tag{3.22}$$

where, similarly to Equation (3.19), the instance $\text{MAB}_j(\mathcal{R}_q, \mu^{(i)}_{\bullet|\mathcal{D}_q})$ selects a document $d^* \in \mathcal{R}_q$ for the j-th ranking position, with $j = |\mathcal{D}_q| + 1$, but instead using the conditional reward $\mu^{(i)}_{d^*|\mathcal{D}_q}$ at time i, by correlating the clicks on d^* to those observed for the documents $d_j \in \mathcal{D}_q$. To reduce the number of required correlation computations, they modelled the reward function $\mu_\bullet$ as a Lipschitz-continuous function in the metric space induced by the documents in $\mathcal{R}_q$ (Searcóid, 2006), which dramatically improved the efficiency of the proposed approach.

Agrawal et al. (2009) sought to diversify a document ranking in light of a taxonomy $\mathcal{T}$ of query intents, represented as different categories from the Open Directory Project (ODP).[10] Given the classification of both queries and documents in light of this taxonomy, they proposed an intent-aware selection (IA-Select) mechanism, instantiating the objective function in Algorithm 3.1 as:

$$f_{\text{IA-Select}}(q, d, \mathcal{D}_q) = \sum_{c \in \mathcal{T}} f(c|q, \mathcal{D}_q)\, f(d|q, c), \tag{3.23}$$

where, for each category $c \in \mathcal{T}$, $f(d|q, c)$ denotes the extent to which the document d covers c, while $f(c|q, \mathcal{D}_q)$ denotes the marginal utility of c given the query q and the documents already in $\mathcal{D}_q$. Intuitively, an already well covered category is deemed less useful, which contributes to the promotion of novel documents.

The search result diversification framework introduced in this thesis also falls into the family of hybrid approaches. In Chapter 4, we will discuss how particular choices for explicitly representing the query aspects and for estimating the diversity of the retrieved documents with respect to each aspect lead to a principled, effective, and flexible solution to the diversification problem. Before that, in the remainder of this chapter, we will introduce approaches for diversity evaluation.

[10] http://www.dmoz.org/

3.4 Diversity Evaluation

A diverse ranking is one that satisfies the multiple information needs possibly underlying an ambiguous query—be these needs from different users or from the same user in different contexts. While traditional web search evaluation is challenging, departing from the assumption that a single information need underlies each query arguably renders the evaluation of diversity even more complex. In this section, we review the literature on diversity evaluation. In particular, Section 3.4.1 extends the discussion in Section 2.3.2 with an emphasis on diversity evaluation benchmarks. In turn, Section 3.4.2 describes diversity evaluation metrics, as an extension of the traditional metrics introduced in Section 2.3.3.

3.4.1 Evaluation Benchmarks

As discussed in Section 2.3.2, search systems have greatly benefited from the controlled evaluation offered by benchmark test collections. On the other hand, query ambiguity has been largely ignored by early test collections, similarly to how traditional ranking approaches have ignored query ambiguity, as discussed in Section 3.2. In practice, the assumption that the user's query represents a single information need reduces the complexity of the underlying evaluation, ensuring that different systems are evaluated with respect to an unambiguously defined information need (Cleverdon, 1991). However, as pointed out by Spärck-Jones et al. (2007), this assumption is far from holding in the real world, particularly with the high incidence of short an ambiguous queries. As discussed in Section 3.1, such queries can negatively impact search effectiveness (Sanderson, 2008).

In order to address such a limitation of the established evaluation paradigm, Spärck-Jones et al. (2007) argued for the development of test collections that explicitly account for queries with different levels of ambiguity. In particular, they claimed that such a test collection should consider each query as representing an ensemble of information needs, as opposed to a single need. In turn, such needs should reflect the interests of the population of users that could have issued the query. Finally, the relevance of each ranked document should be judged separately for each information need, so as to enable the assessment of the effectiveness of the whole ranking at satisfying the multiple needs.

Analogously to the instantiation of diversification approaches, discussed in Section 3.3, diversity evaluation is typically operationalised by representing the possible information needs underlying a query as multiple query aspects.[11] Early attempts to build a test collection for diversity evaluation were made at the TREC 6-8 Interactive tracks (Over, 1997, 1998; Hersh & Over, 1999). The investigated task, called "aspect retrieval", involved finding documents covering as many different aspects of a given query as possible. In this evaluation campaign, a total of 20 topics were adapted from the corresponding years of the TREC Ad hoc tracks (Voorhees & Harman, 1997, 1998, 1999). Each topic included from 7 to 56 aspects, as identified by TREC assessors, with relevance assessments provided at the aspect level. Figure 3.2 illustrates one of such topics, 353i, along with some of its identified aspects, denoted "sub-topics" in the TREC jargon.

```
<topic number="353i">
  <query> antarctic exploration </query>
  <description>
    Identify systematic explorations and scientific investigations of
    Antarctica, current or planned.
  </description>
  <subtopic number="1"> mining prospection </subtopic>
  <subtopic number="2"> oil resources </subtopic>
  <subtopic number="3"> rhodium exploration </subtopic>
  <subtopic number="4"> ozone hole / upper atmosphere </subtopic>
  <subtopic number="5"> greenhouse effect </subtopic>
    ...
</topic>
```

Figure 3.2: TREC-7 Interactive track, topic 353i and its sub-topics.

By relying on expert judges to identify query aspects from the retrieved documents (Lagergren & Over, 1998), the TREC Interactive track test collection arguably lacks in plausibility and completeness in light of the actual information needs of the population of users issuing a query (Radlinski et al., 2010b). In order to overcome this limitation, Radlinski et al. (2010a) proposed to identify realistic query aspects for diversity evaluation from the query and click logs of a commercial search engine. In their approach, candidate aspects were selected as queries that frequently co-occurred with the initial query across multiple sessions

[11] Note that the aspect representation adopted by a diversification approach does not necessarily reflect the ground-truth aspect representation adopted for evaluation purposes.

in the query log. Candidates with a low transition probability after a two-step random walk on the bipartite query-document click graph (Craswell & Szummer, 2007) were filtered out. The remaining candidates were then clustered using a graph partitioning algorithm (Blondel et al., 2008). The highest-scoring aspects from different clusters were shown to better reflect real user needs compared to aspects proposed by expert judges (Radlinski et al., 2010a,b). As a result, these aspects served as the basis for a new test collection, developed in the context of the TREC 2009-2012 Web tracks (Clarke et al., 2009a, 2010, 2011b, 2012).

The diversity task of the TREC 2009-2012 Web tracks currently provides the largest publicly available test collections for diversity evaluation. As of 2011,[12] these test collections comprised a total of 150 topics, with 2 to 8 associated aspects each (Clarke et al., 2009a, 2010, 2011b). As such, these collections are chosen as benchmarks for the experiments conducted in this thesis. An example TREC Web track topic, along with its identified aspects, is shown in Figure 3.3. In contrast to the short description provided by the TREC Interactive track test collection, the TREC Web track aspects include a natural language description of the information need represented by each aspect. Moreover, each aspect is further classified as either informational ("inf") or navigational ("nav") by TREC assessors, depending on the intent of its underlying need (Broder, 2002).

```
<topic number="1">
  <query> obama family tree </query>
  <description>
    Find information on President Barack Obama's family history, including
    genealogy, national origins, places and dates of birth, etc.
  </description>
  <subtopic number="1" type="nav">
    Find the TIME magazine photo essay "Barack Obama's Family Tree".
  </subtopic>
  <subtopic number="2" type="inf">
    Where did Barack Obama's parents and grandparents come from?
  </subtopic>
  <subtopic number="3" type="inf">
    Find biographical information on Barack Obama's mother.
  </subtopic>
</topic>
```

Figure 3.3: TREC 2009 Web track, topic 1 and its sub-topics.

[12]The TREC 2012 Web track is ongoing at the time of writing.

Another test collection for the evaluation of web search result diversification was recently introduced as part of the NTCIR-9 Intent task (Song et al., 2011a).[13] Initiated in 1999, NTCIR is a series of evaluation workshops designed to assess information retrieval on Asian languages, as well as across different languages. For the NTCIR-9 Intent task, two test collections were developed, aimed at evaluating search result diversification on the Chinese and the Japanese Web. In particular, the Chinese collection comprised 100 topics, with 4 to 15 associated aspects each. For Japanese, another 100 topics were developed, each with 3 to 22 aspects. An example Chinese topic (translated to English) is shown in Figure 3.4.

```
<topic number="0015">
  <query> mozart </query>
  <subtopic number="1" probability="0.241379310344828">
    mozart's music download
  </subtopic>
  <subtopic number="2" probability="0.241379310344828">
    mozart's biography
  </subtopic>
  <subtopic number="3" probability="0.241379310344828">
    works by mozart
  </subtopic>
  <subtopic number="4" probability="0.126436781609195">
    mozart's concerts
  </subtopic>
  ...
</topic>
```

Figure 3.4: NTCIR-9 Intent task (Chinese), topic 0015 and its sub-topics.

Different from the diversity task of the TREC 2009-2011 Web tracks, the NTCIR-9 Intent task included graded (i.e., non-binary) relevance assessments at the aspect level. In addition, as shown in Figure 3.4, the identified aspects were assigned non-uniform probabilities, estimated through assessor agreement, in order to place more emphasis on popular aspects during the evaluation (Sakai & Song, 2012). While these extensions certainly introduce interesting nuances for diversity evaluation, in order to ensure a consistently uniform experimental setup throughout this thesis, we opted not to use these test collections. Nonetheless, an evaluation of the framework introduced in this thesis on both NTCIR-9 Intent task test collections was conducted by Santos et al. (2011f).

[13] The NTCIR-10 Intent task is also ongoing at the time of writing.

3.4.2 Evaluation Metrics

Several metrics have been proposed in recent years to evaluate the diversification effectiveness of a document ranking. Given a query q and a cutoff κ, a diversity evaluation metric quantifies the extent to which the top κ documents in a ranking $\mathcal{R}_q$ cover the aspects $\mathcal{A}_q$, representing the information needs $\mathcal{N}_q$ underlying q.

The most straightforward metric for diversity evaluation is perhaps sub-topic recall (SC; Zhai et al., 2003). Also known as intent recall (Sakai et al., 2010), this metric quantifies the amount of unique aspects $\mathcal{A}_q$ of the query q that are covered by the top κ ranked documents $d \in \mathcal{R}_q^{(\kappa)}$, according to:

$$\text{SR}(q, \kappa) = \frac{\cup_{d \in \mathcal{R}_q^{(\kappa)}} |\mathcal{A}_q \cap \mathcal{A}_d|}{|\mathcal{A}_q|}, \tag{3.24}$$

where $\mathcal{A}_d$ is the set of aspects for which the document $d \in \mathcal{R}_q^{(\kappa)}$ is relevant.

A limitation of sub-topic recall is that it does not take into account the probability of different aspects given the submitted query. Ideally, this probability should reflect the fraction of the user population that is interested in the information need represented by each aspect. Two evaluation frameworks that take into account the (potentially non-uniform) probability of different aspects have been proposed in the literature. In common, these frameworks generate diversity-oriented metrics as a natural extension of relevance-oriented evaluation metrics in the presence of multiple query aspects. The first of these frameworks, denoted "intent-aware",[14] was introduced by Agrawal et al. (2009). In particular, they defined an intent-aware (IA) metric Eval-IA(q, κ) as the *expected value* of its counterpart relevance-oriented metric Eval(a, κ), with $a \in \mathcal{A}_q$, according to:

$$\text{Eval-IA}(q, \kappa) = \sum_{a \in \mathcal{A}_q} p(a|q) \text{Eval}(a, \kappa), \tag{3.25}$$

where $p(a|q)$ is the probability of observing the aspect a given the query q, and Eval(a, κ) is computed by assuming that a is the only relevant aspect of q.

[14] Agrawal et al. (2009) use "intent" in the sense of "information need". Throughout this thesis, we adopt the traditional definition of "intent" as a *property* of an information need (e.g., informational, navigational), in the sense proposed by Broder (2002) and Rose & Levinson (2004), and instead generally refer to the information needs underlying a query as "aspects".

An alternative to the intent-aware framework of Agrawal et al. (2009) was proposed by Sakai et al. (2010), as an extension to traditional metrics based upon graded relevance, such as discounted cumulative gain (DCG; Järvelin & Kekäläinen, 2002), described in Equation (2.51). In particular, instead of computing the expected *value* of one such metric across each of the multiple aspects $\mathcal{A}_q$ underlying the query q, as in Equation (3.25), they proposed to extend this metric to leverage the expected *gain* over multiple aspects—as opposed to the raw gain with respect to the query q only. The introduced family of diversity metrics, denoted "D" metrics, can be formalised according to:

$$\text{D-Eval}(q, \kappa) = \text{Eval}(\mathcal{A}_q, \kappa), \tag{3.26}$$

where $\text{Eval}(\mathcal{A}_q, \kappa)$ denotes a traditional graded relevance metric, with the gain of the i-th document computed by aggregating the aspect-specific gains $g_{i|a}$, according to $g_i = \sum_{a \in \mathcal{A}_q} p(a|q)\, g_{i|a}$. One basic advantage of this framework over the intent-aware framework of Agrawal et al. (2009) is that the metric $\text{Eval}(\mathcal{A}_q, \kappa)$ is computed for a single rather than for multiple separate rankings.

A limitation of both the IA and the D evaluation frameworks is that they do not enforce a high coverage of multiple query aspects by design. As a result, some metrics generated by these frameworks, such as DCG-IA (Agrawal et al., 2009) or D-DCG (Sakai et al., 2010), may completely ignore aspects with a low probability $p(a|q)$. In the extreme case, these metrics may end up maximally rewarding a ranking that covers only a single yet dominant aspect. In order to overcome this limitation, Sakai et al. (2010) proposed to linearly interpolate a D metric with sub-topic recall (SR), defined in Equation (3.24). The resulting metric, which they called a "D♯" metric, can be defined according to:

$$\text{D}\sharp\text{-Eval}(q, \kappa) = \gamma\, \text{SR}(q, \kappa) + (1 - \gamma)\text{D-Eval}(q, \kappa), \tag{3.27}$$

where the parameter γ controls the balance between the $\text{SR}(q, \kappa)$ and $\text{D-Eval}(q, \kappa)$ metrics. Typically, this parameter is set as $\gamma = 0.5$, as it was shown to have little impact in the final value of $\text{D}\sharp\text{-Eval}(q, \kappa)$, primarily because $\text{SR}(q, \kappa)$ and $\text{D-Eval}(q, \kappa)$ are highly correlated with each other (Sakai et al., 2010).

Another option to enforce the coverage of multiple aspects is to instantiate either the IA or the D framework by computing the expected value (for IA metrics) or the expected gain (for D metrics) of a *cascade* metric (Clarke et al., 2011a). As discussed in Section 2.3.3, cascade metrics penalise redundancy, by modelling the behaviour of a user who stops inspecting the ranking once a relevant document is observed (Craswell et al., 2008). As an indirect result, these metrics encourage the coverage of multiple, non-redundant query aspects. One such metric is expected reciprocal rank (ERR; Chapelle et al., 2009), described in Equation (2.53). This metric can be extended into its intent-aware counterpart, ERR-IA (Chapelle et al., 2011), according to:

$$\text{ERR-IA}(q, \kappa) = \sum_{a \in \mathcal{A}_q} p(a|q) \text{ERR}(a, \kappa), \tag{3.28}$$

where $\text{ERR}(a, \kappa)$ is computed separately for each aspect $a \in \mathcal{A}_q$, under the assumption that none of the other query aspects is of interest.

Instantiations of the D framework using cascade metrics are also possible. For instance, Clarke et al. (2008) proposed to extend the traditional discounted cumulative gain (DCG) metric (Järvelin & Kekäläinen, 2002), described in Equation (2.51), with the gain at a given ranking position defined in order to reward a high coverage of the query aspects while penalising excessive redundancy with respect to the aspects covered by documents at higher ranks. More precisely, they introduced the α-DCG metric according to:

$$\alpha\text{-DCG}(q, \kappa) = \sum_{i=1}^{\kappa} \frac{\sum_{a \in \mathcal{A}_q} g_{i|a} (1 - \alpha)^{\sum_{j=1}^{i-1} g_{j|a}}}{\log_2(i + 1)}, \tag{3.29}$$

where $g_{i|a}$ is the (binary) relevance grade of the i-th ranked document with respect to each query aspect $a \in \mathcal{A}_q$. As a result, $(1 - \alpha)^{\sum_{j=1}^{i-1} g_{j|a}}$ penalises redundancy by diminishing the value of covering the aspect a, according to how much this aspect is already covered by the documents ranked ahead of the i-th document. The parameter $\alpha \in [0, 1)$ controls the amount of penalisation: $\alpha \to 1$ results in the maximum penalisation, whereas $\alpha = 0$ reduces to the standard DCG, with the number of covered aspects $\sum_{a \in \mathcal{A}_q} g_{i|a}$ used as the gain at rank i.

Extended metrics have also been proposed in recent years, accounting for dimensions of the diversification problem not addressed by the metrics described thus far. For instance, Clarke et al. (2009b) proposed a metric that explicitly distinguishes between aspects related to different interpretations of the user's query. Their basic intuition was that, while a user may be interested in multiple aspects of a given interpretation, only one such interpretation should be of interest. To exploit this intuition, they extended the rank-biased precision (RBP) metric (Moffat & Zobel, 2008), described in Equation (2.52), with a discount factor that penalises redundancy, similarly to α-DCG (Clarke et al., 2008). The resulting metric, novelty- and rank-biased precision (NRBP), was defined as:

$$\text{NRBP}(q, \kappa) = \frac{(1-(1-\alpha)\beta)}{\beta} \sum_{i=1}^{\kappa} \beta^i \sum_{\varphi \in \Omega_q} \frac{p(\varphi|q)}{|\mathcal{A}_\varphi|} \sum_{a \in \mathcal{A}_\varphi} g_{i|a} (1-\alpha)^{\sum_{j=1}^{i-1} g_{j|a}}, \quad (3.30)$$

where Ω_q is the set of possible interpretations of the query q, and $\mathcal{A}_\varphi$ is the set of aspects associated with each interpretation $\varphi \in \Omega_q$, in which case $g_{i|a}$ denotes the (binary) relevance grade of the i-th document with respect to the aspect $a \in \mathcal{A}_\varphi$. Interpretations follow a non-uniform distribution $p(\varphi|q)$, whereas the distribution of aspects for a given interpretation is assumed to be uniform. Analogously to α-DCG in Equation (3.29), $(1-\alpha)^{\sum_{j=1}^{i-1} g_{j|a}}$ penalises the coverage of already well covered interpretation-aspect pairs, with the parameter α controlling the amount of penalisation. The extra parameter β models users with different patience levels, similarly to the standard RBP metric in Equation (2.52).

Sakai (2012) proposed to extend the IA and D frameworks, in order to account for the *intent* of different aspects. For the extended D framework, he computed the gain at rank i by distinguishing between informational and navigational aspects, according to $g_i = \sum_{a \in \mathcal{A}_q} p(a|q)\, g_{i|a} (1 - \mathbf{1}_{\mathcal{A}_q^{\text{nav}}}(a)\, \mathbf{1}_{\cup_{j=1}^{i-1} \mathcal{A}_{d_j}}(a))$, where the indicator functions $\mathbf{1}_{\mathcal{A}_q^{\text{nav}}}(a)$ and $\mathbf{1}_{\cup_{j=1}^{i-1} \mathcal{A}_{d_j}}(a)$ denote whether the aspect $a \in \mathcal{A}_q$ is navigational and whether it is covered by any document ranked ahead of the i-th. His assumption was that redundancy should be penalised for navigational aspects, but not for informational ones. An analogous extension was proposed for the IA framework, by interpolating the expected value of informational- and navigational-oriented metrics over the corresponding subsets of aspects.

In addition to developing diversity evaluation metrics, much effort has been invested in validating such metrics. For instance, Clarke et al. (2011a) analysed the *discriminative power* of diversity metrics, a property that reflects the extent to which a metric can distinguish between pairs of rankings. Using the runs submitted to the TREC 2009 Web track (Clarke et al., 2009a), they observed that sub-topic recall (Equation (3.24)) has the highest discriminative power compared to the other considered diversity metrics. Intent-aware and cascade metrics, on the other hand, showed a discriminative power inferior to that observed for average precision (Equation (2.50)), a relevance-oriented metric.

Ashkan & Clarke (2011) analysed the *informativeness* of diversity metrics, which reflects the extent to which a metric predicts the actual distribution of relevant documents. Using the maximum entropy method to estimate the most plausible relevance distribution according to a given metric (Aslam et al., 2005), they found that intent-aware cascade metrics (which reward coverage and novelty) are more informative than their pure cascade counterpart (which only rewards novelty), with ERR-IA (Chapelle et al., 2011), described in Equation (3.28), showing the highest informativeness among all considered metrics.

Sanderson et al. (2010) investigated the *predictive power* of diversity metrics, in terms of the extent to which these metrics correlate with the behaviour of actual users. In their study, 296 subjects were hired through crowdsourcing to express their preference between pairs of runs submitted to the TREC 2009 Web track (Clarke et al., 2009a). The runs in each pair were also evaluated according to multiple diversity metrics. Their analysis showed a high agreement between the prediction of several diversity metrics and the users' preferences, with no significant difference in predictive power between the considered metrics.

Carterette (2009) analysed the *optimality* of the normalisation component of cascade metrics. In particular, producing an ideal ranking for normalising such metrics is an NP-hard problem, as discussed in Section 3.2.2. Since the ideal ranking is typically computed using the greedy approximation in Algorithm 3.1, a natural question is whether the produced evaluation scores are affected by a sub-optimal normalisation. Fortunately, an analysis of real and simulated topic sets and aspect relevance assessments showed that the greedy and optimal evaluation normalisations agree in 93% and 85% of the cases, respectively.

3.5 Summary

This chapter introduced the search result diversification problem as a departure from the traditional view of ranking as the problem of satisfying a single information need expressed by the user's query, which was the focus of Chapter 2.

In Section 3.1, we described several studies that quantified the occurrence of ambiguous queries in real web search logs, as a motivation for diversifying the search results. In Section 3.2, we discussed the simplifying assumptions underlying traditional probabilistic ranking approaches and the limitation of such assumptions in a real search scenario. This discussion led to the formal definition of the diversification problem and the analysis of its complexity. In Section 3.3, we described the most prominent diversification approaches in the literature, organised according to the complementary dimensions of diversification strategy and aspect representation. Lastly, in Section 3.4, we extended the discussion in Section 2.3 with an emphasis on the evaluation of diversification effectiveness, including a description of the existing evaluation benchmarks and metrics.

As a complement to Chapter 2, this chapter consolidates our account of the related literature on web search ranking, and particularly on diversity-oriented ranking. In the next chapter, we will introduce a novel framework for search result diversification, which exploits the strengths and weaknesses of past research in order to deliver a flexible and effective solution for diversifying the search results.

Chapter 4

The xQuAD Framework

Several approaches have been recently proposed to diversify the documents retrieved for an ambiguous or underspecified query. In common, we argue that none of these approaches addresses the multiple information needs underlying a query in a principled manner. As discussed in Section 3.3, on the one hand, implicit diversification approaches rely on an aspect representation derived from the retrieved documents, as opposed to the query or the possible information needs that it represents. On the other hand, existing explicit approaches rely on arbitrary surrogates or on heuristics to exploit multiple information needs.

In this thesis, we claim that an effective diversification should be explicitly driven by the perspective of the search users, as opposed to the perspective of the retrieved documents. Moreover, such an explicit representation should reflect the multiple information needs that may have motivated the query (Spärck-Jones et al., 2007). In order to formalise this view, we propose a probabilistic objective for search result diversification, which is at the core of the E**x**plicit **Qu**ery **A**spect **D**iversification (xQuAD) framework introduced in this thesis.

The remainder of this chapter describes the xQuAD framework. In particular, Section 4.1 discusses our view towards a user-driven diversification and the requirements involved in pursuing this view. Section 4.2 formalises xQuAD's probabilistic ranking objective, which fulfils the identified requirements in a principled yet practical manner. A complete example of the operation of the proposed framework is provided in Section 4.3. Lastly, a parallel to approaches that inspired the development of xQuAD is drawn in Section 4.4.

4.1 User-driven Diversification

Early diversification approaches typically built an implicit representation of the query aspects based upon some property of the retrieved documents, such as their raw content (Carbonell & Goldstein, 1998), their language models (Zhai et al., 2003), their relevance scores with respect to the initial query (Wang & Zhu, 2009; Rafiei et al., 2010), or their coverage of latent topics (Carterette & Chandar, 2009) or clusters (He et al., 2011; Gil-Costa et al., 2011, 2013). As illustrated in Figure 4.1, these approaches differ from more recent ones that derive an explicit aspect representation driven by the query itself. On the other hand, existing explicit approaches either rely on arbitrary properties of the query, such as its classification according to a fixed taxonomy (Agrawal et al., 2009), or on heuristic diversification strategies, aimed at achieving a proportional coverage of multiple query aspects in the ranking (Radlinski & Dumais, 2006; Capannini et al., 2011).

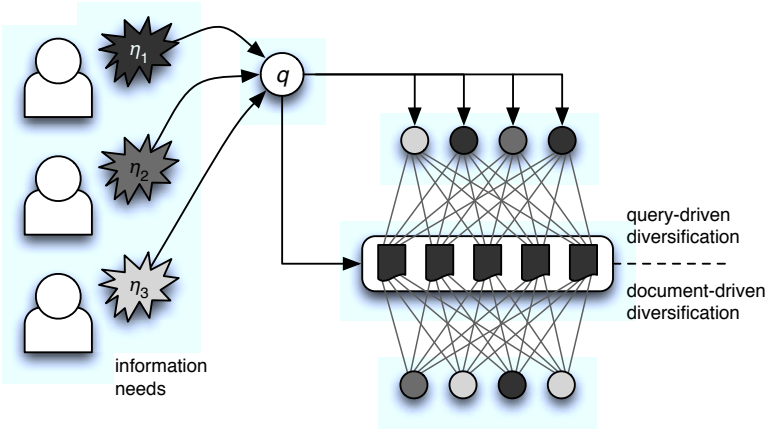


Figure 4.1: Query- vs. document-driven diversification.

We argue that the existing approaches have three key limitations:

L1. The ranking produced by a document-driven approach can be only as diverse as the aspects identified from the documents retrieved for the initial query, which may be biased (Mowshowitz & Kawaguchi, 2002). As a result, important aspects (from the user population perspective) may be overlooked simply because they are not well represented among the initial documents; conversely, marginally important aspects may be overemphasised.

L2. The query aspects identified arbitrarily based on either document or query properties are a loose surrogate for the actual information needs that may have motivated different users to issue the query in the first place. For instance, documents that cover different topics or categories—or documents that are just dissimilar from each other—can feasibly meet the same information need, in which case they would be deemed redundant.

L3. Heuristic ranking strategies may not cater for all dimensions of the diversification problem. For instance, as discussed in Section 3.3.2, ensuring a proportional coverage of multiple aspects is arguably ineffective if these aspects do not represent likely information needs; even when the likelihood of different aspects is appropriately estimated, aiming for a proportional coverage regardless of the incurred redundancy voids the approximation guarantees known for this problem (Nemhauser et al., 1978; Feige, 1998).

In this thesis, we overcome limitation L1 by adopting an explicit aspect representation, which is driven by the query as opposed to the retrieved documents. In turn, limitation L2 is overcome by ensuring that this representation is meaningfully driven towards modelling multiple users' information needs, rather than any arbitrarily defined query properties. Finally, in order to overcome limitation L3, we propose a probabilistic framework that accommodates the different dimensions of the search result diversification problem in a principled yet practical manner. In particular, the proposed framework should account for the overall coverage of each retrieved document with respect to the identified information needs, so as to rank highly diverse documents first. Moreover, it should account for how well each information need is covered by the other retrieved documents, so as to avoid promoting redundant documents. Additionally, the framework should be able to infer how much emphasis should be placed on each of the identified information needs, since there may be dozens of possible information needs underlying the query. Finally, since not all queries are equally ambiguous, the framework should also cater for the ambiguity levels of different queries, so as to infer how much to diversify the retrieved documents on a per-query basis. Our proposed framework, which fulfils all the above requirements, is introduced in the next section.

4.2 Explicit Query Aspect Diversification

The diversification problem, formally defined in Section 3.2.1, can be naturally stated as a trade-off between finding relevant and diverse information:

> Given an initial ranking $\mathcal{R}_q$ produced for a query q, find a re-ranking $\mathcal{D}_q$ that has (1) the maximum *relevance* to q, and (2) the maximum *diversity* with respect to the different aspects underlying q.

As discussed in Section 3.2.2, this bi-criterion optimisation problem can be reduced from the maximum coverage problem (Hochbaum, 1997), which makes it NP-hard (Agrawal et al., 2009). Fortunately, there is a well-known greedy approximation to this problem, as described in Algorithm 3.1, which forms the basis of most of the approaches to search result diversification presented in Section 3.3, and is also the basis for our proposed diversification framework.

In order to solve this optimisation problem, we introduce the E**x**plicit **Qu**ery **A**spect **D**iversification (xQuAD) framework. In particular, inspired by Spärck-Jones et al. (2007), we argue that an ambiguous query should be seen as representing an ensemble of possible information needs. Accordingly, within xQuAD, we model an ambiguous query as comprising a set of *sub-queries*, with each sub-query representing one of the possible information needs underlying the initial query. While different sub-query instantiations are certainly possible, in this thesis, we adopt a keyword-based representation. As we will show in Chapter 6, not only is this representation consistent with the one adopted for the initial query, but it also enables the exploitation of past users' queries as effective representations of multiple information needs. Moreover, such a representation allows xQuAD to tackle search result diversification as an optimisation of the expected relevance of a ranking in light of multiple needs. As a result, our framework can directly leverage a plethora of traditional ranking approaches, such as those introduced in Chapter 2, as we will demonstrate in Chapters 7 and 8. Lastly, by recognising that different queries may have different levels of ambiguity, we explicitly model the trade-off between promoting relevance and diversity within xQuAD, as will be discussed in Chapter 9. In the remainder of this section, we describe a probabilistic formulation that accommodates all these characteristics into a principled ranking objective for search result diversification.

4.2.1 Probabilistic Objective

As limited representations of information needs and information items, respectively, queries and documents naturally incur an uncertainty to the estimation of relevance. By adding to these the representation of multiple information needs, the estimation of diversity exacerbates the problem. In order to leverage an appropriate groundwork for reasoning under uncertainty, we devise a ranking objective for search result diversification in light of probability theory (Good, 1950).

Recalling the greedy approximation in Algorithm 3.1, given a query q and a ranking $\mathcal{R}_q$ of documents retrieved for this query, our goal is to iteratively build a new ranking $\mathcal{D}_q$, with $|\mathcal{D}_q| \leq \tau$, by selecting, at each iteration, the highest scored document $d \in \mathcal{R}_q \setminus \mathcal{D}_q$. To this end, we devise xQuAD's scoring function according to the following probability mixture model:

$$f_{\text{xQuAD}}(q, d, \mathcal{D}_q) = (1 - \lambda)\, p(d|q) + \lambda\, p(d, \bar{\mathcal{D}}_q|q), \tag{4.1}$$

where $p(d|q)$ models the probability of observing the document d given the query q, and $p(d, \bar{\mathcal{D}}_q|q)$ models the probability of observing d but *none* of the documents already in $\mathcal{D}_q$, selected in previous iterations, given q. These probabilities can be interpreted as estimations of the *relevance* and the *diversity* of d, respectively, with the parameter λ controlling the balance between the two.

The probability of relevance, $p(d|q)$, is defined in general terms, without any assumption regarding the underlying statistical mechanism used for estimation. In fact, any ranking approach can be used for this estimation, including the probabilistic ranking approaches as well as the machine-learned approaches introduced in Section 2.2, provided that they produce probabilistic scores. In turn, the probability of diversity, $p(d, \bar{\mathcal{D}}_q|q)$, models the contribution of a document d towards answering the query q, when d is provided *jointly with* the already selected documents in $\mathcal{D}_q$, which are assumed to be non-relevant. In practice, this formulation models the marginal utility of the document d in light of the documents $\mathcal{D}_q$, selected in the previous iterations of the greedy algorithm. As a result, maximising the probability $p(d, \bar{\mathcal{D}}_q|q)$ increases the chance that *at least one* relevant document is retrieved in response to the query, even when different users have different perceptions of this relevance (Sanner et al., 2011).

While estimating $p(d|q)$ is comparatively simpler, the estimation of $p(d, \bar{\mathcal{D}}_q|q)$ requires further development. To this end, it is useful to consider a sample space comprising features (e.g., terms) representing the information carried by the documents in $\mathcal{R}_q$, initially retrieved for q. As a result, d, $\mathcal{D}_q$, and q can be seen as sets of such features or, equivalently, events in this sample space. In order to derive $p(d, \bar{\mathcal{D}}_q|q)$, we further partition the sample space into a set of pairwise disjoint sub-queries $\mathcal{S}_q = \{s_1, s_2, \cdots, s_k\}$, with each sub-query $s \in \mathcal{S}_q$ representing one of the possible information needs underlying q. The resulting probability space is illustrated by the Venn diagram in Figure 4.2 for $k = 4$ sub-queries.

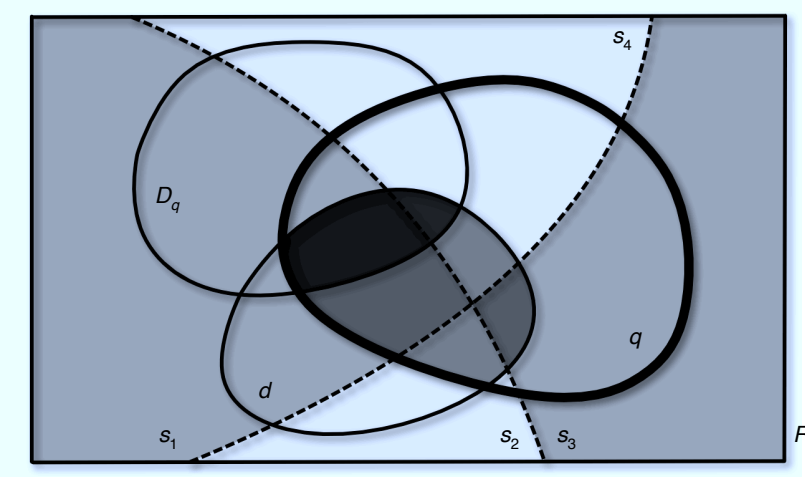


Figure 4.2: Sample space partitioned by sub-queries.

In Figure 4.2, we can identify the three events of interest, denoting the observation of the query q, the document d, and the already selected documents in $\mathcal{D}_q$. The thicker line in the figure restricts the sample space given the observation of q. As a result, the intersection between this region and the region covered by the observation of a document can be seen as a measure of the probability that the document is relevant to the query. In particular, the intersection between the events d and q is highlighted in different shades: the darkest shade denotes the information represented by d that is also covered by the documents already selected in $\mathcal{D}_q$; the lighter shades denote the novel information covered by document d, split across the considered sub-queries. Our goal is then to estimate the probability associated with the event $(d \setminus \mathcal{D}_q) \cap q$ or, equivalently, $p(d, \bar{\mathcal{D}}_q|q)$.

After defining our target probability space, we can derive the probability of diversity, $p(d, \bar{\mathcal{D}}_q|q)$, in a series of steps, according to:

$$p(d, \bar{\mathcal{D}}_q|q) = \sum_{s \in \mathcal{S}_q} p(d, \bar{\mathcal{D}}_q, s|q) \tag{4.2}$$

$$= \sum_{s \in \mathcal{S}_q} p(s|q)\, p(d, \bar{\mathcal{D}}_q|q, s) \tag{4.3}$$

$$\approx \sum_{s \in \mathcal{S}_q} p(s|q)\, p(d|q, s)\, p(\bar{\mathcal{D}}_q|q, s) \tag{4.4}$$

$$\approx \sum_{s \in \mathcal{S}_q} p(s|q)\, p(d|q, s) \prod_{d_j \in \mathcal{D}_q} p(\bar{d}_j|q, s) \tag{4.5}$$

$$= \sum_{s \in \mathcal{S}_q} p(s|q)\, p(d|q, s) \prod_{d_j \in \mathcal{D}_q} (1 - p(d_j|q, s)). \tag{4.6}$$

In order to derive Equation (4.2), we apply the sum rule and marginalise the probability $p(d, \bar{\mathcal{D}}_q|q)$ over the sub-queries $s \in \mathcal{S}_q$. Equation (4.3) follows trivially from the product rule (Good, 1950). The resulting probability $p(s|q)$ can be seen as modelling the *importance* of the sub-query s with respect to the other sub-queries in $\mathcal{S}_q$. This notion could reflect, for instance, users' preferences or the context of their search (Clarke et al., 2008; Agrawal et al., 2009).

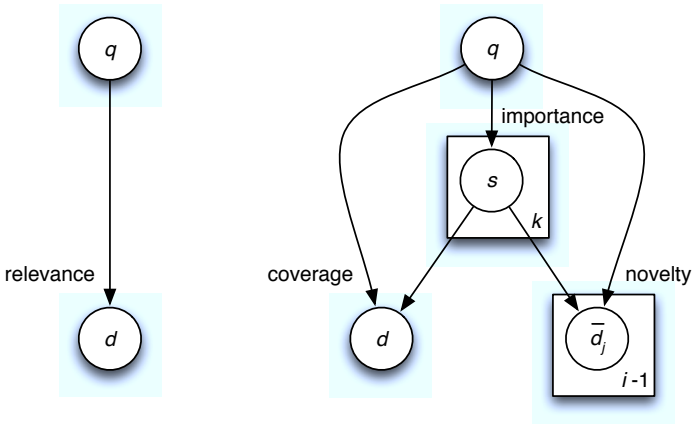
In order to derive $p(d, \bar{\mathcal{D}}_q|q, s)$ in Equation (4.3), we assume that the observation of the document d is independent of the observation of the documents already selected in $\mathcal{D}_q$ (and, by extension, of $\bar{\mathcal{D}}_q$), conditioned on the observation of the query q and the sub-query s. While this assumption is also present in the formulation of other diversification approaches in the literature (e.g., Agrawal et al., 2009; Carterette & Chandar, 2009), in reality, the knowledge of the documents that have already been selected affects the selection of the next document. On the other hand, this knowledge affects *all* candidate documents $d \in \mathcal{R}_q \setminus \mathcal{D}_q$ equally, since $\mathcal{D}_q$ is fixed at each iteration. As a result, it seems plausible to refactor the probability $p(d, \bar{\mathcal{D}}_q|q, s)$ into a more tractable form. Note, however, that such a refactoring does not at all imply that redundancy is ignored in our formulation. Instead, it results in separate models of the *coverage* of each document d with respect to the sub-query s, i.e., $p(d|q, s)$, and its *novelty* in light of how poorly this sub-query is covered by the already selected documents in $\mathcal{D}_q$, i.e., $p(\bar{\mathcal{D}}_q|q, s)$.

The conditional independence assumption in Equation (4.4) has a subtle but important implication: it turns the computation of novelty from a direct comparison between documents into an estimation of the *marginal utility* of any document satisfying each sub-query. In other words, instead of comparing a document d to all documents already selected in $\mathcal{D}_q$, as implicit novelty-based diversification approaches would do (see Section 3.3.1), we estimate the utility of any document satisfying the sub-query s, as the probability that none of the already selected documents in $\mathcal{D}_q$ satisfy this sub-query. Although we achieve the same goal of promoting novelty, we do so in a much more efficient way. In particular, our approach does not require looking up all the terms contained in all documents from the initial ranking $\mathcal{R}_q$, so as to enable their direct comparison. Instead, we just need to update the novelty estimation of a given sub-query, based on the estimation of how much this sub-query is already covered by the documents in $\mathcal{D}_q$. In contrast to implicit approaches, this estimation only incurs a few additional inverted file lookups for the documents matching each of the sub-query terms.

In order to derive $p(\bar{\mathcal{D}}_q|q,s)$ in Equation (4.4), we make a second conditional independence assumption. In particular, we assume that the documents already selected in $\mathcal{D}_q$ are independently relevant to the sub-query s. This assumption seems reasonable, since novelty is estimated as the probability of the entire set $\mathcal{D}_q$ (as opposed to any particular document in $\mathcal{D}_q$) not satisfying s. Lastly, for convenience, Equation (4.5) is derived into Equation (4.6), by replacing $p(\bar{d}_j|q,s)$ with its complementary probability, subtracted from 1, i.e., $1 - p(d_j|q,s)$. It is interesting to observe that this simple algebraic transformation emphasises the similarity of the probabilities $p(d|q,s)$ and $p(d_j|q,s)$, which must be estimated as part of the computation of each document's coverage and novelty, respectively.

The derivation of xQuAD's relevance and diversity components in Equation (4.1) is further illustrated by the graphical models in Figures 4.3(a) and (b), respectively. Finally, by replacing Equation (4.6) into (4.1), the final diversification objective of xQuAD can be expressed according to:

$$f_{\text{xQuAD}}(q,d,\mathcal{D}_q) = (1-\lambda)\, p(d|q) + \lambda \sum_{s \in \mathcal{S}_q} p(s|q)\, p(d|q,s) \prod_{d_j \in \mathcal{D}_q} (1 - p(d_j|q,s)). \quad (4.7)$$



(a) $p(d|q)$ (b) $\sum_{s \in \mathcal{S}_q} p(s|q)\, p(d|q,s) \prod_{d_j \in \mathcal{D}_q} p(\bar{d}_j|q,s)$

Figure 4.3: xQuAD's graphical models of (a) relevance and (b) diversity, which are mixed for the selection of a document $d \in \mathcal{R}_q \setminus \mathcal{D}_q$ at the i-th iteration of Algorithm 3.1.

4.2.2 Framework Components

Several dimensions of the diversification problem are naturally modelled as individual probabilities in Equation (4.7). In practice, these probabilities are estimated by different components of the xQuAD framework, namely:

1. document relevance, $p(d|q)$;
2. document diversity, $p(d, \bar{\mathcal{D}}_q|q), \forall s \in \mathcal{S}_q$:
   (a) sub-query importance, $p(s|q)$;
   (b) document coverage, $p(d|q,s)$;
   (c) document novelty, $\prod_{d_j \in \mathcal{D}_q} 1 - p(d_j|q,s)$.

Further components of the framework include the actual mechanism that generates the set of sub-queries $\mathcal{S}_q$, as well as the mechanism that computes the diversification trade-off λ for a given query q. Each of these components can be instantiated in a variety of ways, essentially generating different diversification models within the xQuAD framework. As we will show in Chapters 5 through 9, not only do these components add to the flexibility of xQuAD, but they also provide multiple opportunities to devise effective diversification models.

4.3 Example Application

In order to illustrate the execution of xQuAD, we introduce a toy example. In particular, consider an unidentified user who issues the query $q =$ "james bond", for which a search engine retrieves a ranking of documents $\mathcal{R}_q = \{d_1, d_2, d_3, d_4, d_5\}$. Also consider that, based upon an analysis of its log of the interactions of previous users who issued the query q, the search engine infers that this query is reformulated 50% of the time. In addition, of all reformulations of q, 60% reflect an underlying information need for "films", with the remaining 40% denoting an information need for "books". As a result, these inferred needs are represented as two sub-queries, i.e., $\mathcal{S}_q = \{s_1, s_2\}$, with $s_1 =$ "films" and $s_2 =$ "books".

The execution of xQuAD towards iteratively producing a diverse ranking $\mathcal{D}_q \subseteq \mathcal{R}_q$ can be illustrated in terms of basic matrix operations. In particular, let $\mathbf{R}$ be a 5×1 matrix representing the distribution of *relevance* probabilities $p(d|q)$, for all $d \in \mathcal{R}_q$. Similarly, let $\mathbf{C}$ be a 5×2 matrix representing the distribution of *coverage* probabilities $p(d|q, s)$, for all $d \in \mathcal{R}_q$ and all $s \in \mathcal{S}_q$. Lastly, let $\mathbf{I}$ be a 2×1 matrix representing the distribution of *importance* probabilities $p(s|q)$, for all $s \in \mathcal{S}_q$. In line with the scenario described above, a hypothetical definition of these matrices could be given according to:

$$\mathbf{R} = \begin{bmatrix} 0.70 \\ 0.50 \\ 0.30 \\ 0.20 \\ 0.10 \end{bmatrix}, \qquad \mathbf{C} = \begin{bmatrix} 0.30 & 0.40 \\ 0.70 & 0.60 \\ 0.20 & 0.30 \\ 0.70 & 0.80 \\ 0.40 & 0.20 \end{bmatrix}, \qquad \mathbf{I} = \begin{bmatrix} 0.60 \\ 0.40 \end{bmatrix}, \tag{4.8}$$

where, for instance, $\mathbf{R}_{21} = 0.50$ is the probability of relevance of d_2; $\mathbf{C}_{21} = 0.70$ and $\mathbf{C}_{22} = 0.60$ denote the coverage probabilities of this document with respect to the sub-queries s_1 and s_2, respectively; in turn, the importance of these sub-queries is given by $\mathbf{I}_{11} = 0.60$ and $\mathbf{I}_{21} = 0.40$, respectively. Still in line with the above example, assuming that the number of times a query is reformulated provides a rough indication of the ambiguity of this query, we can further infer that $\lambda = 0.5$ provides a reasonable setting for effectively balancing the trade-off between promoting relevance and diversity in Equation (4.1).

With the example definitions of the $\mathbf{R}$, $\mathbf{C}$, and $\mathbf{I}$ matrices, we can illustrate the computation of xQuAD's objective in Equation (4.1) according to:

$$\mathbf{X}^{(i)} = (1-\lambda)\mathbf{R} + \lambda\mathbf{D}^{(i-1)}, \tag{4.9}$$

where $\mathbf{X}^{(i)}$ denotes the distribution of probabilities computed by xQuAD at the i-th iteration, as a linear mixture of the distributions of relevance and diversity probabilities, $\mathbf{R}$ and $\mathbf{D}^{(i-1)}$, respectively. The latter distribution is further defined as $\mathbf{D}^{(i-1)} = \mathbf{C}\mathbf{N}^{(i-1)}$, where $\mathbf{N}^{(i-1)}$ is a 2×1 matrix, denoting the *novelty* of any document satisfying each of the sub-queries s_1 and s_2. This matrix is initialised as $\mathbf{N}^{(0)} = \mathbf{I}$, indicating that satisfying either s_1 or s_2 has an initial benefit proportional to the relative importance of each of these sub-queries. According to this formulation, the first iteration of xQuAD can be expressed as:

$$\mathbf{X}^{(1)} = (1-0.5)\begin{bmatrix}0.70\\0.50\\0.30\\0.20\\0.10\end{bmatrix} + 0.5\begin{bmatrix}0.30 & 0.40\\0.70 & 0.60\\0.20 & 0.30\\0.70 & 0.80\\0.40 & 0.20\end{bmatrix}\begin{bmatrix}0.60\\0.40\end{bmatrix} = \begin{bmatrix}0.52\\\mathbf{0.58}\\0.27\\0.47\\0.21\end{bmatrix}\begin{matrix}(d_1)\\\boldsymbol{(d_2)}\\(d_3)\\(d_4)\\(d_5)\end{matrix}, \tag{4.10}$$

in which case d_2 is selected as the highest scoring document.[1] Since this document covers sub-queries s_1 and s_2 with probabilities 0.70 and 0.60, respectively, the novelty of any document that covers either sub-query should be diminished proportionally to these probabilities. In general terms, letting $r = \arg\max_j \mathbf{X}^{(i)}_{j1}$ denote the index of the highest scoring document according to xQuAD at the i-th iteration, the novelty matrix $\mathbf{N}$ can be updated according to:

$$\mathbf{N}^{(i)} = \text{diag}(\mathbf{1} - \mathbf{C}_r)\mathbf{N}^{(i-1)}, \tag{4.11}$$

where $\mathbf{1}$ is a row vector composed of ones, $\mathbf{C}_r$ is the r-th row of the coverage matrix $\mathbf{C}$, corresponding to the selected document, and $\text{diag}(\mathbf{1} - \mathbf{C}_r)$ is the diagonal matrix whose diagonal entries are given by the input vector $\mathbf{1} - \mathbf{C}_r$.

[1] Note that, in contrast to most of the approaches described in Section 3.3, xQuAD does not enforce that the first selected document be the one with the highest estimated relevance $p(d|q)$. Instead, the objective function in Equation (4.7) applies consistently to all iterations.

Given the formulation in Equation (4.11), we can compute the updated novelty matrix $\mathbf{N}^{(1)}$ after the first iteration according to:

$$\mathbf{N}^{(1)} = \begin{bmatrix} 1.00 - 0.70 & 0.00 \\ 0.00 & 1.00 - 0.60 \end{bmatrix} \begin{bmatrix} 0.60 \\ 0.40 \end{bmatrix} = \begin{bmatrix} 0.18 \\ 0.16 \end{bmatrix}. \tag{4.12}$$

Fixing the highest scored document in the first iteration, d_2, and replacing the updated novelty vector from Equation (4.12) back into Equation (4.9), we can select the second most diverse document, according to:

$$\mathbf{X}^{(2)} = (1 - 0.5) \begin{bmatrix} 0.70 \\ 0.30 \\ 0.20 \\ 0.10 \end{bmatrix} + 0.5 \begin{bmatrix} 0.30 & 0.40 \\ 0.20 & 0.30 \\ 0.70 & 0.80 \\ 0.40 & 0.20 \end{bmatrix} \begin{bmatrix} 0.18 \\ 0.16 \end{bmatrix} = \begin{bmatrix} \mathbf{0.41} \\ 0.19 \\ 0.23 \\ 0.10 \end{bmatrix} \begin{matrix} \boldsymbol{(d_1)} \\ (d_3) \\ (d_4) \\ (d_5) \end{matrix}, \tag{4.13}$$

where document d_1 is selected as the next best document. By updating the novelty vector at the end of each iteration using Equation (4.11), and re-scoring the yet unselected documents using Equation (4.9) with the updated novelty estimations, we can iteratively select the next documents, according to:

$$\mathbf{X}^{(3)} = (1 - 0.5) \begin{bmatrix} 0.30 \\ 0.20 \\ 0.10 \end{bmatrix} + 0.5 \begin{bmatrix} 0.20 & 0.30 \\ 0.70 & 0.80 \\ 0.40 & 0.20 \end{bmatrix} \begin{bmatrix} 0.13 \\ 0.10 \end{bmatrix} = \begin{bmatrix} 0.17 \\ \mathbf{0.18} \\ 0.08 \end{bmatrix} \begin{matrix} (d_3) \\ \boldsymbol{(d_4)}, \\ (d_5) \end{matrix} \tag{4.14}$$

$$\mathbf{X}^{(4)} = (1 - 0.5) \begin{bmatrix} 0.30 \\ 0.10 \end{bmatrix} + 0.5 \begin{bmatrix} 0.20 & 0.30 \\ 0.40 & 0.20 \end{bmatrix} \begin{bmatrix} 0.04 \\ 0.02 \end{bmatrix} = \begin{bmatrix} \mathbf{0.16} \\ 0.06 \end{bmatrix} \begin{matrix} \boldsymbol{(d_3)} \\ (d_5) \end{matrix}, \tag{4.15}$$

$$\mathbf{X}^{(5)} = (1 - 0.5) \begin{bmatrix} 0.10 \end{bmatrix} + 0.5 \begin{bmatrix} 0.40 & 0.20 \end{bmatrix} \begin{bmatrix} 0.03 \\ 0.01 \end{bmatrix} = \begin{bmatrix} \mathbf{0.06} \end{bmatrix} \quad \boldsymbol{(d_5)}. \tag{4.16}$$

At the end of the 5-th iteration, $\mathcal{D}_q = \{d_2, d_1, d_4, d_3, d_5\}$ is selected by xQuAD as a diverse permutation of the initial ranking $\mathcal{R}_q$. Importantly, the probabilities computed by xQuAD are guaranteed to be monotonically non-increasing, since the estimations of novelty cannot increase and everything else is held fixed as the diversification progresses. As a result, the final ranking induced by the probabilities computed by xQuAD is stable across multiple iterations.

4.4 Relation to Other Approaches

The development of xQuAD aimed for an effective and general ranking objective for search result diversification, by encompassing successful features of past research in a principled manner. In particular, the explicit aspect representation adopted by xQuAD was inspired by the proportional coverage (PC) approach of Radlinski & Dumais (2006). As formalised in Equation (3.17), their approach seeks to balance the coverage of multiple reformulations of the initial query among the documents ranked in response to this query. Although query reformulations provide a meaningful alternative for representing the multiple possible information needs underlying a query as sub-queries, our framework caters for several dimensions of the diversification problem, which are not addressed by the approach of Radlinski & Dumais (2006), such as the relative importance of different sub-queries and the redundancy of covering already well covered sub-queries.

As a matter of fact, xQuAD can emulate the approach of Radlinski & Dumais (2006) as well as other coverage-based approaches, by assuming that the identified sub-queries do not lose their utility as more documents that cover these sub-queries are selected. In practice, as will be discussed in Section 8.2.2, this can be achieved by dropping xQuAD's novelty component, $p(\bar{\mathcal{D}}_q|q,s)$, from the expanded formulation in Equation (4.4). Furthermore, a proportional coverage of sub-queries, similar to the one deployed by approaches like PC (Equation (3.17)) and WPC (Equation (3.18)), can also be enforced within xQuAD, by conditioning the scoring of documents that cover a particular sub-query s on the total number of documents already covering this sub-query, such that:

$$p(d|q,s) = \begin{cases} p(d|q,s), & \text{if } \left[\sum_{d_j \in \mathcal{D}_q} \mathbf{1}(p(d_j|q,s) > 0)\right] < p(s|q)\,\tau, \\ 0, & \text{otherwise,} \end{cases} \tag{4.17}$$

where $\mathbf{1}$ is the indicator function, returning 1 if $p(d_j|q,s) > 0$ (i.e., if the document d_j covers the sub-query s), or 0 otherwise. On the right-hand side of the inequality, $p(s|q)$ and τ denote the importance of s and the diversification cutoff, respectively, in which case the product $p(s|q)\,\tau$ determines the fraction of the final ranking that should be dedicated to the sub-query s.

With respect to its diversification strategy, the xQuAD framework can be seen as a generalisation of the IA-Select approach of Agrawal et al. (2009). As defined in Equation (3.23), this hybrid approach seeks to maximise the overall utility of the ranked documents in light of the multiple categories associated with the query. In particular, Agrawal et al. (2009) proposed to approximate the marginal utility $f(c|q, \mathcal{D}_q)$ of any document covering each category c, given the query q, and the already selected documents in $\mathcal{D}_q$, according to:

$$f(c|q, \mathcal{D}_q) \approx f(c|q) \prod_{d_j \in \mathcal{D}_q} (1 - f(d_j|q, c)). \tag{4.18}$$

Contrasting Equation (4.18) with the definition of xQuAD in Equation (4.7), we note the similarity between the components in the right-hand side of Equation (4.18) with xQuAD's sub-query importance and novelty components, respectively. In particular, with xQuAD, not only do we provide a formal probabilistic argument for maximising the utility of a ranking, but we also devise this formalisation in light of an aspect representation that better reflects the multiple information needs—as opposed to multiple categories—underlying the query.

Besides formalising the notion of utility in probabilistic terms, we extend this notion to cater for queries with different levels of ambiguity, by mixing relevance and diversity estimates through the diversification trade-off λ, as described in Equation (4.1).[2] The resulting mixture is in turn inspired by the maximal marginal relevance (MMR) approach of Carbonell & Goldstein (1998), described in Section 3.3.1. As we will show in Chapter 9, our generalised formulation enables a selective diversification approach, which automatically adapts itself to diversify more or less aggressively, given the predicted ambiguity of each query. However, a fundamental difference from MMR is our adoption of an explicit aspect representation, enabling the combination of coverage and novelty into a hybrid strategy, which outperforms the pure novelty-based strategy deployed by MMR, as we will show in Chapter 8. Also note that an implicit version of xQuAD can be trivially derived by adopting a document-oriented aspect representation, e.g., by letting $\mathcal{S}_q = \mathcal{V}$, where the lexicon $\mathcal{V}$ comprises all unique terms in the underlying corpus.

[2] In fact, IA-Select can be directly instantiated by deploying xQuAD with $\lambda = 1$.

4.5 Summary

This chapter introduced a novel approach to the search result diversification problem, described in Chapter 3. The proposed **Ex**plicit **Qu**ery **A**spect **D**iversification (xQuAD) framework models multiple dimensions of the diversification problem in a principled manner, under the formalism of probability theory.

In Section 4.1, we identified three limitations of different families of related approaches from the literature, in terms of their reliance solely on the documents initially retrieved for a query, their arbitrarily defined representation of the multiple information needs underlying this query, and their heuristic ranking objectives. In order to overcome these limitations, Section 4.2 introduced the xQuAD framework with the goal of pursuing a diversification driven by the users' information needs. Besides formalising xQuAD's ranking objective in probabilistic terms, we introduced the several components that naturally emerge from this formulation. A complete example of the operation of the framework was provided in Section 4.3, where its underlying computations were defined in terms of basic matrix operations. Finally, Section 4.4 highlighted the key features of related approaches from the literature that inspired the development of xQuAD. In particular, the framework can be seen as a principled generalisation of the most prominent representatives of the three families of diversification approaches described in Section 3.3, namely, novelty-based, coverage-based, and hybrid.

At this stage, perhaps the most distinguishing feature of the xQuAD framework is its generality, as a result of modelling all dimensions of the diversification problem, as introduced in Chapter 3. An immediate advantage of such a general formulation is the possibility of instantiating each of the components of the framework in different ways, with each instantiation having the potential to contribute to an overall effective diversification performance. Experimenting with multiple such instantiations will be the goal of the next chapters. In particular, Chapter 5 will thoroughly assess the xQuAD framework by contrasting it to state-of-the-art representatives of the various families of diversification approaches described in Section 3.3. Chapter 6 will introduce a novel learning to rank approach for generating effective sub-queries, mined as query suggestions from a query log. In turn, Chapter 7 will introduce a supervised approach to predict the effectiveness

of multiple intent-aware ranking models for estimating the coverage of each document with respect to each sub-query, as well as the novelty of the document, given the sub-queries covered by the already retrieved documents. The role of novelty as a diversification strategy will be further analysed in Chapter 8. Lastly, Chapter 9 will introduce a supervised mechanism for selectively diversifying the retrieved documents, by automatically adapting the diversification trade-off given the predicted ambiguity level of each individual query.

Chapter 5

Framework Validation

As introduced in Chapter 4, the xQuAD framework provides a principled and general formulation for tackling the search result diversification problem. Indeed, different components of the framework model different dimensions of this problem, such as the identification of multiple query aspects and the estimation of the relevance of each retrieved document with respect to each identified aspect. Naturally, the effectiveness of the framework depends on the effectiveness of the particular choices for instantiating each of these components. Before introducing effective alternative instantiations for each of these components in the subsequent chapters, in this chapter, we validate the xQuAD framework as a whole, by contrasting it to the current state-of-the-art in search result diversification.

The goals of this chapter are twofold. Firstly, in Section 5.1, we introduce the basic experimental methodology that is used throughout the experimental part of this thesis, which comprises this chapter and Chapters 6 through 9. Secondly, in Section 5.2, we thoroughly validate the effectiveness of the xQuAD framework in comparison to state-of-the-art representatives of different families of diversification approaches in the literature. In addition, we break down this evaluation along the complementary dimensions of aspect representation and diversification strategy, as introduced in Section 3.3. The results of this evaluation not only attest the effectiveness of xQuAD when compared to the current state-of-the-art, but they also validate our option for a hybrid, user-driven diversification.

5.1 Experimental Methodology

The unlimited number of possible instantiations of each component of xQuAD precludes an exhaustive experimentation in this thesis. In order to conduct a thorough yet feasible investigation, in this thesis, we adopt a *fractional* factorial design (Box et al., 2005), by evaluating a limited number of instantiations (factor levels) of each framework component (factor) that are both potentially effective and feasible for a practical deployment. As part of the validation of xQuAD in this chapter, Section 5.2 investigates alternative instantiations of the document relevance component. In turn, Chapter 6 will investigate multiple instantiations of the sub-query generation and importance components, while Chapter 7 will investigate the document coverage and novelty components. A deeper look into the role of the novelty component will be the focus of Chapter 8. Lastly, Chapter 9 will investigate alternative regimes for estimating the diversification trade-off.

While different chapters of this thesis have different experimental setups, in the remainder of this section, we describe the basic experimental methodology that is common to all these chapters. In particular, Section 5.1.1 describes the test collections used in our experiments, including their associated document corpus, queries, and relevance assessments, while Section 5.1.2 describes the procedures for training and evaluating all approaches investigated in this thesis.

5.1.1 Test Collections

Our experiments are based on the evaluation paradigm provided by the TREC 2009, 2010, and 2011 Web tracks (Clarke et al., 2009a, 2010, 2011b), henceforth denoted WT09, WT10, and WT11, respectively. The TREC Web track provides test collections for the assessment of adhoc and diversity search approaches in a web setting. As a document corpus, it uses the ClueWeb09 dataset,[1] a web crawl comprising over 1.2 billion documents in different languages. In our experiments, we use two subsets of ClueWeb09, as used in TREC: the ClueWeb09 A corpus (CW09A), comprising the English portion of ClueWeb09, with 500 million documents; and the ClueWeb09 B corpus (CW09B), a subset of CW09A with 50

[1] http://boston.lti.cs.cmu.edu/Data/clueweb09/

million documents, aimed to represent the first tier of a commercial search engine index (Santos et al., 2011b). We index these corpora using the Terrier IR platform[2] (Ounis et al., 2006; Santos et al., 2011g; Macdonald et al., 2012a), after applying Porter's weak stemmer (Porter, 1980) and without removing stopwords.

As of 2011,[3] the TREC Web track provides a total of 150 queries, sampled from the query log of a commercial search engine. In our experiments, we discard the queries numbered 20, 95, 100, 112, and 143, as they do not have any document in the ClueWeb09 B corpus judged relevant for either the adhoc or the diversity task. The statistics of the resulting test collections with a total of 145 queries are provided in Table 5.1. As described in Section 3.4.1, for each query, TREC assessors identified multiple sub-topics, representing different aspects of the query, with relevance assessments conducted at the sub-topic level (Clarke et al., 2009a, 2010, 2011b). In some of our experiments, these sub-topics will be used as an oracle aspect representation. While alternative representations will be proposed and investigated in both Section 5.2 and Chapter 6, this oracle provides a controlled environment for evaluating the effectiveness of different diversification approaches while isolating the impact of any particular aspect representation.

Table 5.1: Statistics of the test collections used in this thesis. Relevance assessment figures are broken down by corpus (CW09A or CW09B) and task (adhoc or diversity).

			WT09	WT10	WT11
		#queries	49	48	48
		#sub-topics	228	194	158
CW09A	adhoc	#judged	23,205	23,898	18,362
		#relevant	6,858	5,233	3,157
	diversity	#judged	25,833	[4]6,553	1,9381
		#relevant	4,895	6,553	5,030
CW09B	adhoc	#judged	12,859	15,130	12,132
		#relevant	4,002	3,090	1,662
	diversity	#judged	14,951	[4]3,960	12,599
		#relevant	3,026	3,960	2,764

[2]http://terrier.org
[3]The TREC 2012 Web track is ongoing at the time of writing.
[4]The total number of judged documents for WT10 is not available.

5.1.2 Training and Evaluation

Several supervised machine learning approaches—including learning to rank, classification, and regression approaches—are deployed in this thesis, which require some form of training data. A natural direction for producing training examples from the test collections described in Section 5.1.1 is to partition the available queries into training and test sets. In our experiments, two alternative regimes are considered. In particular, the experiments in this chapter as well as those in Chapters 6 and 9 deploy a cross-validation regime, mixing together the available queries and randomly splitting these queries into multiple folds. In each of the cross-validation rounds, we organise the available queries into training (60%), validation (20%), and test (20%) queries. As discussed in Section 2.2.3.1, the use of validation data reduces the possibility that the learned parameters are overfitted to the training data. Our second training regime is used for the experiments in Chapters 7 and 8, where we deploy a cross-year validation, with the available queries split into year-oriented folds, as opposed to randomly. To ensure a fair evaluation with a complete separation from training and test, all results in this thesis are reported as an average across the test queries from the different cross-validation (or cross-year) rounds. A breakdown of the corpus, queries, and training regime used in each experimental chapter is provided in Table 5.2.

Table 5.2: Corpus, queries, and training regime used in each chapter.

	Chapter 5	Chapter 6	Chapter 7	Chapter 8	Chapter 9
Corpus	CW09B	CW09A	CW09B	CW09B	CW09B
Queries	WT09 WT10 WT11	WT09 WT10 WT11	WT09 WT10	WT09 WT10	WT09
Training	5-fold cross-valid.	5-fold cross-valid.	2-fold cross-year	2-fold cross-year	5-fold cross-valid.

To evaluate the various approaches investigated in this thesis, we deploy the two primary metrics used in the diversity task of the TREC Web track (Clarke et al., 2009a, 2010, 2011b): ERR-IA (Equation (3.28)) and α-nDCG (Equation (3.29)). As discussed in Section 3.4.2, these metrics implement a cascade

model (Craswell et al., 2008), which penalises redundancy across multiple query aspects, by assuming a diminishing probability that the users will continue to examine the ranking once they find relevant information (Clarke et al., 2011a). Following the standard TREC setting, both metrics are reported at rank 20, reflecting web searchers' interest for documents at early ranks (Jansen et al., 1998).

Lastly, in order to ensure that our findings are not a mere reflection of chance, all results reported in this thesis are validated statistically. As a statistical hypothesis test, we use Student's t-test to contrast pairs of ranking approaches (Sanderson & Zobel, 2005; Smucker et al., 2007). In particular, throughout this thesis, we use the symbols $\triangle$ ($\triangledown$) and $\blacktriangle$ ($\blacktriangledown$) to denote a statistically significant increase (decrease) at the $p < 0.05$ and $p < 0.01$ levels, respectively, while the symbol $\circ$ is used to denote no significant difference. The baseline against which significance is reported will be made clear in each case. In addition, we report the number of queries negatively affected ($-$), positively affected ($+$), and unaffected ($=$) by each tested approach compared to this baseline.

5.2 Experimental Evaluation

In Chapter 4, we introduced the xQuAD framework for search result diversification, building upon two fundamental pillars: (1) an explicit query aspect representation, aimed to reflect multiple users' information needs and (2) a hybrid diversification strategy, formulated as a principled and general probabilistic objective. In this section, we thoroughly validate the xQuAD framework and the impact of these two pillars on the effectiveness of the framework as a whole. As a result, we investigate the first claim from our thesis statement:

> *"The statement of this thesis is that an effective diversification performance can be attained by explicitly representing the multiple possible information needs underlying a query as sub-queries."*

In order to address this claim, the experiments in this chapter aim to answer three main research questions:

Q1. How does xQuAD compare to the state-of-the-art?

Q2. How effective is xQuAD's diversification strategy?

Q3. How effective is xQuAD's aspect representation?

In the following, Section 5.2.1 details the specific setup that supports our investigations, while Section 5.2.2 analyses our results.

5.2.1 Experimental Setup

In addition to the general methodology adopted in all experiments of this thesis, as described in Section 5.1, in this section, we describe the specific experimental setup that underlies the investigations in this chapter.

5.2.1.1 Retrieval Baselines

The most straightforward baseline for any diversification approach is arguably a ranking approach that does not perform any diversification at all. With this mind, we evaluate the effectiveness of different diversification approaches in this chapter at re-ranking the documents retrieved by a relevance-oriented baseline. In particular, we consider two such baselines. The first of these is the DPH model (Amati et al., 2007). As described in Section 2.2.1.3, DPH is a non-parametric ranking model from the divergence from randomness framework (Amati, 2003). As such, it provides an effective retrieval performance without requiring any training.

Besides DPH, we consider a machine-learned ranking model produced by LambdaMART (Wu et al., 2008), a state-of-the-art learning to rank algorithm. As described in Section 2.2.3.2, this listwise learning algorithm falls into the general framework of boosting (Kearns, 1988; Schapire, 1990): given some training data, the algorithm iteratively learns an ensemble of boosted regression trees, with the gradient of a standard evaluation metric used as a loss function. In order to instantiate this approach, we use nDCG@1000 (Equation (2.51)) as a loss function. As a learning sample for each query, we use the top 5,000 documents returned by DPH. This setup has been shown to be particularly effective for learning to rank for web search (Macdonald, Santos & Ounis, 2013). Lastly, as ranking features, we consider a total of 45 features commonly used in the learning to rank literature (Qin et al., 2010; Liu, 2009), including both query-dependent and

query-independent ones. These features are described in Table 5.3, along with a reference to their corresponding definition in Section 2.2.1. In particular, query-dependent features are computed separately for four different document fields, namely, title, URL, body, and anchor-text of incoming hyperlinks, as well as for the four fields combined as a full representation of the document. The exceptions are the Markov random fields (MRF; Equation (2.20)) and pBiL (Equation (2.32)) proximity features, which are only computed on the full representation. In order to compute the click likelihood (CL; Equation (2.46)) query-independent feature, we use the MSN 2006 query log,[5] a one-month log with 15 million queries submitted by US users to MSN Search (now Bing) during spring 2006.

Table 5.3: Document features used in this chapter. The top half of the table includes query-dependent features, while the bottom half includes query-independent ones.

	Feature	Description	Equation	Total
Query-dependent	CLM	Full and per-field CLM score	(2.5)	5
	BM25	Full and per-field BM25 score	(2.13)	5
	LM	Full and per-field LM score	(2.25)	5
	MRF	Full MRF score	(2.20)	1
	PL2	Full and per-field PL2 score	(2.29)	5
	DPH	Full and per-field DPH score	(2.31)	5
	pBiL	Full pBiL score	(2.32)	1
Query-independent	l_d	Full and per-field length	(2.2)	5
	UT	URL type	(2.33)	1
	UL	URL length	(2.35)	1
	ATL	Average term length	(2.36)	1
	TC	Topic cohesiveness	(2.37)	1
	SF	Stopword fraction	(2.38)	1
	SC	Stopword coverage	(2.39)	1
	TT	Table text ratio	(2.40)	1
	CR	Compression ratio	(2.41)	1
	HL	Ham (non-spam) likelihood	(2.42)	1
	ID	Indegree	(2.43)	1
	OD	Outdegree	(2.44)	1
	PR	PageRank	(2.45)	1
	CL	Click likelihood	(2.46)	1
	Grand total			45

[5] http://research.microsoft.com/en-us/um/people/nickcr/wscd09

On top of DPH and LambdaMART, we contrast the effectiveness of xQuAD to state-of-the-art representatives of the three families of diversification approaches introduced in Section 3.3. In particular, as a novelty-based approach, we use MMR (Equation (3.8)), which promotes documents with a low similarity to other documents (Carbonell & Goldstein, 1998). As a coverage-based approach, we use PC (Equation (3.17)), which enforces a proportional coverage of multiple query reformulations in the ranking (Radlinski & Dumais, 2006). Lastly, as a hybrid approach, we use IA-Select (Equation (3.23)), which maximises the marginal utility of the ranking in light of a taxonomy of categories (Agrawal et al., 2009). To ensure all approaches leverage probabilistic scores, the raw scores produced by either DPH or LambdaMART are normalised by the sum of the scores of all documents returned for the initial query and each of its identified aspects. The same score normalisation procedure is performed consistently for all approaches investigated in the remaining chapters of this thesis.

5.2.1.2 Training Procedure

While PC (Equation (3.17)) and IA-Select (Equation (3.23)) are non-parametric approaches, both MMR (Equation (3.8)) and xQuAD (Equation (4.7)) have one parameter to train, namely, the diversification trade-off λ, which controls the balance between promoting relevance or diversity. Using the standard 5-fold cross validation setup described in Section 5.1.2, we optimise λ for both approaches. To this end, we perform a simulated annealing optimisation (Kirkpatrick et al., 1983) to maximise ERR-IA@100 (Equation (3.28)) on the training queries, and use the identified λ setting on the corresponding test queries in each round.

5.2.1.3 Aspect Representations

While MMR uses an implicit aspect representation in the space of the unique terms covered by each document, both xQuAD and the other diversification baselines introduced in Section 5.2.1.1 leverage explicit aspect representations, as discussed in Section 3.3. To instantiate the latter approaches, we use three alternative explicit representations: ODP categories (DZ), Bing suggestions (BS), and the official TREC Web track sub-topics (WT). These representations are ex-

emplified in Table 5.4, while their statistics are summarised in Table 5.5 in terms of the mean number of aspects per query, the mean length (in tokens) of each aspect, and the mean overlap between each aspect and the initial query, measured as the fraction of unique query terms covered by the aspect. As a reference for comparison, the mean length of the initial queries is also shown.

Table 5.4: Example aspects for ambiguous (query #6: "*kcs*") and underspecified (query #10: "*cheap internet*") queries, leveraged from ODP categories (DZ), Bing suggestions (BS), and the official TREC Web track sub-topics (WT).

	query #6: "*kcs*" (ambiguous)		
	DZ	BS	WT
1	business	kanawha county schools	kansas city southern railroad
2	computers	klinicki centar srbije	kansas city southern railroad jobs
3	games	union pacific	kanawha county schools west virginia
4	health	kcs railroad	knox county school system tennessee
5	home	kcs energy	kcs energy petrohawk merger
	query #10: "*cheap internet*" (underspecified)		
	DZ	BS	WT
1	business	cheap high-speed internet	low-cost broadband providers
2	computers	cheap dsl internet	dial up internet providers
3	games	cheap internet no phone line	cable television bundle
4	health	cheap internet service	vonage homepage
5	home	cheap broadband	free wireless providers

The first of our considered representations, DZ, models different query aspects as different top-level categories from the Open Directory Project[6] (ODP): adult, arts, business, computers, games, health, home, news, recreation, reference, regional, science, shopping, society, and sports. In turn, the BS representation exploits query reformulations produced by a web search engine in order to model multiple query aspects. In particular, using the Bing Suggestion API,[7] we obtain a set of suggestions for each of the 145 TREC Web track queries. Lastly, as an oracle aspect representation, we consider the official TREC Web track sub-topics (WT). As discussed in Section 5.1.1, these sub-topics were identified by TREC

[6] http://www.dmoz.org
[7] http://msdn.microsoft.com/en-us/library/dd251072.aspx

Table 5.5: Statistics of the explicit aspect representations used in the experiments in this chapter: ODP categories (DZ), Bing suggestions (BS), and the official TREC Web track sub-topics (WT). On the left: average query length and number of aspects per query. On the right: average aspect length and query-aspect overlap.

	WT09	WT10	WT11
	Mean query length		
	2.122	1.979	3.396
	Aspects per query		
DZ	15.000	15.000	15.000
BS	8.653	8.479	6.979
WT	4.837	4.312	3.333

	WT09	WT10	WT11
	Mean aspect length		
BS	3.000	2.862	3.786
WT	3.772	3.769	5.049
	Mean query-aspect overlap		
BS	0.678	0.684	0.628
WT	0.711	0.670	0.817

assessors as representing the actual information needs underlying each query. As such, they enable a direct comparison of explicit diversification approaches regardless of the impact of any particular aspect representation. Since TREC only provides a natural language description for each sub-topic, we obtain a shorter, keyword-like version using Amazon's Mechanical Turk.[8] This step was necessary to make these sub-topics better resemble real web search queries, which facilitates the query classification tasks performed in Chapters 7 and 9. Note that this procedure by no means interfere with our conclusions, as these keyword-like sub-topics are uniformly deployed for all tested diversification approaches.

In order to instantiate PC (Radlinski & Dumais, 2006), as defined in Equation (3.17), we uniformly redistribute the top 100 documents retrieved by the relevance baseline among the k aspects (i.e., query categories or suggestions) identified for the initial query, in which case both the query itself and each aspect are covered by at most $100/(k+1)$ documents in the final ranking. For both IA-Select and xQuAD, the probability that a document satisfies each aspect is directly incorporated as an estimation of coverage, as defined in Equations (3.23) and (4.7), respectively. In particular, following Agrawal et al. (2009), we estimate the probability that a document satisfies a particular category by deploying a Rocchio classifier (Manning et al., 2008), with the centroid that represents the category comprising 3,000 documents randomly selected from the ClueWeb09 B

[8] http://www.mturk.com

corpus and that belong exclusively to this category in ODP. As for both the BS and WT aspect representations, the probability that a document satisfies a given suggestion or sub-topic is computed as the estimated relevance of the document with respect to this suggestion or sub-topic. For consistency, this probability is estimated by the same mechanism used to produce the initial relevance baseline, namely, DPH or LambdaMART, as introduced in Section 5.2.1.1.

5.2.2 Experimental Results

In the following sections, we analyse the results of our investigations concerning the three research questions stated in Section 5.2. In particular, Section 5.2.2.1 addresses research question Q1, by validating the effectiveness of the xQuAD framework in light of the current state-of-the-art in search result diversification. Section 5.2.2.2 addresses Q2, by validating the hybrid diversification strategy implemented by our probabilistic ranking objective. Lastly, Section 5.2.2.3 addresses Q3, by validating our choice for a user-driven aspect representation.

5.2.2.1 Framework Validation

In this section, we address research question Q1, by validating the xQuAD framework in light of the current state-of-the-art in search result diversification. To this end, we contrast the diversification effectiveness of the framework to that attained by both relevance and diversification baselines, as described in Section 5.2.1.1. Regarding the diversification baselines, they are instantiated in this experiment as per their original description. In particular, MMR (Equation (3.8)) is deployed using cosine as a similarity metric (Carbonell & Goldstein, 1998). For PC (Equation (3.8)), we use Bing suggestions (BS in Table 5.5) as alternative query reformulations (Radlinski & Dumais, 2006). In turn, IA-Select is deployed using ODP categories (DZ in Table 5.5) as a taxonomy of query intents (Agrawal et al., 2009). In order to instantiate our xQuAD framework, we also use Bing suggestions as alternative sub-queries, as it naturally adheres to our view of a user-driven diversification, as discussed in Section 4.1

Table 5.6 shows the results of this investigation, in terms of both ERR-IA@20 and α-nDCG@20. For each diversification approach, a first significance symbol

denotes a statistically significant difference (or lack thereof) from the relevance baseline, namely, DPH or LambdaMART. As described in Section 5.1.2, for each evaluation metric, we also report the number of queries negatively affected (−), positively affected (+), and unaffected (=) with respect to these baselines. In addition, a second significance symbol, when present, denotes a significant difference from the best performing approach in each group, which is underlined. The overall best approach in each column is highlighted in bold.

Table 5.6: Diversification performance of the xQuAD framework compared to MMR, PC, and IA-Select, as prominent representatives of novelty-based, coverage-based, and hybrid diversification approaches, respectively.

	$\mathcal{S}_q$	ERR-IA				α-nDCG			
		@20	−	=	+	@20	−	=	+
DPH		0.253				0.364			
+MMR		$0.253^{\circ\circ}$	55	30	60	$0.367^{\circ\triangledown}$	56	28	61
+PC	BS	$0.256^{\blacktriangle\circ}$	25	58	62	$0.375^{\blacktriangle\triangledown}$	29	55	61
+IA-Select	DZ	$0.250^{\circ\circ}$	67	12	66	$0.356^{\circ\triangledown}$	70	12	63
+xQuAD	BS	$\underline{0.281}^{\circ}$	40	24	81	$\underline{0.402}^{\blacktriangle}$	37	24	84
LambdaMART		0.337				0.464			
+MMR		$0.338^{\circ\circ}$	69	20	56	$0.466^{\circ\circ}$	69	20	56
+PC	BS	$0.339^{\blacktriangle\circ}$	27	52	66	$0.472^{\blacktriangle\circ}$	32	45	68
+IA-Select	DZ	$0.217^{\blacktriangledown\blacktriangledown}$	93	13	39	$0.329^{\blacktriangledown\blacktriangledown}$	98	13	34
+xQuAD	BS	$\underline{\mathbf{0.351}}^{\triangle}$	43	24	78	$\underline{\mathbf{0.479}}^{\blacktriangle}$	42	23	80

From Table 5.6, we first observe that xQuAD is the best performing of all considered approaches in terms of both ERR-IA@20 and α-nDCG@20, with gains of up to 11% on top of DPH, and 4% on top of LambdaMART. These results show that, while a high performing relevance baseline improves the overall diversification performance, it also leaves less room for improvement. Nevertheless, significant improvements compared to these relevance baselines are observed in all cases, except for ERR-IA@20 when xQuAD is deployed on top of DPH.

Compared to the diversification baselines, significant improvements are observed in many cases, particularly on top of DPH for α-nDCG, when MMR, PC, and IA-Select are all significantly outperformed. Indeed, not only does xQuAD perform consistently better on average, but it also compares favourably to all

diversification baselines in terms of the number of affected queries. In particular, MMR improves almost as many queries as it hurts on top of DPH, and even hurts more queries than it improves on top of the stronger LambdaMART baseline. In contrast, PC performs more consistently, always improving more queries than it hurts, on top of both DPH and LambdaMART. Nonetheless, it has the lowest impact among all considered diversification approaches, showing the highest number of unaffected queries. IA-Select, on the other hand, shows an unstable behaviour, consistently hurting more queries than it improves. The reasons for such an instability will be further discussed in Sections 5.2.2.2 and 5.2.2.3. Lastly, xQuAD shows the highest number of improved queries and the second lowest number of hurt queries, behind only PC. These results are consistent for both ERR-IA@20 and α-nDCG@20, and on top of both DPH and LambdaMART.

Overall, the magnitude and consistency of the results in Table 5.6 attest the effectiveness of xQuAD and answer research question Q1, regarding the performance of the framework in light of the current state-of-the-art in search result diversification. In particular, these results validate our proposed framework, showing that it compares favourably to effective novelty-based, coverage-based, and hybrid diversification approaches from the literature. In the remainder of this section, we analyse the reasons for such an improved effectiveness in terms of the aspect representation and the diversification strategy deployed by xQuAD.

5.2.2.2 Diversification Strategy

As discussed in Section 3.3, the various diversification approaches in the literature differ essentially according to two dimensions: aspect representation and diversification strategy. While the aspect representation defines the underlying view of the retrieved documents in light of multiple query aspects, the diversification strategy defines how these documents should be ranked given the considered aspect representation. In Section 5.2.2.1, we evaluated the xQuAD framework in contrast to three representative diversification approaches from the literature, namely, MMR, PC, and IA-Select. Although this investigation served the purpose of validating xQuAD in light of the current state-of-the-art, it is unclear where the observed superior performance of the framework comes from, mostly

because the considered approaches, instantiated in their originally proposed form, deployed different aspect representations and diversification strategies.

In order to better understand the role of the complementary dimensions of aspect representation and diversification strategy on the overall effectiveness of xQuAD, in this and the next section, we evaluate each of these dimensions separately. In particular, in this section, we address research question Q2, by assessing the effectiveness of xQuAD's hybrid diversification strategy, based upon a probabilistic mixture of relevance and diversity estimates, as defined in Equation (4.7). To this end, we contrast the strategies deployed by PC, IA-Select, and xQuAD,[9] while holding their underlying aspect representation fixed. In addition to ODP categories and Bing suggestions (the DZ and BS aspect representations in Table 5.5, respectively), in this experiment, we also consider the official TREC Web track sub-topics (WT in Table 5.5) as an oracle aspect representation.

The results of this investigation are shown in Table 5.7. In particular, the strategies deployed by PC, IA-Select, and xQuAD are tested across each of the DZ, BS, and WT aspect representations, on top of both the DPH and LambdaMART relevance baselines. As in the previous section, for each diversification approach, a first significance symbol denotes a statistically significant difference (or lack thereof) with respect to the relevance baseline. A second such symbol, when present, denotes significance with respect to the best performing diversification approach for each aspect representation, which is underlined in the table. The overall best approach in each column is highlighted in bold.

From Table 5.7, we first observe that the hybrid diversification strategy deployed by xQuAD is consistently the most effective across all three considered aspect representations. In particular, regarding the DZ aspect representation, the coverage-based strategy deployed by PC consistently hurts more queries than it improves. As for the hybrid strategy deployed by IA-Select, despite having been originally proposed to leverage a taxonomy of query categories as an aspect representation, it also hurts more queries than it improves. This is partly due to the fact that IA-Select's formulation, as defined in Equation (3.23), does not explicitly incorporate a notion of relevance. As a result, many irrelevant docu-

[9] MMR is left out of this experiment, as it cannot directly leverage an explicit aspect representation. The role of novelty as a diversification strategy is investigated in Chapter 8.

ments can be inadvertently promoted when trying to achieve a high coverage of multiple categories. This effect is particularly exacerbated on top of the stronger LambdaMART baseline, in which case a mishandled diversification may significantly hurt an otherwise effective ranking. This problem is overcome by xQuAD's strategy that trades off relevance and diversity in the ranking, as defined in Equation (4.7). Indeed, xQuAD is the only approach that can significantly improve upon both DPH and LambdaMART using the DZ representation.

Table 5.7: Diversification strategy performance for fixed aspect representations.

	$\mathcal{S}_q$	ERR-IA				α-nDCG			
		@20	−	=	+	@20	−	=	+
DPH		0.253				0.364			
+PC	DZ	$0.253^{\circ\blacktriangledown}$	20	113	12	$0.367^{\circ\blacktriangledown}$	20	113	12
+IA-Select	DZ	$0.250^{\circ\blacktriangledown}$	67	12	66	$0.356^{\circ\blacktriangledown}$	70	12	63
+xQuAD	DZ	$\underline{0.312}^{\blacktriangle}$	49	12	84	$\underline{0.425}^{\blacktriangle}$	52	12	81
+PC	BS	$0.256^{\blacktriangle\circ}$	25	58	62	$0.375^{\blacktriangle\triangledown}$	29	55	61
+IA-Select	BS	$0.267^{\circ\circ}$	55	9	81	$0.382^{\circ\triangledown}$	55	9	81
+xQuAD	BS	$\underline{0.281}^{\circ}$	40	24	81	$\underline{0.402}^{\blacktriangle}$	37	24	84
+PC	WT	$0.253^{\circ\blacktriangledown}$	7	114	24	$0.369^{\blacktriangle\blacktriangledown}$	7	113	25
+IA-Select	WT	$0.330^{\blacktriangle\circ}$	46	9	90	$0.446^{\blacktriangle\circ}$	46	10	89
+xQuAD	WT	$\underline{0.331}^{\blacktriangle}$	42	11	92	$\underline{0.448}^{\blacktriangle}$	39	11	95
LambdaMART		0.337				0.464			
+PC	DZ	$0.338^{\circ\triangledown}$	50	73	22	$0.466^{\circ\circ}$	50	73	22
+IA-Select	DZ	$0.217^{\blacktriangledown\blacktriangledown}$	93	13	39	$0.329^{\blacktriangledown\blacktriangledown}$	98	13	34
+xQuAD	DZ	$\underline{0.359}^{\triangle}$	55	21	69	$\underline{0.476}^{\circ}$	55	20	70
+PC	BS	$0.339^{\blacktriangle\circ}$	27	52	66	$0.472^{\blacktriangle\circ}$	32	45	68
+IA-Select	BS	$0.343^{\circ\circ}$	61	17	67	$0.470^{\circ\circ}$	58	17	70
+xQuAD	BS	$\underline{0.352}^{\triangle}$	43	24	78	$\underline{0.479}^{\blacktriangle}$	42	23	80
+PC	WT	$0.338^{\triangle\blacktriangledown}$	6	119	20	$0.469^{\triangle\blacktriangledown}$	8	116	21
+IA-Select	WT	$0.373^{\blacktriangle\circ}$	48	18	79	$0.503^{\blacktriangle\circ}$	48	17	80
+xQuAD	WT	$\underline{\mathbf{0.376}}^{\blacktriangle}$	42	19	84	$\underline{\mathbf{0.506}}^{\blacktriangle}$	36	17	92

Regarding the BS aspect representation, all approaches improve compared to the relevance baselines, with significant gains observed for both PC and xQuAD. Nevertheless, although effective, this representation may not accurately represent

the exact information needs underlying each query. As a result, aggressive diversification strategies such as the one deployed by IA-Select[10] may be unsafe. Indeed, as observed from Table 5.7, IA-Select hurts a higher number of queries compared to PC and xQuAD, ultimately precluding a more pronounced performance in terms of ERR-IA@20 and α-nDCG@20. Once again, xQuAD overcomes this problem by appropriately balancing relevance and diversity in the ranking.

Lastly, we contrast the diversification strategies deployed by the considered approaches using the official TREC Web track sub-topics as an oracle aspect representation. In particular, the WT representation enables a direct comparison of the diversification strategies deployed by PC, IA-Select, and xQuAD at their full potential. Indeed, in this scenario, both IA-Select and xQuAD excel, attesting to the effectiveness of their deployed strategies. On the other hand, even under these idealised conditions, the mixture model implemented by xQuAD helps reduce the risk of an overly aggressive diversification, further improving compared to IA-Select in terms of ERR-IA@20 and α-nDCG@20, as well as in terms of the total number of queries positively and negatively affected.

Overall, the results in Table 5.7 answer research question Q2, regarding the effectiveness of xQuAD's hybrid diversification strategy. In particular, this strategy was shown to be significantly more effective than those deployed by both PC and IA-Select, as state-of-the-art representatives of coverage-based and hybrid diversification approaches, respectively. Furthermore, besides being effective, the diversification strategy deployed by xQuAD was also shown to be more robust, consistently improving more queries than it hurts for all the considered aspect representations, as a result of appropriately balancing the trade-off between promoting relevance or diversity in different scenarios. As we will show in Chapter 9, such a robustness can be improved even further, by automatically adapting this trade-off according to the level of ambiguity of each individual query.

5.2.2.3 Aspect Representation

The results in the previous section attested the effectiveness of xQuAD's diversification strategy compared to those deployed by PC and IA-Select across multiple

[10] As discussed in Section 4.4, IA-Select can be reduced to a special case of xQuAD, with $\lambda = 1$, indicating the maximum emphasis on promoting diversity rather than relevance.

aspect representations. In order to address research question Q3, in this section, we perform the complementary investigation, by assessing the effectiveness of xQuAD's user-driven aspect representation. To this end, we contrast the DZ, BS, and WT aspect representations, described in Table 5.5, while holding the diversification strategy fixed. In particular, as discussed in Section 4.1, we hypothesise that a representation that explicitly aims to reflect the possible information needs underlying a query, such as BS and WT, is more effective.

The results of this investigation are shown in Table 5.8, which provides a complementary view of the results in Table 5.7. In particular, each of the DZ, BS, and WT aspect representations are tested across the diversification strategies implemented by PC, IA-Select, and xQuAD, on top of both the DPH and LambdaMART relevance baselines. As in the previous sections, for each diversification approach and a given aspect representation, a first significance symbol denotes a statistically significant difference (or lack thereof) with respect to the relevance baseline. A second such symbol, when present, denotes significance with respect to the best performing aspect representation for each approach, which is underlined. The overall best approach in each column is highlighted in bold.

From Table 5.8, we observe a few trends, depending on the deployed diversification strategy. In particular, for the coverage-based strategy deployed by PC, BS is the most effective of the considered aspect representations. Indeed, on top of DPH, this representation significantly outperforms both the DZ and even the oracle WT representation. On top of the stronger LambdaMART relevance baseline, BS is also the best performing representation for PC, although not significantly. Interestingly, the WT representation results in the lowest impact of PC, with most of the queries unaffected when leveraging this representation.

A different situation is observed for the hybrid strategies deployed by IA-Select and xQuAD. As discussed in the previous section, since IA-Select does not directly incorporate a notion of relevance, it performs better for aspect representations that are somewhat correlated with relevance. In particular, as keyword-based representations of the possible information needs underlying a query, BS and WT often comprise aspects (i.e., query suggestions or sub-topics) that share common terms with the query, as previously shown in Table 5.5 in terms of overlap. As a result, these aspect representations are able to at least partially convey an es-

Table 5.8: Aspect representation performance for fixed diversification strategies.

	$\mathcal{S}_q$	ERR-IA				α-nDCG			
		@20	−	=	+	@20	−	=	+
DPH		0.253				0.364			
+PC	DZ	$0.253^{\circ\blacktriangledown}$	20	113	12	$0.367^{\circ\blacktriangledown}$	20	113	12
+PC	BS	$\underline{0.256}^{\blacktriangle}$	25	58	62	$\underline{0.375}^{\blacktriangle}$	29	55	61
+PC	WT	$0.253^{\circ\blacktriangledown}$	7	114	24	$0.369^{\blacktriangle\triangledown}$	7	113	25
+IA-Select	DZ	$0.250^{\circ\blacktriangledown}$	67	12	66	$0.356^{\circ\blacktriangledown}$	70	12	63
+IA-Select	BS	$0.267^{\circ\blacktriangledown}$	55	9	81	$0.382^{\circ\blacktriangledown}$	55	9	81
+IA-Select	WT	$\underline{0.330}^{\blacktriangle}$	46	9	90	$\underline{0.446}^{\blacktriangle}$	46	10	89
+xQuAD	DZ	$0.312^{\blacktriangle\circ}$	49	12	84	$0.425^{\blacktriangle\circ}$	52	12	81
+xQuAD	BS	$0.281^{\circ\blacktriangledown}$	40	24	81	$0.402^{\blacktriangle\blacktriangledown}$	37	24	84
+xQuAD	WT	$\underline{0.331}^{\blacktriangle}$	42	11	92	$\underline{0.448}^{\blacktriangle}$	39	11	95
LambdaMART		0.337				0.464			
+PC	DZ	$0.338^{\circ\circ}$	50	73	22	$0.466^{\circ\circ}$	50	73	22
+PC	BS	$\underline{0.339}^{\blacktriangle}$	27	52	66	$\underline{0.472}^{\blacktriangle}$	32	45	68
+PC	WT	$0.338^{\triangle\circ}$	6	119	20	$0.469^{\triangle\circ}$	8	116	21
+IA-Select	DZ	$0.217^{\blacktriangledown\blacktriangledown}$	93	13	39	$0.329^{\blacktriangledown\blacktriangledown}$	98	13	34
+IA-Select	BS	$0.343^{\circ\triangledown}$	61	17	67	$0.470^{\circ\blacktriangledown}$	58	17	70
+IA-Select	WT	$\underline{0.373}^{\blacktriangle}$	48	18	79	$\underline{0.503}^{\blacktriangle}$	48	17	80
+xQuAD	DZ	$0.359^{\triangle\circ}$	55	21	69	$0.476^{\circ\triangledown}$	55	20	70
+xQuAD	BS	$0.352^{\triangle\triangledown}$	43	24	78	$0.479^{\blacktriangle\blacktriangledown}$	42	23	80
+xQuAD	WT	$\underline{\mathbf{0.376}}^{\blacktriangle}$	42	19	84	$\underline{\mathbf{0.506}}^{\blacktriangle}$	36	17	92

timation of the relevance of a given document with respect to the initial query, significantly outperforming the DZ representation, which notably lacks this property. As also observed in the previous section, while xQuAD is able to perform effectively even for aspect representations seemingly uncorrelated with relevance, like DZ, representations such as BS and WT improve the overall robustness of the framework. Indeed, on top of both relevance baselines, BS consistently results in more improved and fewer hurt queries compared to DZ, with the oracle WT representation significantly outperforming both representations in most cases.

Recalling research question Q3, on the effectiveness of a user-driven aspect representation, while different approaches seem to benefit more from different

representations, leveraging query reformulations as a representation of the multiple possible information needs underlying a query is a consistently effective alternative, at least for hybrid diversification approaches. This observation is corroborated by the significantly higher performance attained by the oracle WT representation for both IA-Select and xQuAD, which conveys a representation of the actual needs underlying each query. Further approaches aimed to achieve an effective user-driven aspect representation will be the focus of Chapter 6.

5.3 Summary

The previous chapter introduced xQuAD as general framework for search result diversification, aimed at maximising the satisfaction of the multiple possible information needs underlying a query. In this chapter, we have thoroughly validated the *effectiveness* of the framework, by contrasting it to the current state-of-the-art in search result diversification, as introduced in Chapter 3.

In Section 5.1, we have detailed the basic methodology that underlies all the experiments conducted in this thesis. In particular, in Section 5.1.1, we have described the test collections used in our experiments, based upon the evaluation paradigm provided by the TREC 2009, 2010, and 2011 Web tracks (Clarke et al., 2009a, 2010, 2011b). In Section 5.1.2, we have further detailed the procedures for training and testing the several approaches investigated in this thesis, with a view towards ensuring a fair and thorough evaluation.

In Section 5.2, we have instantiated the aforementioned experimental methodology in order to validate the xQuAD framework. In particular, in Section 5.2.2.1, the diversification performance of xQuAD was contrasted to that of effective representatives of the three families of diversification approaches introduced in Section 3.3, namely, novelty-based, coverage-based, and hybrid approaches. Our proposed framework was shown to consistently outperform these diversification approaches under multiple experimental conditions and according to multiple evaluation metrics. Indeed, we have shown that not only does xQuAD bring larger improvements on top of two effective relevance-oriented baselines, but it also performs more robustly than the considered diversification approaches, in terms of the number of queries positively and negatively affected.

The reasons for such a superior performance were further investigated in Sections 5.2.2.2 and 5.2.2.3, by breaking down the evaluation in Section 5.2.2.1 across the complementary dimensions of diversification strategy and aspect representation, as introduced in Section 3.3. By methodically combining multiple instantiations of each dimension, we validated both the diversification strategy deployed by xQuAD, which appropriately mixes relevance and diversity estimates in a probabilistic ranking objective, as well as its user-driven aspect representation, based upon an explicit account of multiple information needs as sub-queries. In particular, in Section 5.2.2.2, we showed that xQuAD's hybrid diversification strategy consistently outperforms the strategies deployed by the considered diversification approaches across multiple (fixed) aspect representations. In addition, in Section 5.2.2.3, we showed that, while different diversification approaches benefit more or less from different aspect representations, the user-driven representation adopted by xQuAD based on query suggestions is consistently effective for all the considered approaches. Moreover, in contrast to other diversification approaches, by incorporating a probability of relevance as part of its ranking objective, xQuAD is able to successfully leverage aspect representations that have no apparent bearing on topical relevance, such as query categories.

After validating the framework in light of the current state-of-the-art, in the subsequent chapters, we will experiment with each of its components in turn. As a starting point, Chapter 6 will introduce an effective and efficient approach for identifying sub-queries from the query logs of a web search engine. In turn, Chapter 7 will describe a supervised approach for predicting the possible intents underlying each sub-query in order to effectively estimate their coverage among the retrieved documents. Chapter 8 will deeply analyse the role of novelty as a diversification strategy, both in isolation as well as when combined with coverage. Lastly, Chapter 9 will introduce a selective approach for automatically determining how much to diversify the retrieved documents on a per-query basis.

Chapter 6

Sub-Query Generation

One of the pillars of the xQuAD framework, as discussed in Section 4.1, is an aspect representation that reflects real users' information needs. In particular, xQuAD represents the multiple possible information needs underlying a query as a set of sub-queries. As shown in the previous chapter, our framework can successfully leverage the sub-queries produced by different mechanisms.

Of the investigated mechanisms, the query suggestions produced by a commercial web search engine were shown to be particularly effective. On the other hand, such a mechanism operates as a black box, which precludes a deeper understanding of how such an effective sub-query set could be generated in practice. To further our understanding of the characteristics of effective sub-queries, in this chapter, we investigate alternative mechanisms to generate and score the relative importance of sub-queries. To this end, we propose a learning to rank approach that identifies query suggestions from a query log as sub-queries. Moreover, we introduce a framework for the quantitative evaluation of query suggestions, both on their own, as well as when used as a resource for diversification.

In the remainder of this chapter, Section 6.1 provides background on query suggestions for web search. Sections 6.2 and 6.3 introduce our approaches for ranking and evaluating query suggestions, respectively. The results of our thorough experiments are discussed in Section 6.4, and attest the effectiveness of our proposed learning to rank approach in comparison to suggestions produced by a state-of-the-art approach from the literature, as well as by a commercial web search engine, even for queries with little or no past usage in a query log.

6.1 Query Suggestions in Web Search

Web search queries are typically short, ill-defined representations of more complex information needs (Jansen et al., 1998). As a result, they can lead to unsatisfactory retrieval performance. Query suggestions have been introduced as a mechanism to alleviate this problem. Such a mechanism builds upon the vast amount of querying behaviour recorded by search engines in the form of query logs, in order to suggest related queries previously issued by other users with a similar information need (Silvestri, 2010). The mined suggestions can be exploited in a variety of ways. For instance, a suggestion identified with high confidence can be considered for automatically rewriting the user's initial query (Jones et al., 2006). Alternatively, a few high quality suggestions can be offered to the user as alternatives to the initial query (Baeza-Yates et al., 2004), or to help diversify the documents retrieved for this query, as we will show in Section 6.4.

6.1.1 Query Suggestion Approaches

Several approaches have been proposed in recent years to infer the importance of a candidate suggestion for a given query based on these queries' textual similarity, their co-occurrence in common sessions, or their common clicked URLs (Silvestri, 2010). For instance, Jones et al. (2006) proposed to generate candidate suggestions from co-session queries with a common substring. The strength of the relationship between the query and each candidate suggestion was further estimated by leveraging various similarity features, such as the edit distance and the mutual information between these queries. Analogously, Wang & Zhai (2008) proposed to mine term association patterns from a query log. Their approach analysed the co-occurrence of terms in multi-word co-session queries and built a translation model in order to mine query suggestions.

A session-based approach was proposed by Fonseca et al. (2003). In particular, they deployed an association rule mining algorithm in order to identify query pairs with sufficient co-occurrence across multiple sessions. Such association rules were then used as the basis for identifying query suggestions from a query log. Relatedly, Zhang & Nasraoui (2006) exploited the sequence of queries in a query log session. In particular, their approach created a graph with queries as nodes,

and with edges connecting consecutive queries in each session, weighted by these queries' textual similarity. A candidate suggestion for a given query was then scored based on the length of the path between the two queries, accumulated across all sessions where the query and the suggestion co-occurred.

A click-based approach was proposed by Baeza-Yates et al. (2004). In particular, they proposed to cluster queries represented using the terms present in the URLs clicked for these queries. Given an input query, candidate suggestions from the same cluster as the query were then weighted based on their similarity to the query and their success rate, as measured by their fraction of clicked documents in a query log. Relatedly, Mei et al. (2008) exploited random walks on a bipartite query-click graph. To this end, they weighted a candidate suggestion for a query based on its "hitting" time (i.e., the time it took for the node representing this query suggestion to be visited for the first time) for a random walk starting from the input query. Similarly, Boldi et al. (2009a) proposed to weight candidate suggestions by performing a short random walk on different slices of a query-flow graph, a query transition graph with edges classified as generalisations, specialisations, error corrections, or parallel moves (Boldi et al., 2008).

6.1.2 Query Suggestion under Sparsity

Random walk approaches are generally regarded as the state-of-the-art in the literature dedicated to the query suggestion problem (Silvestri, 2010). Despite their relative success, most of these approaches share a common shortcoming. In particular, they underperform and can even fail to produce any relevant suggestion for queries with sparse or no past usage in a query log, which amount to a substantial fraction of the web search traffic (Downey et al., 2007). In order to overcome this issue, Szpektor et al. (2011) proposed the notion of query template, a generalisation of a query in which entities are replaced with their type. By enriching the query-flow graph (Boldi et al., 2008) with query templates, their approach was able to effectively generate suggestions for long-tail queries. A different approach aimed at tackling query sparsity was proposed by Broccolo et al. (2012). In particular, they proposed to index each query in a query log as a *virtual document* comprising the terms in the query itself and those of other queries

from common sessions. As a result, they cast the query suggestion problem as an efficient search over the inverted index of virtual documents.

In Section 6.2, we will build upon the query representation strategy proposed by Broccolo et al. (2012), which was shown to perform at least as effectively as the state-of-the-art query-flow graph approach of Boldi et al. (2009a) for head queries, while consistently outperforming it for queries with little or no past evidence (Broccolo et al., 2012, Section 4.4). Inspired by this approach, we devise a *structured* virtual document representation, by treating terms from different sources as distinct *fields*. In particular, besides the candidate suggestion itself and its co-session queries, we also leverage evidence from queries that share at least one click with the suggestion. This enriched representation provides multiple criteria for ranking suggestions with respect to a query, which we encode as query-dependent features in a unified ranking model automatically learned from training data. To further improve this model, we propose several query-independent features as quality indicators for a candidate suggestion.

In this vein, Dang et al. (2010) proposed a machine learning approach to identify effective terms from a query log to be appended to an input query. More recently, Song et al. (2011b) proposed a learning approach to produce diverse suggestions in response to a query. While also employing learning to rank, our approach differs from the aforementioned approaches in two fundamental ways. In particular, while Dang et al. (2010) identified effective *expansion terms*, we are interested in the more general problem of query suggestion. As for the approach of Song et al. (2011b), instead of relying on the human assessment of suggestion effectiveness, which can be misleading (Hauff et al., 2010), we explicitly incorporate the *observed* retrieval effectiveness (in terms of adhoc and diversity search) of a set of candidate suggestions in order to guide the learning process.

6.2 Learning to Rank Query Suggestions

With the abundant usage data available to commercial web search engines, query suggestion has traditionally been approached as a data-driven problem, as exemplified by the various approaches described in Section 6.1. While different approaches have exploited such rich data with more or less success, we argue that

the importance of a suggestion with respect to a query cannot be fully explained by a single criterion. Instead, we propose to estimate this importance by leveraging multiple ranking criteria within a supervised learning to rank setting. As a result, not only do we move beyond the traditional approaches to query suggestion, but we make it possible to leverage these otherwise successful approaches as additional features in a robust query suggestion model.

In the following, Section 6.2.1 discusses alternatives for producing a sample of candidate suggestions for learning to rank. Section 6.2.2 describes our learning approach deployed to re-rank the produced samples. Lastly, Section 6.2.3 describes our proposed features to represent a candidate suggestion for ranking.

6.2.1 Sampling Query Suggestions

As discussed in Section 2.2.3, learning to rank approaches typically operate on top of a *sample* of documents retrieved by a standard ranking model (Liu, 2009), such as BM25 (Equation (2.13)) or DPH (Equation (2.31)). The documents in this sample are then used by the learning approach to produce a feature-rich ranking model, which will be later used to rank the documents retrieved for unseen queries. An effective sample should have high recall, in order to increase the number of relevant examples from which to learn (Liu, 2009). In the case of query suggestions, such a high-recall sample can be obtained by exploiting the rich information about a user query contained in a query log. In particular, we can describe a query log $\mathcal{L}$ as a set of records $\langle u_j, q_i, b_i, \mathcal{R}_{q_i}, \mathcal{K}_{q_i} \rangle$, where q_i is a query issued by user u_j at timestamp b_i. For this query, the user was shown a set of documents $\mathcal{R}_{q_i}$ and clicked on the subset $\mathcal{K}_{q_i} \subseteq \mathcal{R}_{q_i}$. Typically, queries issued by the same user within a short timeframe (say, 30 min) are further grouped into a logical session, ideally reflecting a cohesive search mission (Silvestri, 2010).

Most query suggestion approaches in the literature exploit the co-occurrence of queries in a session or their clicks in a common document in order to produce effective suggestions. However, as discussed in Section 6.1.2, these approaches generally underperform for rare or unseen queries (Silvestri, 2010). In the former case, there is little evidence of the query's co-occurrence with potential suggestions in the query log. In the latter case, the initial query itself cannot even be

located in the log. To tackle this data sparsity problem, Broccolo et al. (2012) proposed to represent queries in a query log as *virtual documents*. This bag-of-words representation comprises not only the words in the query itself, but also those present in other queries with a common session in the log. Such a representation combats data sparsity, since even previously unseen queries (i.e., queries without an exact match in the query log) will likely have at least one of their constituent words present in the log, which in turn may occur frequently in the virtual document representation of a relevant suggestion. Additionally, this representation enables the suggestion problem to be efficiently tackled as a standard search over an inverted index, with the potential to scale to extremely large query logs (Dean, 2009). On the other hand, this representation lacks a more fine-grained treatment of the multiple evidence available for ranking. In particular, it does not distinguish between words from different sources.

In order to address this issue and to produce an effective sample of candidate suggestions for learning to rank, we improve upon the bag-of-words representation proposed by Broccolo et al. (2012) by considering each available source of evidence as a separate *field* in a *structured virtual document*.[1] As a result, words that appear in a query suggestion can be weighted differently from those that appear in related queries with a common session. Moreover, we integrate an additional source of evidence as a third field in our structured virtual document representation. In particular, for each candidate suggestion, we also store words from queries with at least one common click in the query log. As an illustrative example, Figure 6.1 shows an excerpt of the structured virtual document representing "metallica" as a candidate suggestion, highlighting this query itself (Q), co-session queries (S), and queries with a common click (C) as separate fields. Also note the "count" attribute for each entry (E) in Figure 6.1, which denotes the frequency with which this entry co-occurs with "metallica" in the entire query log (e.g., the queries "metallica" and "james hetfield" have 60 common clicks). During indexing, the term frequency $tf_{t,s}$ of each term t in a suggestion s is computed as the sum of the "count" values across all entries of s where t occurs.

[1] An analogy to the document ranking problem can be made in which field-based models, such as BM25F (Zaragoza et al., 2004), leverage evidence from fields such as the title, body, URL, or the anchor text of incoming hyperlinks in order to score a document.

```
<DOC>
  <DOCNO> metallica </DOCNO>
  <Q> metallica </Q>
  <S> <E count="1"> metalica </E>
      <E count="1"> queensryche </E>
      <E count="1"> ac dc </E>
      <E count="1"> pantera </E>
        ... </S>
  <C> <E count="4"> history of mettalica </E>
      <E count="1"> metallica concerts </E>
      <E count="18"> metclub </E>
      <E count="60"> james hetfield </E>
        ... </C>
</DOC>
```

Figure 6.1: Virtual document representation for the suggestion "metallica".

When retrieving a sample of suggestions for a given query, there are multiple choices regarding which of the available fields to use: different choices lead to different samples for the same query (e.g., a sample of suggestions built by searching the Q field will probably be different from a sample based upon the S or C fields). A more fundamental question is which sessions should contribute candidate suggestions. In particular, satisfactory sessions are those with at least one click in the last query in the session (Broccolo et al., 2012). Figure 6.2 provides an illustration of unsatisfactory and satisfactory 3-query sessions.

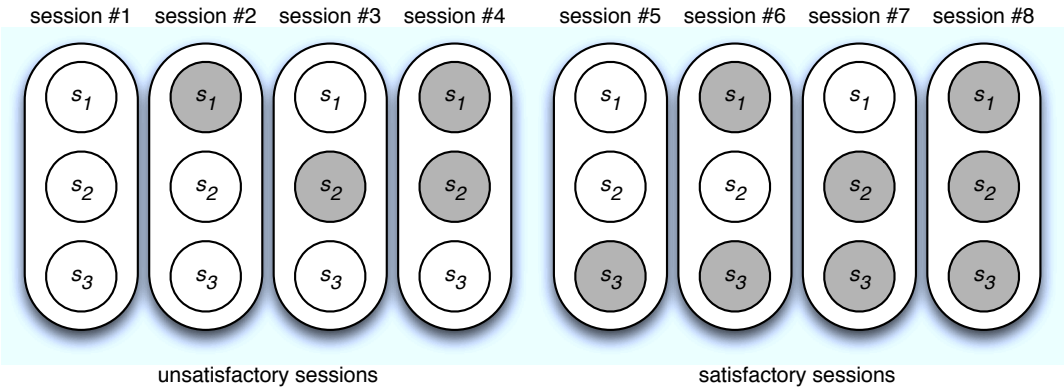


Figure 6.2: Unsatisfactory (#1 to #4) and satisfactory (#5 to #8) sessions with suggestions s_1, s_2, and s_3. Queries with clicks in each session are shaded.

In their approach, Broccolo et al. (2012) used only queries that ended a satisfactory session (i.e., s_3 in sessions #5 to #8 in Figure 6.2). Arguably, non-satisfactory sessions (i.e., sessions with no clicks, such as session #1, or without clicks on the last query in the session, such as sessions #2 to #4) can also contribute relevant suggestions. Moreover, non-final queries (queries s_1 and s_2 in Figure 6.2) in both satisfactory and non-satisfactory sessions may also be useful. In Section 6.4.2, we will investigate multiple structured virtual document representations based on different combinations of the available fields (i.e., Q, S, and C), as well as on different sampling criteria (i.e., whether to index queries from all sessions or from only satisfactory sessions, and whether to index all or only the last query in each of these sessions). A breakdown of these alternative representations in terms of the storage overhead incurred by each of them is provided in Table 6.1. Percentage figures denote the incurred overhead compared to storing only the query string (Q) of each suggestion. The total number of queries indexed for different representations is shown in the bottom row of the table.

Table 6.1: Space requirements for storing each of the seven considered structured virtual document representations: Q, S, C, QS, QC, SC, QSC.

	Sessions	All				Satisfactory			
	Queries	All		Last		All		Last	
Uncompressed	Q	141.7		78.3		86.4		44.2	
	S	513.4	(+262%)	92.7	(+18%)	322.4	(+273%)	62.1	(+41%)
	C	278.8	(+97%)	210.5	(+169%)	256.2	(+196%)	201.2	(+356%)
	QS	655.1	(+362%)	171.0	(+118%)	408.8	(+373%)	106.3	(+141%)
	QC	420.5	(+197%)	288.8	(+269%)	342.6	(+296%)	245.3	(+456%)
	SC	792.2	(+459%)	303.2	(+287%)	578.6	(+570%)	263.2	(+496%)
	QSC	933.9	(+559%)	381.5	(+387%)	665.0	(+670%)	307.4	(+596%)
Compressed	Q	56.0		32.0		33.4		16.8	
	S	139.3	(+149%)	34.1	(+7%)	95.3	(+185%)	22.8	(+35%)
	C	56.6	(+1%)	44.5	(+39%)	52.7	(+58%)	42.7	(+154%)
	QS	195.3	(+249%)	66.1	(+107%)	128.7	(+285%)	39.7	(+135%)
	QC	112.6	(+101%)	76.5	(+139%)	86.1	(+158%)	59.6	(+254%)
	SC	195.9	(+250%)	78.6	(+146%)	145.0	(+343%)	65.5	(+289%)
	QSC	251.9	(+350%)	110.6	(+246%)	181.4	(+443%)	82.4	(+389%)
# suggestions		6,382,973		3,484,172		4,075,725		2,118,571	

Firstly, restricting the index to comprise only satisfactory sessions or only the last query in each session naturally reduces the required storage space, since fewer queries are considered as candidate suggestions. More interestingly, compared to a suggestion representation based upon the query string (Q) only, a representation enriched with co-session (S) and co-clicked (C) queries does not affect the asymptotic space complexity of our approach. Indeed, all increases in space requirements stay within an order of magnitude of the space required to store the query alone. In particular, the most space-consuming representation (QSC) requires only 6.7 times more storage space (4.4 times after compression with gzip[2]) compared to the least space-consuming one (Q).

6.2.2 Learning a Query Suggestion Model

Given a query q and an indexed query log $\mathcal{L}$, we can now define query suggestion as the problem of retrieving a list of queries $\mathcal{S}_q \subseteq \mathcal{L}$, in decreasing order of importance $p(s|q), \forall s \in \mathcal{S}_q$. The retrieved suggestions could help the user better specify the information need originally expressed by q, or to diversify the documents retrieved for this query, in the hope of providing at least one relevant document for each of the possible information needs underlying the query.

In order to estimate the importance $p(s|q)$ of a suggestion s given a query q, we must learn an optimal ranking function $h : \mathcal{X} \rightarrow \mathcal{Y}$, mapping the input space $\mathcal{X}$ to the output space $\mathcal{Y}$. In particular, we define the input space $\mathcal{X}$ of the query q as comprising a sample $\mathbf{x} = \{\mathbf{x}_j\}_{j=1}^{n_q}$ of n_q suggestions mined for q, as discussed in the previous section. Each element $\mathbf{x}_j = \Phi(q, s_j)$ in the sample is a vector representation of a candidate suggestion s_j, according to the feature extractor Φ. In Section 6.2.3, we will describe the various query suggestion features used in our investigation, including query-dependent and query-independent ones.

The output space $\mathcal{Y}$ for our learning problem contains a set of ground-truth labels $\mathbf{y} = \{y_j\}_{j=1}^{n_q}$. In order to target the learning process towards identifying effective query suggestions, each label y_j is automatically defined based on the *observed* retrieval effectiveness e_j of the document ranking produced for the query suggestion s_j. Precisely, we defined the label y_j as:

[2]http://www.gnu.org/software/gzip

$$y_j = \begin{cases} 3 : \text{if } e_j > e, \\ 2 : \text{if } e_j = e, \\ 1 : \text{if } e_j > 0, \\ 0 : \text{otherwise}, \end{cases} \tag{6.1}$$

where $e = \Delta_r(\mathcal{R}_q|q, \kappa_r)$ and $e_j = \Delta_r(\mathcal{R}_{s_j}|q, \kappa_r)$ denote the retrieval performance at rank κ_r (given by any standard evaluation metric Δ_r, such as nDCG@10 or any of the metrics in Section 2.3.3) attained by the ranking $\mathcal{R}_\bullet$ produced by a reference retrieval system for a given input (i.e., the query q or its suggestion s_j).

Lastly, we must define a loss function to guide our learning process. In particular, we define $\Delta_s(\mathcal{S}_q|q, \kappa_s)$ as the loss at rank κ_s of retrieving the suggestions $\mathcal{S}_q$ in response to the query q. Note that, different from the document ranking evaluation metric Δ_r used to define our ground-truth labels in Equation (6.1), this metric is used to evaluate rankings of *query suggestions*. Our experimental setup choices for the sample size n_q, labelling function Δ_r and cutoff κ_r, loss function Δ_s and cutoff κ_s, and learning algorithms are fully described in Section 6.4.1.3.

6.2.3 Query Suggestion Features

Having discussed alternative approaches for sampling candidate suggestions from a query log and how to learn an effective ranking function for a given sample, we now describe the features used to represent each suggestion in the learning process. As summarised in Table 6.2, we broadly organise all query suggestion features used by our approach as either query-dependent or query-independent, according to whether they are computed on-the-fly at querying time or offline at indexing time, respectively. While the considered query-dependent features are standard features commonly used in the literature for learning to rank for web search (Liu, 2009), the query-independent ones are specifically proposed here to estimate the quality of different candidate suggestions.

Given a query q, the query-dependent features are directly computed by scoring the occurrences of the terms of q in each field of each candidate suggestion.

Table 6.2: Features used in this chapter for each candidate suggestion s_j.

	Feature	Description	Equation	Total
Query-dependent	CLM	Full and per-field CLM score	(2.5)	4
	BM25	Full and per-field BM25 score	(2.13)	4
	LM	Full and per-field LM score	(2.25)	4
	MRF	Full MRF score	(2.20)	1
	PL2	Full and per-field PL2 score	(2.29)	4
	DPH	Full and per-field DPH score	(2.31)	4
	pBiL	Full pBiL score	(2.32)	1
Query-independent	Tokens	Full and per-field token count		4
	Terms	Fraction of unique terms in s_j		1
	Chars	Number of characters in s_j		1
	RepChars	Presence, number, fraction of repeated characters in s_j		3
	Digits	Number and fraction of digits in s_j		2
	Punctuation	Number and fraction of punctuation characters in s_j		2
	Badwords	Mean, s.d., and median number of swearing words in s_j		3
	UrlFragments	Whether s_j contains a URL		2
	Clicks	Number of clicked documents for s_j		1
	Sessions	Number of sessions with s_j		1
	SessionClicks	Mean, s.d., and max number of clicks on s_j per session		3
	SessionLength	Mean, s.d., and max number of queries in sessions with s_j		3
	SessionPosition	Mean, s.d., and max position of s_j per session		3
	SessionSuccess	Fraction of successful sessions with s_j		1
	Grand total			52

To this end, we leverage multiple query-dependent ranking approaches, including standard weighting models, such as BM25 (Equation (2.13)), language modelling with Dirichlet smoothing (LM; Equation (2.25)), the DFR DPH (Equation (2.31)) and PL2 (Equation (2.29)) models, and a simple coordination level matching (CLM; Equation (2.5)). Additionally, we use term dependence models based on Markov Random Fields (MRF; Equation (2.20)) and the DFR framework (pBiL; Equation (2.32)), which highly score suggestions where the query terms co-occur in close proximity. All query-dependent features are efficiently computed at querying time with a single pass over the posting lists for the query q in the index of structured virtual documents (Macdonald, Santos & Ounis, 2013).

As for the query-independent features, they are all computed at indexing time. In particular, we consider features that can be directly estimated from the

query log itself, so as to draw insights regarding which query log evidence can be helpful for ranking query suggestions. The considered features include quality signals, such as the length of the query suggestion in tokens and characters (too long suggestions may denote robot-submitted queries) and the presence of digits, punctuation, and swearing (which usually indicate low-quality or adult-oriented queries). Additionally, we also derive features that quantify the popularity of a query suggestion in terms of number of sessions and clicks, as popular suggestions arguably indicate higher quality a priori. Finally, we consider features that summarise the profile of a suggestion across the sessions where it occurs. These include the number of clicks received, the total number of queries and the ratio of clicked queries, and the suggestion's relative position in each session.

6.3 Evaluating Query Suggestions

The effectiveness of a query suggestion mechanism is typically assessed qualitatively, based on user studies (Silvestri, 2010). On the other hand, Hauff et al. (2010) have shown that users are not good at predicting the retrieval performance of query suggestions. At the same time, it seems natural to assess the performance of a suggestion in terms of how much it helps the users satisfy their information need. More precisely, we argue that the effectiveness of a query suggestion mechanism should be assessed as to whether its suggested queries help the users satisfy the information need expressed by their query. With this in mind, we formalise a framework for the quantitative evaluation of query suggestions that directly builds upon existing retrieval evaluation efforts. In particular, we envisage two scenarios, depending on whether or not the user's initial query is ambiguous.

The first scenario assumes that the user's query is unambiguously defined. In this scenario, given a query q and a ranking of suggestions $\mathcal{S}_q$ produced for this query, our goal is to evaluate these suggestions in terms of their retrieval performance when used as a replacement for q. In particular, we introduce $s\text{-}eval_{\Psi}(\bullet)$ for query suggestion evaluation as the counterpart of a standard retrieval evaluation metric $eval(\bullet)$ (e.g., nDCG in Equation (2.51)), according to:

$$s\text{-}eval_{\Psi}(\mathcal{S}_q|q,k,\kappa) = \Psi_{j=1}^{k}\big[eval(\mathcal{R}_{s_j}|q,\kappa)\big], \tag{6.2}$$

where k is the number of top suggestions to be evaluated (the *suggestion* evaluation cutoff), κ is the number of top documents to be evaluated for each suggestion (the *retrieval* evaluation cutoff), $\mathcal{R}_{s_j}$ is the ranking produced for the query suggestion s_j by a reference retrieval system, and Ψ is a summary statistic. In Section 6.4.2, we report both the maximum (Ψ = "max") and the average (Ψ = "avg") retrieval performance attained by the top k suggestions. For instance, with nDCG (Equation (2.51)) used as a document ranking evaluation metric $eval(\bullet)$, Ψ = "max", $k = 1$, and $\kappa = 10$, we can instantiate Equation (6.2) in order to have s-nDCG$_{\text{max}}$@1,10 as a query suggestion evaluation metric. This metric quantifies the effectiveness (in terms of the nDCG@10 performance of the resulting document ranking) of a query suggestion mechanism at providing a single suggestion. Such a suggestion could be used, e.g., for automatically reformulating the initial query. With Ψ = "avg", $k = 8$, and $\kappa = 10$, we can have s-nDCG$_{\text{avg}}$@8,10, which models a typical application of query suggestion, as seen on the search box of modern web search engines. Note that both the Ψ = "max" and Ψ = "avg" summary statistics consider the top k suggestions as an unordered set, regardless of how these suggestions were ranked with respect to each other. Although rank-based summary statistics are certainly possible, this would imply assuming that users prefer the top ranked suggestion over the others. Since, to the best of our knowledge, there is no empirical study supporting this assumption, we opted for a set-based evaluation in our investigations.

The query suggestion evaluation metrics generated by Equation (6.2) assume that the query q unambiguously expresses the user's information need. Indeed, both q and the suggestion s_j are evaluated with respect to the information need represented by q. In practice, however, the queries submitted to a web search engine are often ambiguous (Song et al., 2009), with the same query being used by different search users to represent different information needs (Spärck-Jones et al., 2007). In this situation, providing a diverse list of suggestions could not only help the users better specify their need, but would also enable an effec-

tive diversification of the retrieved documents, as mentioned in Section 6.1. To cater for query ambiguity, we formulate a second scenario within our evaluation framework to quantitatively assess the diversity of the suggestions produced by a given mechanism. Analogously to the definition in Equation (6.2), we introduce *s-deval*(•) for query suggestion evaluation as the counterpart of a diversity evaluation metric *deval*(•) (e.g., α-nDCG in Equation (3.29)), according to:

$$s\text{-}deval(\mathcal{S}_q|q,k,\kappa) = deval(\mathcal{D}_{q,\mathcal{S}_q,k}|q,\kappa), \tag{6.3}$$

where $\mathcal{D}_{q,\mathcal{S}_q,k}$ is the document ranking produced by a reference diversification system for the query q using the top k produced suggestions from $\mathcal{S}_q$, and κ is the depth at which this ranking should be evaluated. For instance, s-α-nDCG@8,10 measures the diversification performance (in terms of α-nDCG@κ, with $\kappa = 10$) of the top $k = 8$ suggestions produced by a given mechanism, when used as input to a diversification approach, such as xQuAD, as introduced in Chapter 4.

With the proposed evaluation framework, using a fixed reference retrieval system, we can quantitatively compare the suggestions produced by different mechanisms with respect to one another, as well as with respect to the retrieval effectiveness attained by the initial query alone (i.e., $eval(\mathcal{R}_q|q,\kappa)$). Likewise, we can also contrast the diversification performance of different suggestion mechanisms in contrast to one another, as well as in comparison to the diversification performance of the initial query (i.e., $deval(\mathcal{R}_q|q,\kappa)$). In the next section, we will leverage this framework to assess the effectiveness of our proposed learning to rank approach as well as of state-of-the-art query suggestion baselines at providing query suggestions for both adhoc and diversity search.

6.4 Experimental Evaluation

In this section, we address the second claim from our thesis statement:

> *"By inferring the relative importance of each sub-query in the context of the initial query, the retrieved results can better cater for the information needs of the user population."*

In order to address this claim, we experiment with our proposed learning to rank approach to produce query suggestions that are effective both on their own as well as when used as sub-queries within our xQuAD diversification framework. In particular, we aim to answer the following research questions:

Q1. How effective is our query suggestion approach for adhoc search?

Q2. How effective is our query suggestion approach for diversity search?

Q3. How robust to data sparsity is our query suggestion approach?

Q4. Which features (from Table 6.2) are useful for ranking query suggestions?

Q5. How robust to missing relevance assessments is our evaluation framework?

In the following, Section 6.4.1 details the experimental setup that supports the investigation of these questions in Section 6.4.2.

6.4.1 Experimental Setup

In the remainder of this section, we describe the test collections and retrieval baselines used in our investigation, as well as the training procedure carried out to enable our proposed learning to rank approach for query suggestion.

6.4.1.1 Test Collections

Our experiments use the WT09, WT10, and WT11 test collections, described in Table 5.1, comprising 148 queries from the TREC 2009, 2010, and 2011 Web tracks (Clarke et al., 2009a, 2010, 2011b). To retrieve candidate suggestions for each query, we use the MSN 2006 query log, a one-month log with 15 million queries submitted by US users to MSN Search (now Bing) during spring 2006.[3] We index the structured virtual documents produced from this log using Terrier (Macdonald et al., 2012a) with positional information, so as to enable the extraction of proximity features, as discussed in Section 6.2.3. In particular, we apply Porter's weak stemming and do not remove stopwords. Finally, sessions are determined using a standard 30 min timeout. In addition, sessions with more

[3] http://research.microsoft.com/en-us/um/people/nickcr/wscd09

than 50 queries are discarded, as they are likely produced by robots (Silvestri, 2010). Salient statistics of the MSN 2006 query log are provided in Table 6.3.

Table 6.3: Salient statistics of the MSN 2006 query log.

#queries	14,921,285
#unique queries	6,623,635
#sessions	7,470,915
#clicks	12,251,067

Candidate suggestions are evaluated with respect to their performance at ranking documents from the category A portion of the ClueWeb09 corpus, described in Section 5.1.1. To this end, we use the Bing Search API as the reference adhoc retrieval system, by directly evaluating its returned URLs against those judged relevant in this corpus.[4] While using the Bing API provides a state-of-the-art reference retrieval system and is efficient enough to enable the large-scale evaluation conducted in this chapter, the rankings produced by using this API should be seen as a crude approximation of what Bing could achieve if restricted to searching only the ClueWeb09 corpus in the first place (Santos et al., 2011c). Nonetheless, Clarke et al. (2009a, 2010) have shown that rankings produced by a commercial search engine outperform almost all submitted runs in the TREC 2009 and 2010 Web tracks. Finally, as the reference diversification system, we employ our xQuAD framework, so as to assess the effectiveness of our produced suggestions when used as sub-queries for search result diversification.

6.4.1.2 Query Suggestion Baselines

To answer our first three research questions, we compare our proposed approach to two query suggestion baselines. The first of these is the approach of Broccolo et al. (2012), which inspired the suggestion representation adopted in this work, as described in Section 6.2.1. As discussed in Section 6.1, their approach was shown to perform at least as effectively as the state-of-the-art query-flow graph approach of Boldi et al. (2009a) for head queries, while consistently outperforming it for queries with little or no past evidence in the MSN 2006 query log. Hence, it

[4] All rankings were obtained in February 2012 using Bing API v2.0.

is used here as a representative of state-of-the-art query suggestion mechanisms. Additionally, we compare both our approach and that by Broccolo et al. (2012) to the query suggestions produced by the Bing Suggestion (BS) API.[5] While Bing can suggest queries not present in our test query logs (and arguably has suggestion models built from much larger query logs), this provides a reference performance for an industrial-strength query suggestion mechanism.

6.4.1.3 Training and Evaluation Procedures

To enable our learning approach, following the scheme formalised in Equation (6.1), we automatically label a pool of 105,325 suggested queries, comprising the union of all suggestion samples in Section 6.2.1 (e.g., suggestions retrieved based on different fields, or those that come from satisfactory sessions). For the labelling function Δ_r in Equation (6.1), we use nDCG@10, which is a typical target in a web search setting (Jansen et al., 1998). For our learning setup, following common practice (Qin et al., 2010), we consider a sample of $n_q = 1000$ suggestions retrieved for a given query using BM25 (Equation (2.13)). As a loss function, we use nDCG@100, in order to provide a more informative guidance to our learning process, by capturing swaps between relevant and non-relevant documents beyond our target evaluation cutoff ($\kappa = 10$) (Robertson, 2008). As learning algorithms, we employ the listwise AFS and LambdaMART algorithms, introduced in Section 2.2.3.2, by performing a 5-fold cross validation. To this end, we split the available queries into training (60%), validation (20%), and test (20%) sets. Accordingly, our results are reported on the test queries across all folds.

6.4.2 Experimental Results

In this section, we assess the effectiveness of our proposed learning to rank approach for the query suggestion problem, with applications to both adhoc and diversity search. To this end, Sections 6.4.2.1 through 6.4.2.5 address each of the five research questions stated in Section 6.4.1 in turn.

[5] All query suggestions were obtained in February 2012 using Bing API v2.0.

6.4.2.1 Adhoc Retrieval Performance

In order to address research question Q1, regarding the effectiveness of our query suggestion approach for adhoc search, we analyse both the impact of different criteria for producing an initial sample of candidate suggestions, as well as the improvements brought by our learning to rank approach.

Sampled Suggestions Before evaluating our learning approach to query suggestion, we assess the alternative choices introduced in Section 6.2.1 for producing suggestion samples. In particular, we analyse this question in light of three orthogonal dimensions. The first dimension concerns the sessions from which to mine candidate suggestions: all sessions vs. satisfactory sessions (i.e., those with a click on the last query). The second dimension concerns the queries from a given session to be indexed as candidate suggestions: all queries vs. the last one. Finally, the third dimension relates to the sources of evidence to index as fields for each candidate suggestion: the suggestion itself (Q), its co-session queries (S), its queries with a common click (C), or any combination of these three fields.

To assess the full potential of these sampling alternatives, Table 6.4 summarises their performance in terms of the number of relevant suggestions retrieved at maximum recall depth (i.e., RelRet@1000). For this investigation, relevance labels are defined as per Equation (6.1).[6] The significance symbols introduced in Section 5.1.2 are used to denote a significant difference compared to the best result in each column, which is highlighted in bold. From the table, regarding our first dimension of interest, we observe that indexing only queries from satisfactory sessions (as opposed to all sessions) leads to improved recall for BM25(Q), BM25(S), BM25(C), and BM25(QS). This corroborates the findings of Broccolo et al. (2012), by showing that such sessions are more likely to contain effective suggestions. However, for the remaining variants, namely, BM25(QC), BM25(SC), and BM25(QSC), using all sessions performs slightly better.

Regarding our second considered dimension, we observe that indexing only the last query in a session, as proposed by Broccolo et al. (2012), substantially

[6]Note that suggestions with a relevance label 1 (i.e., with a positive yet lower retrieval effectiveness than that attained by the initial query) are also considered, as they may bring useful evidence for the diversification scenario addressed in Section 6.4.2.2.

Table 6.4: Performance of different sampling strategies at ranking effective suggestions in terms of RelRet@1000, with suggestion relevance labels defined as per Equation (6.1). The representation used by Broccolo et al. (2012) is marked with a † symbol.

Sessions	All		Satisfactory	
Queries	All	Last	All	Last
BM25(Q)	73▼	68▼	76▼	69▼
BM25(S)	68▼	55▼	72▼	58▼
BM25(C)	60▼	59▼	61▼	60▼
BM25(QS)	96▼	91▼	102▼	†94°
BM25(QC)	92▼	85▼	91▼	83▼
BM25(SC)	104▼	84▼	105▼	82▼
BM25(QSC)	**115**	**101**	**117**	**98**

decreases performance, regardless of whether this session is satisfactory or not. Lastly, regarding our third dimension of interest, we observe an increase in recall as we combine more fields together, with QSC being the overall best combination. This shows that click evidence further improves the QS combination used by Broccolo et al. (2012). However, taking into account the performance of individual fields can also be beneficial. As shown in Table 6.4, the Q field is the most effective, with S and C showing a similar performance. Recalling research question Q1, on the effectiveness of our approach for adhoc search, we conclude that mining suggestions among all queries in satisfactory sessions, and considering a multi-field representation (i.e., QSC), particularly with the added click evidence, provides the most effective sampling for learning to rank query suggestions.

Learned Suggestions After investigating alternative strategies for building an initial sample of suggestions for a query, we analyse whether this sample can be further improved by our learning to rank approach. In particular, we focus on the most promising samples identified in our previous experiment, namely, those comprising all queries from satisfactory sessions (the penultimate column in Table 6.4). For this investigation, we instantiate our proposed evaluation framework described in Section 6.3 and report our results in terms of s-nDCG$_\Psi$@k,10—i.e., the summary ("max" or "avg") document retrieval performance (in terms of the standard nDCG@10) attained by the top k ranked suggestions. As discussed in

Section 6.3, we test two values of k, representing two distinct scenarios. In the first scenario, we set $k = 1$, which assesses the effectiveness of each query suggestion mechanism at providing a single suggestion that could be used, e.g., for an automatic reformulation of the initial query. In the second scenario, we set $k = 8$, which is the maximum number of suggestions retrieved by the Bing Suggestion API as well as by the search interfaces of current commercial web search engines, and hence represents a typical application of query suggestion.

Table 6.5 shows the results of this investigation. In each cell, up to three symbols are used to denote statistically significant differences. For all suggestion mechanisms (i.e., BS, BM25, AFS, and LambdaMART), a first symbol denotes significance with respect to the initial query. For BM25, AFS, and LambdaMART, a second symbol denotes significance compared to the suggestions produced by BS. Lastly, for each of the considered samples (i.e., Q, S, C, QS, QC, SC, and QSC), a third symbol denotes whether AFS and LambdaMART differ significantly from the unsupervised BM25 baseline of Broccolo et al. (2012). The best performance on top of each sample is underlined, whereas the best overall performance across all samples is highlighted in bold.

From Table 6.5, we first observe that, compared to the BM25 variants, our learning approach using either AFS or LambdaMART consistently improves in all considered scenarios ($k = 1$ and $k = 8$ with both $\Psi =$ "max" and $\Psi =$ "avg"), with significant gains in most cases. Moreover, when retrieving multiple suggestions (i.e., $k = 8$), the suggestions produced by our approach are comparable to those provided by BS. Still compared to BS, significant gains are observed for the task of returning exactly one suggestion for automatically reformulating the initial query (i.e., $k = 1$). As discussed in Section 6.4.1.2, this is a remarkable result, particularly since the Bing API is not constrained to returning candidate suggestions from our one-month-long query log, instead arguably making use of much larger query logs. Lastly, compared to the initial query, no query suggestion mechanism improves for the task of finding a single effective suggestion ($k = 1$)—this is the case even for BS. Indeed, as denoted by the number of affected queries, all approaches harm substantially more queries than they improve, showing that an automatic reformulation of the initial query using the top suggestion would be risky. However, when the best performing suggestion among the top 8 is used

Table 6.5: Adhoc performance (in terms of s-nDCG$_{\max}$@1, 10, s-nDCG$_{\max}$@8, 10, and s-nDCG$_{\text{avg}}$@8, 10) attained by the suggestions produced by various mechanisms.

	s-nDCG$_{\max}$@k,10				s-nDCG$_{\max}$@k,10				s-nDCG$_{\text{avg}}$@k,10			
	$k=1$	$-$	$=$	$+$	$k=8$	$-$	$=$	$+$	$k=8$	$-$	$=$	$+$
Query	**0.115**				0.115				**0.115**			
BS	$0.048^{\blacktriangledown}$	110	14	24	$\mathbf{0.119}^{\circ}$	60	12	76	$0.045^{\blacktriangledown}$	126	9	13
BM25(Q)	$0.064^{\blacktriangledown\circ}$	79	59	10	$0.106^{\circ\circ}$	44	54	50	$0.039^{\blacktriangledown\circ}$	129	10	9
+AFS	$0.022^{\blacktriangledown\blacktriangledown\blacktriangledown}$	111	32	5	$0.043^{\blacktriangledown\blacktriangledown\blacktriangledown}$	99	27	22	$0.014^{\blacktriangledown\blacktriangledown\blacktriangledown}$	130	11	7
+LambdaMART	$\underline{0.074}^{\blacktriangledown\blacktriangle\circ}$	65	74	9	$\underline{0.112}^{\circ\circ\circ}$	31	63	54	$\underline{0.045}^{\blacktriangledown\circ\triangle}$	127	11	10
BM25(S)	$0.020^{\blacktriangledown\blacktriangledown}$	125	12	11	$0.071^{\blacktriangledown\blacktriangledown}$	94	23	31	$0.018^{\blacktriangledown\blacktriangledown}$	132	10	6
+AFS	$\underline{0.067}^{\blacktriangledown\triangle\blacktriangle}$	68	65	15	$\underline{0.103}^{\circ\circ\blacktriangle}$	50	45	53	$\underline{0.038}^{\blacktriangledown\circ\blacktriangle}$	129	8	11
+LambdaMART	$0.057^{\blacktriangledown\circ\blacktriangle}$	91	45	12	$0.096^{\triangledown\triangledown\blacktriangle}$	57	46	45	$0.035^{\blacktriangledown\triangledown\blacktriangle}$	126	10	12
BM25(C)	$0.045^{\blacktriangledown\circ}$	114	21	13	$0.081^{\blacktriangledown\blacktriangledown}$	70	29	49	$0.032^{\blacktriangledown\blacktriangledown}$	128	12	8
+AFS	$\underline{0.072}^{\blacktriangledown\blacktriangle\blacktriangle}$	64	74	10	$\underline{0.097}^{\circ\circ\blacktriangle}$	46	49	53	$\underline{0.039}^{\blacktriangledown\circ\blacktriangle}$	126	10	12
+LambdaMART	$0.063^{\blacktriangledown\circ\blacktriangle}$	73	62	13	$0.095^{\triangledown\triangledown\blacktriangle}$	45	49	54	$\underline{0.039}^{\blacktriangledown\circ\blacktriangle}$	126	10	12
BM25(QS)	$0.043^{\blacktriangledown\circ}$	114	26	8	$0.091^{\blacktriangledown\blacktriangledown}$	69	38	41	$0.030^{\blacktriangledown\blacktriangledown}$	131	9	8
+AFS	$\underline{0.090}^{\blacktriangledown\blacktriangle\blacktriangle}$	50	82	16	$\underline{0.114}^{\circ\circ\blacktriangle}$	35	57	56	$\underline{0.048}^{\blacktriangledown\circ\blacktriangle}$	127	9	12
+LambdaMART	$0.074^{\blacktriangledown\blacktriangle\blacktriangle}$	67	62	19	$0.113^{\circ\circ\blacktriangle}$	36	57	55	$0.046^{\blacktriangledown\circ\blacktriangle}$	124	10	14
BM25(QC)	$0.048^{\blacktriangledown\circ}$	110	24	14	$0.091^{\triangledown\triangledown}$	59	36	53	$0.038^{\blacktriangledown\circ}$	129	10	9
+AFS	$\underline{0.086}^{\blacktriangledown\blacktriangle\blacktriangle}$	52	83	13	$\underline{0.112}^{\circ\circ\blacktriangle}$	36	58	54	$0.045^{\blacktriangledown\circ\blacktriangle}$	127	9	12
+LambdaMART	$0.077^{\blacktriangledown\blacktriangle\blacktriangle}$	63	71	14	$0.109^{\circ\circ\triangle}$	38	56	54	$\underline{0.046}^{\blacktriangledown\circ\blacktriangle}$	127	11	10
BM25(SC)	$0.046^{\blacktriangledown\circ}$	109	24	15	$0.084^{\blacktriangledown\blacktriangledown}$	68	32	48	$0.034^{\blacktriangledown\triangledown}$	127	12	9
+AFS	$\underline{0.082}^{\blacktriangledown\blacktriangle\blacktriangle}$	59	75	14	$\underline{0.108}^{\circ\circ\blacktriangle}$	41	57	50	$0.043^{\blacktriangledown\circ\blacktriangle}$	128	10	10
+LambdaMART	$0.066^{\blacktriangledown\triangle\blacktriangle}$	78	57	13	$0.105^{\circ\circ\blacktriangle}$	42	54	52	$\underline{0.045}^{\blacktriangledown\circ\blacktriangle}$	125	11	12
BM25(QSC)	$0.050^{\blacktriangledown\circ}$	105	28	15	$0.098^{\triangledown\circ}$	58	38	52	$0.038^{\blacktriangledown\circ}$	128	9	11
+AFS	$\underline{0.085}^{\blacktriangledown\blacktriangle\blacktriangle}$	53	83	12	$\underline{0.117}^{\circ\circ\blacktriangle}$	35	55	58	$\underline{0.047}^{\blacktriangledown\circ\blacktriangle}$	128	8	12
+LambdaMART	$0.078^{\blacktriangledown\blacktriangle\blacktriangle}$	67	64	17	$0.110^{\circ\circ\blacktriangle}$	39	55	54	$0.046^{\blacktriangledown\circ\blacktriangle}$	128	12	8

(i.e., $k = 8$ and $\Psi =$ "max"), both BS as well as one of the variants of our learning approach (namely, BM25(QSC)+AFS) are able to outperform the initial query, although not significantly. Still in light of research question Q1, the results in this section attest the effectiveness of our learning to rank approach compared to the state-of-the-art query suggestion approach of Broccolo et al. (2012) as well as to the industrial-strength suggestion mechanism provided by the Bing API.

6.4.2.2 Diversification Performance

In this section, we address research question Q2, by assessing the effectiveness of our produced suggestions when used for diversifying the retrieved documents. As discussed in Section 6.3, for this evaluation, we use two different instantiations of the *s-deval* metric defined in Equation (6.3), by leveraging the primary metrics for diversity search evaluation used in the TREC Web track (Clarke et al., 2011b): ERR-IA (Equation (3.28)) and α-nDCG (Equation (3.29)). In particular, we consider the scenario where a user would inspect the top $\kappa = 20$ documents, diversified by the xQuAD framework (as the reference diversification system) using the top k suggestions provided by each query suggestion mechanism as a set of sub-queries $\mathcal{S}_q$. As baselines for this investigation, we consider the initial query, as well as the suggestions produced by Bing (BS) and BM25(QSC), as the best performing unsupervised variant from Table 6.4. Table 6.6 shows the results of this investigation, with significance symbols defined as in Table 6.5.

Table 6.6: Diversification performance (in terms of both s-ERR-IA@8,20 and s-α-nDCG@8,20) attained by the suggestions produced by various mechanisms.

	$\mathcal{S}_q$	s-ERR-IA				s-α-nDCG			
		@8,20	−	=	+	@8,20	−	=	+
Bing	(Query)	0.382				0.502			
+xQuAD	BS	$0.406^{\blacktriangle}$	33	27	90	$0.524^{\blacktriangle}$	35	23	92
	BM25(QSC)	$0.403^{\triangle\circ}$	42	32	76	$0.521^{\blacktriangle\circ}$	41	30	79
	+AFS	$0.404^{\blacktriangle\circ\circ}$	41	21	88	$0.522^{\blacktriangle\circ\circ}$	38	22	90
	+LambdaMART	$\mathbf{0.412}^{\blacktriangle\circ\triangle}$	44	15	91	$\mathbf{0.527}^{\blacktriangle\circ\triangle}$	47	14	89

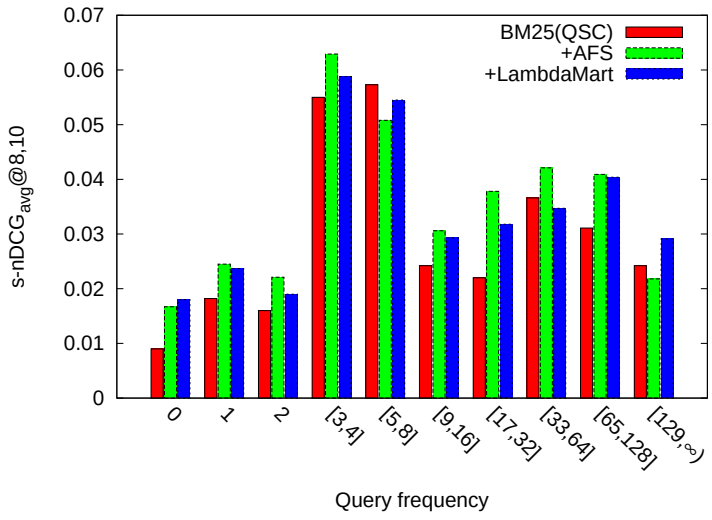
From Table 6.6, we first observe that both the unsupervised approach of Broccolo et al. (2012) using BM25 as well as our learning to rank approach using AFS and LambdaMART significantly improve upon the initial query, attesting the suitability of mining effective sub-queries from a query log to diversify a ranking of documents. Moreover, the performance attained by these approaches does not differ significantly from that attained by the suggestions produced by the Bing API. Once again, this is a remarkable result, given the substantially larger amount of data available to Bing compared to our one-month query log snapshot. Finally,

we observe that our approach consistently outperforms the strong baseline suggestion mechanism of Broccolo et al. (2012), with significant improvements when using LambdaMART. Overall, this answers research question Q2, by showing that our learning approach is also effective at providing query suggestions to be used for search result diversification.

6.4.2.3 Performance under Sparsity

An inherited characteristic of the query suggestion representation adopted by our approach is its resilience to sparse data. As discussed in Section 6.2.1, most existing query suggestion approaches suffer when there is limited session or click information for a given query. Instead, by indexing candidate suggestions at the term level, our approach improves the chance of identifying at least one of these suggestions as a potentially relevant match for even an unseen query, provided that the query and the suggestion share at least one term. In order to analyse the impact of query sparsity on the effectiveness of our proposed learning to rank approach for the query suggestion problem, Figure 6.3 breaks down the performance of our approach, as well as the approach of Broccolo et al. (2012), for input queries with different frequencies in the MSN 2006 query log.

Figure 6.3: Suggestion adhoc effectiveness (in terms of s-nDCG$_{avg}$@8,10) for queries with various frequencies in the MSN 2006 query log. Query frequencies are split into exponentially-sized bins, so that the number of queries in each bin is roughly balanced.



As shown in the figure, both of our learning to rank variants as well as the approach of Broccolo et al. (2012) are able to provide effective suggestions even for completely unseen queries (i.e., queries with a zero frequency in the query log, such as "*wedding budget calculator*"). While this resilience to sparsity comes mostly from the structured virtual document representation inspired by the approach of Broccolo et al. (2012), it is interesting to note that our learning variants further improve on top of their approach for almost the entire range of query frequencies. Recalling research question Q3, these results attest the robustness of our learning approach in light of data sparsity, and further corroborate the findings in Sections 6.4.2.1 and 6.4.2.2 regarding the effectiveness of our approach.

6.4.2.4 Feature Analysis

Besides breaking down the analysis of our approach for queries with different frequencies in a query log, in this section, we address research question Q4, by analysing which of the features in Table 6.2 are effective for learning to rank query suggestions. To this end, we measure the predictive power of each of these features individually. In particular, Table 6.7 lists the top 10 query-dependent and top 10 query-independent features, selected according to their correlation (Pearson's ρ) with the training labels, as defined in Equation (6.1).

Table 6.7: Top 10 query-dependent and query-independent features for learning to rank suggestions, ranked by their correlation (Pearson's ρ) with the learning labels.

Query-dependent features			Query-independent features		
Rank	Feature	ρ	Rank	Feature	ρ
1	CLM(QSC)	0.166	5	Tokens(Q)	0.078
2	BM25(QSC)	0.148	6	Chars	0.072
3	DPH(QSC)	0.104	18	Tokens(C)	0.029
4	PL2(QSC)	0.090	20	Clicks	0.022
7	PL2(S)	0.065	22	SessionClicks (s.d.)	0.021
8	pBiL(QSC)	0.063	23	SessionLength (mean)	0.020
9	CLM(C)	0.058	25	Digits (fraction)	0.019
10	CLM(S)	0.057	26	Terms	0.019
11	DPH(C)	0.054	29	SessionLength (s.d.)	0.015
12	BM25(C)	0.051	30	Badwords (presence)	0.014

From Table 6.7, we observe that 8 of the overall top 10 features are query-dependent. This observation highlights both the topical nature of the query suggestion task, as well as the benefit of leveraging evidence from multiple sources in a query log (i.e., the Q, S, and C fields), with an aggregation of all available evidence (i.e., the QSC combination) performing the best. Interestingly, CLM (Equation (2.5)) is the best feature, emphasising the importance of covering multiple terms from the input query. As for the top query-independent features, our learned suggestion models generally benefit from lexical features, such as the suggestion length in tokens or characters. In addition, features based on past usage behaviour, including session and click information, are also effective.

6.4.2.5 Robustness to Missing Relevance Assessments

As discussed in Section 6.3, our evaluation framework leverages document relevance assessments from the adhoc and diversity test collections of the TREC Web track (Clarke et al., 2009a, 2010, 2011b). As a result, the robustness of the framework directly depends on its ability to reuse the relevance assessments from these test collections. In particular, in our evaluation framework, the diversity document relevance assessments produced for a given query are directly reused to evaluate a different document ranking produced for the *same* query, namely, the ranking produced by using query suggestions as input to a reference diversification approach, such as xQuAD. On the other hand, the adhoc relevance assessments for a query are reused to assess the effectiveness of a ranking produced for *different* queries, i.e., each of the suggestions produced for the initial query. The latter scenario explicitly assumes a user with a clearly specified information need, hence considering query suggestion as the task of identifying effective replacements for the user's original query. However, the effectiveness of such replacement queries may be underestimated in our evaluation framework, simply because they can retrieve documents that were not judged at all for the initial query.

In order to address research question Q5, Table 6.8 shows the extent to which missing relevance assessments impact the reusability of the TREC 2009, 2010, and 2011 Web track assessments within our evaluation framework. In particular, we consider both the number of judged (J@10) and relevant (P@10) documents

among the top 10 documents retrieved for each suggestion in the BM25(QSC) sample, which served as the basis for most of the query suggestion mechanisms investigated in Sections 6.4.2.1 through 6.4.2.4. The per-query figures are summarised by multiple statistics and broken down according to the considered adhoc ($k = 1$ and $k = 8$) and diversity ($k = 8$) search scenarios. As a baseline for measuring the reusability of the TREC Web track relevance assessments, these figures are compared to a BM25 ranking for the initial query, which represents a typical use case of reuse of the TREC Web track assessments for evaluation. In addition, we also include a ranking produced by Bing, which was used as the reference retrieval system in this chapter, as discussed in Section 6.4.1.1.

Table 6.8: Ratio of judged (J@10) and relevant (P@10) documents among the top 10 documents retrieved by Bing for each of the suggestions produced by BM25(QSC).

Suggestions	Adhoc						Diversity	
BM25(QSC)	k=1		k=8				k=8	
	Ψ = "max"		Ψ = "max"		Ψ = "avg"			
	J@10	P@10	J@10	P@10	J@10	P@10	J@10	P@10
average	0.034	0.021	0.506	0.312	0.227	0.132	0.565	0.393
median	0.000	0.000	0.500	0.300	0.200	0.075	0.600	0.400
std. dev.	0.137	0.096	0.277	0.264	0.179	0.144	0.234	0.235
minimum	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
maximum	0.900	0.900	0.900	0.900	0.775	0.687	0.900	0.900

Query	Adhoc		Diversity	
	J@10	P@10	J@10	P@10
Bing	0.589	0.340	0.563	0.398
BM25	0.329	0.097	0.249	0.090

From the bottom half of Table 6.8 (the "Query" half), we observe that, despite not targeting the ClueWeb09 corpus exclusively, Bing attains a much higher coverage of judged (J@10) and relevant (P@10) documents than BM25 (adhoc: Bing's J@10 = 0.589, P@10 = 0.340 vs. BM25's J@10 = 0.329, P@10 = 0.589; diversity: Bing's J@10 = 0.563, P@10 = 0.398 vs. BM25's J@10 = 0.249, P@10 = 0.090). This highlights the importance of having a high performing reference retrieval system for evaluating the effectiveness of query suggestions in large web corpora

such as ClueWeb09. Moreover, it corroborates our choice in Section 6.4.1.1 for using the API of a commercial web search engine for this purpose.

Next, with specific regards to research question Q5, on the robustness of our proposed evaluation framework to missing document relevance assessments, from the top half of Table 6.8 (the "Suggestions" half), we observe that most of the considered search scenarios show a reasonable coverage of the relevance assessments leveraged from the TREC 2009, 2010, and 2011 Web tracks. In particular, the evaluation of the effectiveness of a set of query suggestions (adhoc scenario, $k = 8$) shows a high robustness to missing assessments, with a coverage of judged (J@10) and relevant (P@10) documents that compares favourably to that attained by a standard BM25 ranking produced for the initial query (average J@10 up to 0.506 vs. BM25's 0.329; average P@10 up to 0.312 vs. BM25's 0.097). The diversity scenario, in turn, shows an even higher reuse of the underlying document relevance assessments, with average figures of J@10 = 0.565 and P@10 = 0.393. Such a higher coverage is due to the fact that this scenario evaluates the effectiveness of a set of suggestions with respect to their impact in diversifying the ranking for the initial query, as opposed to evaluating each suggestion individually. The only exception is the adhoc search scenario that considers only the top ranked suggestion ($k = 1$) for evaluation (e.g., for automatically reformulating the user's original query), which exhibits a low reuse of the TREC Web track assessments, with an average fraction of judged and relevant documents of 0.034 and 0.021, respectively. The evaluation in this specific scenario could be made more robust by incorporating alternative evaluation methodologies that take into account assessment sparsity (Carterette et al., 2006, 2009a), and by conducting additional relevance assessments, e.g., through crowdsourcing (Alonso et al., 2008). Alternatively, the effectiveness of a set of suggestions could be evaluated based upon their combined ability to improve the adhoc performance of the original query, in a similar fashion to our conducted diversity evaluation, as defined by Equation (6.3). This could be achieved by diversifying the initial ranking, e.g., using the xQuAD diversification framework, in the same manner as we did for the diversity search scenario. Alternatively, one could simply enrich the initial ranking with documents retrieved for different query suggestions (Sheldon et al., 2011). We leave these investigations as directions for future work.

6.5 Summary

In this chapter, we have addressed the second claim from our thesis statement, by showing that an effective mechanism for generating sub-queries positively impacts the diversification performance of our xQuAD framework, introduced in Chapter 4. To this end, we proposed a learning to rank approach to score the relative importance of sub-queries, identified as multiple query suggestions mined from a sample of the query log of a commercial web search engine.

In Section 6.1, we provided an overview of existing approaches for the query suggestion problem, in order to lay the ground for our proposed learning to rank approach, introduced in Section 6.2. In particular, our approach represents candidate suggestions from a query log as structured virtual documents comprising terms from related queries with common clicks, in addition to those from common sessions, as proposed by previous research. Besides helping overcome data sparsity, this enriched representation enables multiple query-dependent features to be computed for each candidate suggestion. We have also proposed several query-independent features specifically targeted to identify quality suggestions. Finally, we have integrated all these features in order to automatically learn effective models for ranking candidate suggestions in response to a user's query.

To evaluate our approach, in Section 6.3, we introduced an evaluation framework that directly leverages document relevance assessments from existing web search evaluation campaigns, hence requiring no extra assessment efforts. In Section 6.4, we deployed this framework for quantitatively evaluating the effectiveness of query suggestions for two practical search scenarios, namely, to provide effective alternatives to the initial query, or to help diversify the documents retrieved for this query. Under this framework, we contrasted our learning approach to a state-of-the-art query suggestion baseline from the literature. In Sections 6.4.2.1 and 6.4.2.2, we showed that our approach significantly outperforms this baseline in both scenarios. For the diversification scenario, in Section 6.4.2.2, our produced suggestions were also statistically comparable to those produced by a commercial web search engine. This is a remarkable achievement, given that commercial search engines arguably use much larger query logs than the one-month log snapshot available to our approach.

As demonstrated in Section 6.4.2.3, another benefit of our approach is its resilience to data sparsity. Indeed, our approach is able to produce effective suggestions even for a previously unseen query, provided that this query shares at least one term with relevant suggestions (or other queries related to these suggestions) in the log. Regarding the representation of candidate suggestions, our investigation in Section 6.4.2.4 showed that features dependent on the input query (computed using terms from the suggestion itself, as well as those from other queries with a common session or click with the suggestion) are the most effective descriptors of effective suggestions, denoting the topical nature of this task. Nevertheless, query-independent features reflecting lexical characteristics of a suggestion (e.g., its length) or its usage history (e.g., the amount of clicks it received across sessions) were also effective. Finally, a comprehensive analysis in Section 6.4.2.5 showed the robustness of our proposed evaluation methodology for quantifying suggestion effectiveness in light of missing relevance assessments.

After introducing an effective mechanism for generating sub-queries, in the next chapter, we will discuss an effective mechanism for estimating the coverage of each retrieved document with respect to each identified sub-query. To this end, we will exploit the intent underlying each sub-query as a means to select the most appropriate ranking model to perform such estimations.

Chapter 7

Document Coverage

Chapter 6 highlighted the importance of identifying effective sub-queries in order to achieve an improved diversification performance. Another pillar for effectively instantiating the xQuAD framework is an accurate estimation of the relevance of each retrieved document with respect to each identified sub-query. From the perspective of a document, such an estimation gives a measure of *coverage*; from the perspective of a sub-query, this estimation gives a measure of *novelty*.

In this chapter, we hypothesise that the more refined xQuAD's underlying estimation of the relevance of a document with respect to multiple sub-queries, the more effective its diversification performance. To test this hypothesis, we exploit the *intent* underlying each sub-query—e.g., informational or navigational (Broder, 2002; Rose & Levinson, 2004)—which has been previously shown to affect the estimation of relevance (e.g., Kang & Kim, 2003; Geng et al., 2008; Peng et al., 2010). In particular, we introduce a classification approach to predict the effectiveness of multiple intent-aware ranking models for estimating the relevance of the retrieved documents to each identified sub-query. As a result of this prediction, we can either select the ranking model most likely to be effective, or merge multiple models by taking into account their predicted effectiveness.

In the remainder of this chapter, Section 7.1 describes the use of intents in web search. Section 7.2 introduces our intent-aware approach to search result diversification, which is thoroughly evaluated in Section 7.3. The results not only attest the effectiveness of our approach, but also show that an improved estimation of coverage and novelty leads to a significantly improved diversification.

7.1 Intents in Web Search

Not all information needs have the same underlying *intent*. In particular, Broder (2002) proposed a taxonomy of information needs in web search, categorising their intent according to three classes: *navigational*, denoting a need to find a specific document; *informational*, denoting a need for information about a topic, which may be covered on one or more documents; and *transactional*, denoting a need to perform a web-mediated activity. This taxonomy was later extended by Rose & Levinson (2004), who devised a hierarchy of intents stemming from the three broad classes proposed by Broder (2002). Nevertheless, Broder's taxonomy remains the most widely adopted in the literature (Calderón-Benavides, 2011).

Several ranking approaches have benefited from exploiting the intent underlying web search queries. Such *intent-aware* approaches can be categorised as to whether they rely on the classification of queries into predefined intents. For instance, query intent detection approaches classify a query with respect to a predefined set of intents. A ranking model trained for the predicted intent is then applied to rank documents for the query. In this vein, Kang & Kim (2003) showed that queries of distinct intents can benefit from intent-aware ranking models. A major shortcoming of this family of approaches, however, is the limited accuracy of existing intent detection mechanisms (Craswell & Hawking, 2004).

Instead of classifying a query into a predefined target intent, an alternative is to identify similar queries from a training set, and then apply a ranking model appropriate for this set. This approach has an advantage over a classification of queries based on a fixed set of intents, as queries of the same intent often benefit from different ranking models (Craswell & Hawking, 2004). For example, Geng et al. (2008) proposed an instance-based learning approach using k-nearest neighbour (k-NN) classification (Aha et al., 1991) to improve web search effectiveness. In their approach, a k-NN classifier was used to identify training queries similar to an unseen query. A ranking model was then learned using the identified neighbouring queries and applied to the unseen query. A more general approach was proposed by Peng et al. (2010). In their work, multiple ranking functions were chosen from a pool of candidate functions, according to their retrieval performance on training queries similar to an unseen query.

Our intent-aware approach, introduced in the next section, is similar in spirit to the approaches of Kang & Kim (2003), Geng et al. (2008), and Peng et al. (2010). On the other hand, while these approaches focused on inferring the intent of a *query*, we aim to infer the intent of different *sub-queries* underlying this query. Besides this difference in granularity, our intent-aware approach tackles a different search scenario, namely, search result diversification.

7.2 Intent-aware Search Result Diversification

As discussed in Section 4.1, in order to diversify the documents retrieved for a query, we must be able to estimate the relevance of each document with respect to the multiple possible information needs underlying this query. In Chapters 5 and 6, we have experimented with both unsupervised as well as supervised ranking models in order to perform such estimations. Nonetheless, in both cases, the same ranking model was applied uniformly for all sub-queries.

In this chapter, we argue that the relevance of a document to the information need underlying a particular sub-query may depend on the intent of this sub-query. Additionally, different sub-queries can feasibly represent information needs with different intents. For instance, consider the query "*led zeppelin*". Also assume that, using a sub-query generation mechanism, such as the learning approach introduced in Chapter 6, we identify the following sub-queries for this query: "*led zeppelin website*", "*led zeppelin downloads*", and "*led zeppelin biography*". Arguably, these sub-queries represent navigational, transactional, and informational needs underlying the initial query, respectively.

Queries with different intents have been shown to benefit from intent-aware ranking models, as discussed in Section 7.1. Likewise, we hypothesise that an explicit diversification of the retrieved documents may benefit from taking into account the intents of different sub-queries. For instance, relevance estimations computed with respect to the "*led zeppelin website*" sub-query could arguably be improved by applying a ranking model suitable for navigational queries, while "*led zeppelin biography*" and "*led zeppelin downloads*" could benefit from models suitable for informational and transactional queries, respectively.

7.2.1 Covering Multiple Intents

Given a query q, our ultimate goal is to maximise the diversity of the retrieved documents with respect to the multiple information needs underlying this query. For explicit diversification approaches—such as those introduced in Section 3.3—at the heart of this goal lies an estimation of how well each document satisfies each of these information needs. For xQuAD, as formalised in Section 4.2.1, this equates to estimating the probability $p(d|q,s)$ of observing the document d given the query q and the sub-query s. In this section, we propose a supervised learning approach to perform this estimation, by predicting the appropriateness of multiple intent-aware ranking models for each identified sub-query.

In order to formalise our approach, we further derive the probability $p(d|q,s)$, by marginalising it over a target set of intents $\mathcal{I}$, according to:

$$p(d|q,s) = \sum_{\iota \in \mathcal{I}} p(\iota|s)\, p(d|q,s,\iota), \tag{7.1}$$

where $p(\iota|s)$ is the probability that the sub-query $s \in \mathcal{S}_q$ conveys the intent ι. Accordingly, $p(d|q,s,\iota)$ denotes the relevance of the document d given the query q, the sub-query s, and the intent ι. As a consequence, in order to estimate the probability $p(d|q,s)$, our task becomes two-fold:

1. Infer the probability $p(\iota|s)$ of each intent ι given the sub-query s;
2. Learn an intent-aware model $p(d|q,s,\iota)$ for each predicted intent ι.

In Section 7.2.2, we propose a classification approach for the first task. For the second task, as we will show in Section 7.2.3, we resort to learning to rank.

7.2.2 Inferring Sub-Query Intents

In order to infer the probability of multiple intents for a sub-query, we propose a linear classification approach. In particular, given a sub-query s, our goal is to estimate the probability of an intent $\iota \in \mathcal{I}$ as:

$$p(\iota|s) = f(\mathbf{w} \cdot \mathbf{x}_s), \tag{7.2}$$

where $\mathbf{x}_s$ is a feature vector representing the sub-query s, and $\mathbf{w}$ is a weight vector, learned from labelled training data. The function f maps the dot product of the weight and feature vectors into the desired prediction outcome. In Section 7.2.2.1, we propose two classification regimes to instantiate this function. Section 7.2.2.2 describes our choices for labelling training data. Lastly, Section 7.2.2.3 describes the sub-query features used in this classification task.

7.2.2.1 Classification Regimes

We propose two regimes for instantiating the function f in Equation (7.2): *model selection* and *model merging*. The model selection regime performs a hard classification (Witten & Frank, 2005), by assigning each sub-query a single (i.e., the most likely) intent. For instance, for a target set of intents $\mathcal{I} = \{\iota_1, \iota_2, \iota_3\}$, a possible model selection outcome could be: $p(\iota_1|s) = 1, p(\iota_2|s) = 0, p(\iota_3|s) = 0$. In this example, the sub-query s would be associated with its most likely intent, ι_1, and only the ranking model $p(d|q, s, \iota_1)$ would have an impact on the estimated relevance of document d to the sub-query s. This regime resembles the selective ranking approaches described in Section 7.1, except that the most appropriate model is selected at the sub-query level, as opposed to the query level.

Our second regime, model merging, provides a relaxed alternative to model selection. In particular, it deploys a soft classification approach, in order to obtain a full probability distribution over the considered intents (Witten & Frank, 2005). For the above example, a possible outcome of this classification regime could be $p(\iota_1|s) = 0.6, p(\iota_2|s) = 0.3, p(\iota_3|s) = 0.1$. In this case, the relevance of a document d to the sub-query s would be estimated by a linear combination:

$$\begin{aligned} p(d|q, s) = & \ 0.6 \times p(d|q, s, \iota_1) \\ & + 0.3 \times p(d|q, s, \iota_2) \\ & + 0.1 \times p(d|q, s, \iota_3). \end{aligned}$$

Different classifiers can be used to implement both the model selection and model merging regimes. Further details about the specific classifiers that enable both regimes in our investigation are provided in Section 7.3.1.3.

7.2.2.2 Classification Labels

In order to determine the ground-truth intent of each sub-query, we investigate two alternative labelling strategies. The first one, denoted JUDG, relies on a manual classification performed by TREC assessors for the sub-topics underlying a query, as discussed in Section 3.4.1. Nevertheless, the differences between these sub-queries may go beyond their apparent characteristics. For instance, sub-queries with the same judged intent could still benefit from leveraging different ranking models (Craswell & Hawking, 2004). Additionally, judging the intent of different sub-queries may be costly for large training datasets.

To overcome these limitations, we propose a second labelling strategy, denoted PERF, aimed to automatically label training sub-queries. In particular, given a training query q with sub-queries $\mathcal{S}_q$, $|\mathcal{S}_q| = k$, and a set of target intents $\mathcal{I}$, we devise an oracle selection mechanism. According to a target evaluation metric, such a mechanism always chooses the most effective out of the $|\mathcal{I}|^k$ possible selections of the available models to be leveraged by a reference diversification approach for the k sub-queries underlying q. In our investigation in Section 7.3, we use ERR-IA@20 (Equation (3.28)) as the target evaluation metric, and xQuAD as the reference diversification approach. Although estimating this oracle may be infeasible for large values of k, it can be easily estimated for most practical settings. For instance, the maximum number of sub-topics per query in the TREC 2009, 2010, and 2011 Web tracks is $k = 8$. Moreover, if many more sub-queries were available for a particular query, less plausible ones could be discarded without much loss. Indeed, this is precisely what leading web search engines do when displaying only the top suggestions for a user's query, as discussed in Chapter 6. To avoid training xQuAD's diversification trade-off λ for evaluating each selection of intents, we instantiate xQuAD with a fixed $\lambda = 1$, which equates to the formulation of IA-Select (Equation (3.23)), as discussed in Section 4.4. Finally, it is worth noting that the entire labelling process is conducted offline.

7.2.2.3 Classification Features

In order to enable our investigation in Section 7.3, we restrict the space of target intents to navigational and informational ones, since the TREC test collections

used in our experiments have query aspects labelled with one of these intents, as described in Section 3.4.1. Based on this representation of intents, and inspired by research on related query analysis tasks, we devise a large feature set for classifying the intent of each sub-query, including features computed from the words in the sub-query itself, as well as from the top documents retrieved for this sub-query. These features are summarised in Table 7.1, and organised into four groups, as described in the remainder of this section. In total, we devise 838 features, computed as variants of 28 different feature classes, as highlighted in the penultimate column (the "variants" column) of Table 7.1. For instance, retrieval-based features are computed using five distinct ranking models (denoted "m" in the "variants" column), namely, CLM (Equation (2.5)), BM25 (Equation (2.13)), DPH (Equation (2.31)), PL2 (Equation (2.29)), and LM (Equation (2.25)). Additionally, these features are estimated at six rank cutoffs (denoted "c"): 1, 3, 5, 10, 50, and 100. Entity-oriented features are computed for up to four entity types (denoted "t"): persons, organisations, products, and locations.[1] Finally, distributional features (e.g., number of entities per document) are summarised with three statistics (denoted "s"): mean, standard deviation, and maximum.

Query Concept Identification (QCI) Navigational information needs typically seek more clearly defined targets, such as a particular website (Broder, 2002). To quantify the extent to which a given sub-query has a clearly defined target, we compute the number of distinct entities in the sub-query. Our intuition is that sub-queries mentioning multiple entities are less likely to be navigational. For named entity recognition, we employ an efficient dictionary-based approach (Santos et al., 2010c), backed up by a dictionary of entity names built from DBPedia 3.3,[2] with additional person names from the 1990 US Census.[3] Likewise, we compute the number of ambiguous entries in the ranking, represented by Wikipedia disambiguation pages, as such pages represent ambiguous concepts and their associated senses or interpretations (Sanderson, 2008).

[1] Locations are only used for the EntityCount feature.
[2] http://dbpedia.org
[3] http://www.census.gov

Table 7.1: Sub-query features used for intent detection.

	Feature	Description	Variants	Total
QCI	DisambCount	Number of disamb. pages	5m × 6c	30
	DisambSenses	Number of disamb. senses	5m × 6c × 3s	90
	EntityCount	Number of entities in the query	4t	4
QLM	QueryFrequency	Number of occurrences		1
	ClickEntropy	URL-level click entropy		1
	HostEntropy	Host-level click entropy		1
	ResultCount	Examined documents per session	3s	3
	ClickCount	Clicked documents per session	3s	3
	ReformCount	Reformulations per session	3s	3
	SessionDuration	Session duration (in sec.)	3s	3
QPP	AvICTF	Pre-retrieval predictor		1
	AvIDF	Pre-retrieval predictor		1
	AvPMI	Pre-retrieval predictor		1
	EnIDF	Pre-retrieval predictor		1
	Gamma1	Pre-retrieval predictor		1
	Gamma2	Pre-retrieval predictor		1
	QueryScope	Pre-retrieval predictor		1
	Terms	Pre-retrieval predictor		1
	Tokens	Pre-retrieval predictor		1
	ClarityScore	Post-retrieval predictor	5m × 6c	30
	QueryDifficulty	Post-retrieval predictor	5m × 6c	30
	QueryFeedback	Post-retrieval predictor	5m × 6c	30
QTC	CategoryCosine	Cosine over categories	5m × 6c × 3s	90
	CategoryCount	Number of categories	5m × 6c	30
	CategoryEntropy	Category entropy	5m × 6c	30
	ConceptCosine	Concept cosine	5m × 3t × 6c × 3s	270
	ConceptCount	Number of concepts	5m × 3t × 6c	90
	ConceptEntropy	Concept entropy	5m × 3t × 6c	90
	Grand total			838

Query Log Mining (QLM) Query logs provide valuable evidence for discriminating between informational and navigational intents. In order to exploit such evidence, we compute several sub-query features based on the MSN 2006 query log, previously described in Section 6.4.1.1. For instance, we count the raw frequency of sub-queries, as navigational sub-queries are generally more popular than informational ones. Likewise, informational sub-queries intuitively require

more effort from the users while inspecting the retrieved results. We quantify this intuition in terms of the number of retrieved documents examined and the time spent in doing so. In addition, we compute the click entropy (Clough et al., 2009), as a measure of the variability of clicks on the documents retrieved for the sub-query, as another indicator of an informational intent.

Query Performance Prediction (QPP) The intent of a sub-query may be reflected not only on the sub-query itself, but also on the documents retrieved for this sub-query. For instance, a low coherence of the top-retrieved documents could indicate a sub-query with an informational intent. This, in turn, can reflect on the performance of this sub-query when used in a retrieval system. To exploit this intuition, we build upon a large body of research on query performance prediction (Carmel & Yom-Tov, 2010) and leverage both pre- and post-retrieval predictors as sub-query features. In particular, pre-retrieval predictors—e.g., AvICTF, AvIDF, AvPMI, EnIDF, Gamma, and QueryScope (He & Ounis, 2006)—are solely based on statistics of the sub-query terms. In turn, post-retrieval predictors—e.g., ClarityScore (Cronen-Townsend et al., 2002), QueryDifficulty (Amati et al., 2004), and QueryFeedback (Zhou & Croft, 2007)—also leverage information from the documents retrieved for the sub-query.

Query Topic Classification (QTC) Informational needs intuitively involve broader concepts than navigational ones. To quantify this intuition, we devise several features based on concepts from two taxonomies derived from Wikipedia: categories and named entities. For the latter, we consider people, organisations, products, and locations. In particular, we represent the documents retrieved for each sub-query in the space of the concepts from either taxonomy. From this representation, we compute various distributional features, considering the number of concepts per document, the distance between pairs of documents, and the concept entropy of the entire ranking (Song et al., 2009).

7.2.3 Learning Intent-aware Ranking Models

In Section 7.2.2.1, we proposed two regimes for inferring an intent distribution $p(\iota|s)$ for each sub-query s. In this section, we propose a learning to rank approach for producing suitable intent-aware ranking models for each intent of s.

7.2.3.1 Model Learning

In order to produce an intent-aware model $p(d|q, s, \iota)$ for each intent ι underlying the sub-query s, we once again resort to machine learning. In particular, we deploy a large set of document features, and leave it to a learning to rank algorithm to generate ranking models optimised for different intents. To achieve this goal, each model is learned using the entire feature set, but with a different training set of queries for each target intent. Given the intents considered in our investigation (i.e., informational and navigational), we use two intent-targeted query sets from the TREC 2009 Million Query track (Carterette et al., 2009b). The first set contains 70 informational queries and the second set contains 70 navigational queries, as judged by TREC assessors. As a learning algorithm, we use AFS (Metzler, 2007), as described in Section 2.2.3.2. In our experiments, it is deployed to optimise mean average precision (MAP; Equation (2.50)).

7.2.3.2 Document Features

To enable the generation of effective intent-aware ranking models, we deploy a total of 60 document features, summarised in Table 7.2. Besides the query-dependent features previously described in Table 5.3, we include field-based extensions of BM25 and PL2, namely, BM25F (Zaragoza et al., 2004) and PL2F (Macdonald et al., 2006). As additional query-independent features, we include URL features—UD and UW, denoting the number of digits in the URL of the document and whether this URL comes from Wikipedia, respectively—and link analysis features—ER (Becchetti et al., 2006), denoting the likelihood that the outlinks of the document are reciprocated, and the score produced by the Absorbing Model (AM, Plachouras et al. (2005)), a link analysis algorithm based on absorbing Markov chains (Kemeny & Snell, 1960). In particular, each feature is computed for a sample of 5000 documents retrieved by DPH (Equation (2.31)).

Table 7.2: Document features used for learning intent-aware ranking models.

	Feature	Description	Equation	Total
Query-dependent	CLM	Full and per-field CLM score	(2.5)	5
	BM25	Full, per-field, and field-based BM25 score	(2.13)	6
	LM	Full and per-field LM score	(2.25)	5
	MRF	Full MRF score	(2.20)	8
	PL2	Full, per-field, and field-based PL2 score	(2.29)	6
	DPH	Full, and per-field DPH score	(2.31)	5
	pBiL	Full pBiL score	(2.32)	8
Query-independent	UC	Presence of host, domain, path, and query string	(2.33)	4
	UL	Length of URL host, path, and query string	(2.35)	3
	UD	Number of digits in the host and domain		2
	UW	Whether the URL is from Wikipedia		1
	HL	Ham (non-spam) likelihood	(2.42)	1
	ID	Indegree	(2.43)	1
	OD	Outdegree	(2.44)	1
	PR	Original and transposed PageRank score	(2.45)	2
	AM	Absorbing Model score		1
	ER	Edge reciprocity score		1
	Grand total			60

Table 7.3 lists the top 10 features as they were selected by AFS for each of our produced intent-aware models. For each feature, we show its attained performance in terms of MAP when combined with the features selected before it. From the table, we observe that the top features are generally intuitive. For instance, DPH (which is used to generate the learning sample) is the top feature for both models. Likewise, as expected, various URL and link analysis features (e.g., UW, UL, AM, PR, IL) are ranked high in the navigational model. Besides producing intuitive intent-aware models, we believe that our data-driven approach based on a large set of features provides a more robust alternative to hand-picking features traditionally associated with each intent. Lastly, it is worth noting that, although the choice of appropriate feature sets naturally depends on how learning instances (i.e., sub-queries) and labels (i.e., intents) are represented, our approach is agnostic to these representations. Indeed, while instantiating it for a different aspect representation or a different set of intents may require devising different features, no modification to the approach itself would be necessary.

Table 7.3: Top 10 document features in the informational and navigational models.

	Informational		Navigational	
	Feature	MAP	Feature	MAP
1	DPH	0.261	DPH	0.211
2	UD	0.275	MRF (body)	0.227
3	PL2 (title)	0.282	BM25 (title)	0.241
4	BM25 (field-based)	0.291	UW	0.252
5	pBiL (body)	0.296	CLM	0.259
6	pBiL (anchor)	0.298	UL	0.263
7	ER	0.300	AM	0.267
8	LM (title)	0.301	PR (transposed)	0.269
9	CLM (body)	0.302	IL	0.272
10	CLM	0.303	pBiL (body)	0.274

7.3 Experimental Evaluation

In this section, we address the third claim from our thesis statement:

> *"By maximising the relevance of the retrieved documents to multiple sub-queries, a high coverage of these sub-queries can be achieved."*

To address this claim, we evaluate the effectiveness of our intent-aware approach to improve the coverage estimates leveraged by the xQuAD framework.[4] In particular, we aim to answer the following research questions:

Q1. Can we improve diversification performance with our *model selection* regime?

Q2. Can we improve diversification performance with our *model merging* regime?

In the following, Section 7.3.1 details the experimental setup that supports the investigation of these questions, including the test collections, the diversification baselines, and the classification approaches used by the two regimes, as well as the procedure carried out for training and evaluating all approaches. The results of this investigation are discussed in Section 7.3.2.

[4]While the estimated relevance of a document with respect to a sub-query also impacts xQuAD's estimation of novelty, we leave the analysis of this component to Chapter 8.

7.3.1 Experimental Setup

In this section, we describe the specific setup that supports our investigation in Section 7.3.2, as an extension of the general methodology described in Section 5.1.

7.3.1.1 Test Collections

Our analysis is based on the WT09 and WT10 test collections, described in Table 5.1, comprising 49 and 48 queries from the diversity task of the TREC 2009 and 2010 Web tracks (Clarke et al., 2009a, 2010), respectively. For each of these 97 queries, we consider both the TREC Web track sub-topics (WT) as well as query suggestions provided by Bing (BS) as alternative sub-query sets. Both the WT and BS sub-query sets are described in Section 5.2.1.3. In particular, as discussed in Section 7.2.2.2, the WT sub-query set provides judged intent labels for each sub-query, which can be contrasted to our performance-oriented labelling of training data. Finally, as a document corpus, we consider the category B portion of ClueWeb09, as described in Section 5.1.1.

7.3.1.2 Diversification Baselines

As diversification baselines for the experiments in Section 7.3.2, we consider two deployments of our xQuAD framework. Each of these deployments uniformly applies one of the informational (Uni(inf)) or the navigational (Uni(nav)) models described in Section 7.2.3.1 for all sub-queries, regardless of the intent of each sub-query. Using either the Uni(inf) or the Uni(nav) model, xQuAD is deployed to diversify the top 1000 documents retrieved by the DPH ranking model (Equation (2.31)), which serves itself as a non-diversification, relevance-only baseline.

7.3.1.3 Classification Approaches

In Section 7.2.2, we introduced two regimes for exploiting the inferred intents of different sub-queries: model selection and model merging. The model selection regime builds upon a hard classification of intents. To enable a thorough evaluation, we consider variants of this regime of the form Sel(c,l), where c and l denote a classifier and a set of classification training labels, respectively. In

particular, C can be one of three classifiers: an oracle (ORA), which simulates a perfect classification of the intent of each sub-query, a support vector machine (SVM) classifier with a polynomial kernel (Platt, 1998), and a multinomial logistic regression (LOG) with a ridge estimator (le Cessie & van Houwelingen, 1992). Regarding the classification labels L, as described in Section 7.2.2.2, we consider both human judgements (JUDG) as well as the selection with best diversification performance (PERF) on the training data. In all cases, the single most likely intent is chosen for each sub-query, in a typical selective fashion. To enable our second regime, model merging (MRG(C,L)), we fit the output of the SVM classifier to a logistic regression model, hence obtaining a full probability distribution over intents for each aspect underlying the query (Witten & Frank, 2005). To cope with the high dimensionality of our sub-query feature set, classification is performed after a dimensionality reduction via principal component analysis (Pearson, 1901). All classification tasks are performed using Weka.[5]

7.3.1.4 Training and Evaluation Procedure

Our evaluation ensures a complete separation between training and test settings. In particular, we use the WT09 and WT10 queries interchangeably, in a cross-year evaluation fashion (i.e., we train on WT09 and test on WT10, and vice versa), in order to train the classification approaches described in Section 7.3.1.3. As described in Section 5.1.2, diversification performance is reported in terms of ERR-IA (Equation (3.28)) and α-nDCG (Equation (3.29)), with the symbols defined in that section denoting significance as verified by a paired t-test.

7.3.2 Experimental Results

In the remainder of this section, we evaluate our intent-aware diversification approach, in order to answer the two research questions stated in Section 7.3. In particular, Section 7.3.2.1 evaluates our model selection regime, whereas Section 7.3.2.2 evaluates our model merging regime.

[5] http://www.cs.waikato.ac.nz/ml/weka/

7.3.2.1 Intent-aware Model Selection

In order to address Q1 from Section 7.3, we assess the effectiveness of our model selection regime, introduced in Section 7.2.2.1, in contrast to the uniform regimes described in Section 7.3.1.2. To this end, Table 7.4 compares the diversification performance of xQuAD using the aforementioned regimes with the WT and BS sub-queries. For all deployments of xQuAD, a first significance symbol denotes a significant difference (or lack thereof) compared to the DPH baseline. For deployments using our model selection regime (Sel(•,•)), two additional symbols denote significance with respect to the informational (Uni(inf)) and navigational (Uni(nav)) uniform regimes, respectively. The best variant for each classification label (i.e., judg and perf) is underlined. The best overall variant is in bold.

Table 7.4: Diversification performance of xQuAD using informational (inf) or navigational (nav) models uniformly (Uni) or selectively (Sel).

	$\mathcal{S}_q$	$p(d\|q,s)$	ERR-IA				α-nDCG			
			@20	−	=	+	@20	−	=	+
DPH			0.178				0.282			
+xQuAD	WT	Uni(inf)	$0.215^{\triangle}$	35	9	53	$0.331^{\blacktriangle}$	31	9	57
+xQuAD	WT	Uni(nav)	$\underline{0.247}^{\blacktriangle}$	32	6	59	$\underline{0.358}^{\blacktriangle}$	29	6	62
+xQuAD	WT	Sel(log,judg)	$0.241^{\blacktriangle\circ\circ}$	31	6	60	$0.354^{\blacktriangle\circ\circ}$	31	6	60
+xQuAD	WT	Sel(svm,judg)	$\underline{0.244}^{\blacktriangle\circ\circ}$	32	6	59	$0.357^{\blacktriangle\circ\circ}$	30	6	61
+xQuAD	WT	Sel(ora,judg)	$\underline{0.244}^{\blacktriangle\triangle\circ}$	34	7	56	$\underline{0.362}^{\blacktriangle\triangle\circ}$	31	7	59
+xQuAD	WT	Sel(log,perf)	$0.269^{\blacktriangle\blacktriangle\triangle}$	26	6	65	$0.382^{\blacktriangle\blacktriangle\triangle}$	27	6	64
+xQuAD	WT	Sel(svm,perf)	$0.265^{\blacktriangle\blacktriangle\circ}$	26	6	65	$0.380^{\blacktriangle\blacktriangle\triangle}$	27	6	64
+xQuAD	WT	Sel(ora,perf)	$\underline{\mathbf{0.304}}^{\blacktriangle\blacktriangle\blacktriangle}$	21	5	71	$\underline{\mathbf{0.425}}^{\blacktriangle\blacktriangle\blacktriangle}$	22	5	70
+xQuAD	BS	Uni(inf)	0.202°	34	12	51	0.308°	37	11	49
+xQuAD	BS	Uni(nav)	$\underline{0.235}^{\blacktriangle}$	27	7	63	$\underline{0.343}^{\blacktriangle}$	28	7	62
+xQuAD	BS	Sel(log,perf)	$0.240^{\blacktriangle\triangle\circ}$	27	6	64	$0.354^{\blacktriangle\blacktriangle\circ}$	27	6	64
+xQuAD	BS	Sel(svm,perf)	$0.241^{\blacktriangle\blacktriangle\circ}$	25	6	66	$0.355^{\blacktriangle\blacktriangle\circ}$	29	6	62
+xQuAD	BS	Sel(ora,perf)	$\underline{0.292}^{\blacktriangle\blacktriangle\blacktriangle}$	20	6	71	$\underline{0.414}^{\blacktriangle\blacktriangle\blacktriangle}$	18	6	73

From Table 7.4, we first note that Uni(inf) and Uni(nav) provide a strong baseline performance, with significant gains compared to the non-diversified DPH baseline in almost every setting. To see whether our model selection regime can

improve upon these strong baselines, we first look at the performance of this regime using human-judged intents as classification labels, i.e., the SEL(•,JUDG) variants. As observed from Table 7.4, no instance of SEL(•,JUDG) can significantly outperform both UNI(INF) and UNI(NAV). This is the case even for SEL(ORA,JUDG), which deploys an oracle classifier. This observation demonstrates that human judgements provide a suboptimal labelling criterion. As discussed in Section 7.2.2.2, this further confirms our intuition that the appropriateness of an intent-aware retrieval model for a given sub-query cannot be effectively judged purely on the basis of the apparent characteristics of this sub-query. On the other hand, the variants that use performance-oriented labels, i.e., SEL(•,PERF), bring consistent and substantially larger improvements. Indeed, our model selection regime using both logistic regression (i.e., SEL(LOG,PERF)) and support vector machines (i.e., SEL(SVM,PERF)) always improves compared to a uniform regime, often significantly. For instance, considering the WT sub-queries, compared to the stronger UNI(NAV) baseline, improvements for the SEL(LOG,PERF) variant are as high as 8.9% (0.269 vs. 0.247) in terms of ERR-IA@20, and 6.7% (0.382 vs. 0.358) in terms of α-nDCG@20. Similar improvements for the SEL(SVM,PERF) variant are also consistently observed. When the BS sub-query set is considered, although the observed improvements are less pronounced, they are consistent and can still be significant.

Overall, the results in this section answer research question Q1, by showing that diversification performance can be significantly improved by our model selection regime, which chooses the most appropriate intent-aware ranking model for each sub-query. The variants of this regime using performance-oriented labels (i.e., the SEL(•,PERF) variants) are particularly effective, significantly improving upon strongly performing uniform regimes trained on informational and navigational queries. Furthermore, the consistency of our observations for multiple evaluation metrics attests the robustness of the model selection regime. In the next section, we will contrast this regime against the model merging regime.

7.3.2.2 Intent-aware Model Merging

After demonstrating the effectiveness of selecting a single ranking model for each sub-query with our model selection regime, in this experiment, we address research question Q2 from Section 7.3, by investigating whether deploying our model merging regime could bring further improvements. As discussed in Section 7.3.1.3, both regimes are based on the predictions given by an SVM classifier. In particular, the model merging regime is enabled by fitting the SVM predictions to a logistic regression model. For this particular investigation, we focus our attention to the WT sub-queries, as they allow for assessing the effectiveness of our merging regime across the two proposed training labelling alternatives, JUDG and PERF. The results based on BS sub-queries using PERF labels lead to identical conclusions and are hence omitted for brevity. In particular, Table 7.5 shows the diversification performance of xQuAD under the model merging regime (MRG(SVM,PERF)), in contrast to its performance under the model selection regime (SEL(SVM,PERF)). Once again, a first significance symbol for both regimes denotes a significant difference (or lack thereof) compared to DPH. A second symbol for MRG(SVM,PERF) denotes significance compared to SEL(SVM,PERF), which serves as a further baseline for this investigation.

Table 7.5: Diversification performance of xQuAD using informational (INF) or navigational (NAV) models selectively (SEL) or through merging (MRG).

	$\mathcal{S}_q$	$p(d\|q,s)$	ERR-IA				α-nDCG			
			@20	−	=	+	@20	−	=	+
DPH			0.178				0.282			
+xQuAD	WT	SEL(SVM,JUDG)	0.244$^{\blacktriangle}$	32	6	59	0.357$^{\blacktriangle}$	30	6	61
+xQuAD	WT	MRG(SVM,JUDG)	0.255$^{\blacktriangle\circ}$	29	6	62	0.368$^{\blacktriangle\circ}$	28	6	63
+xQuAD	WT	SEL(SVM,PERF)	0.265$^{\blacktriangle}$	26	6	65	0.380$^{\blacktriangle}$	27	6	64
+xQuAD	WT	MRG(SVM,PERF)	**0.268**$^{\blacktriangle\circ}$	26	6	65	**0.381**$^{\blacktriangle\circ}$	27	6	64
+xQuAD	BS	SEL(SVM,PERF)	0.241$^{\blacktriangle}$	25	6	66	0.355$^{\blacktriangle}$	29	6	62
+xQuAD	BS	MRG(SVM,PERF)	0.237$^{\blacktriangle\circ}$	24	6	67	0.352$^{\blacktriangle\circ}$	27	6	64

From Table 7.5, we observe that the model merging regime can improve upon the model selection regime in most cases. In particular, when using JUDG labels,

we observe improvements of 4.3% (0.255 vs. 0.244) in terms of ERR-IA@20, and 3.1% (0.368 vs. 0.357) in terms of α-nDCG@20. With PERF labels, lower and inconsistent differences are observed, with the merging regime performing slightly better for the WT sub-queries and the selection regime performing better for BS sub-queries. Nevertheless, the observed differences between the two regimes are not statistically significant. These results answer research question Q2, by showing that merging multiple intent-aware ranking models can be at least as effective as selecting the single most effective model. Moreover, we believe that the merging regime can offer additional benefits for an intent-aware diversification. For one, it can help attenuate the harm of selecting the wrong model for a particular sub-query. Additionally, it provides a natural upper-bound for the selection regime. Indeed, model selection is a special instance of model merging, with a mutually exclusive probability distribution of intents $p(\iota|s)$.

7.4 Summary

In this chapter, we have addressed the third claim of our thesis statement, by showing that an improved estimate of the relevance of a document with respect to each sub-query leads to an improved coverage of this sub-query. In turn, an improved coverage of multiple sub-queries leads to an improved diversification performance, as demonstrated using our xQuAD framework, introduced in Chapter 4. As a means to improve coverage estimates, we built upon previous research on query intent detection for web search. In particular, we proposed to leverage ranking models that estimate the relevance of a document with respect to each sub-query by taking into account the intent of this sub-query.

In Section 7.1, we provided background on the categorisation of intents in web search, and described ranking approaches from the literature that successfully exploited intent information in order to improve search effectiveness. In Section 7.2, we proposed two classification regimes for leveraging intent-aware ranking models according to the predicted intent of each sub-query: model selection, which applies a single model given the most likely intent of each sub-query, and model merging, which combines relevance estimates produced by multiple models proportionally to the likelihood of each intent for a particular sub-query.

The model selection and model merging regimes were thoroughly evaluated in Section 7.3. In particular, in Section 7.3.2.1, our experiments showed that the model selection regime, choosing between an informational and a navigational ranking models on a per-sub-query basis, significantly outperforms each of these models when applied uniformly for all sub-queries, regardless of their predicted intent. In addition, in Section 7.3.2.2, we showed that the model merging regime, which mixes the scores produced by the informational and the navigational models, performs at least as effectively as the model selection regime.

Arguably, refined relevance estimates with respect to a sub-query could provide not only an improved estimate of the coverage of a document that satisfies this sub-query, but also an improved estimate of the novelty of any further document satisfying this sub-query, given the previously ranked documents. Hence, it is not clear whether the gains in diversification performance observed in this chapter are merely due to an improved estimation of coverage, or whether novelty also plays a role. Investigating this question is the purpose of the next chapter.

Chapter 8

Document Novelty

The previous chapter showed that an improved diversification can be achieved by improving the estimation of the relevance of each retrieved document with respect to each identified sub-query. For hybrid diversification approaches, such as xQuAD, this estimation can be leveraged to compute both the coverage and the novelty of a document. Nevertheless, it is not clear how coverage and novelty interplay, or what the role of novelty is when diversifying the search results.

In this chapter, we challenge the common view of novelty as an intuitive diversification strategy, and thoroughly assess the impact of this strategy in contrast to and in combination with coverage. To this end, Section 8.1 briefly recaps on our definitions of aspect representation and diversification strategy, as introduced in Section 3.3. Section 8.2 proposes a unifying methodology to enable the direct comparison of existing diversification approaches across these two dimensions. Following the proposed methodology, in Section 8.3, we thoroughly investigate the role of novelty as a diversification strategy, through both an empirical evaluation as well as through simulations. Our results show that existing approaches based solely on novelty cannot consistently improve upon a non-diversified baseline ranking. Moreover, when deployed as an additional component by hybrid approaches, we show that novelty does not bring significant improvements, while adding considerable efficiency overheads. Finally, through a comprehensive analysis with simulated rankings of various quality, we demonstrate that, although inherently limited by the performance of the initial ranking, novelty plays a role at breaking the tie between documents with similar coverage scores.

8.1 Diversification Dimensions

The most prominent diversification approaches in the literature can be organised according to two orthogonal dimensions, as proposed in Section 3.3: aspect representation and diversification strategy. The *aspect representation* determines whether the possible information needs underlying a query are represented *explicitly*, based upon properties of the query itself (e.g., query reformulations or categories), or *implicitly*, based upon properties of the retrieved documents (e.g., the terms comprised by each document). In turn, the *diversification strategy* determines how a particular aspect representation is leveraged to diversify the retrieved documents. In particular, *novelty*-based approaches achieve this goal by comparing the retrieved documents to one another, in order to promote those that carry new information. In contrast, *coverage*-based approaches directly estimate how well each document covers the identified query aspects. Finally, *hybrid* approaches combine the goals of coverage and novelty into a unified strategy.

Unfortunately, the prevalence of different aspect representations has precluded a direct comparison between coverage and novelty. As a result, it remains unclear whether the striking difference in performance commonly observed between coverage and novelty-based approaches is due to their underlying aspect representation (explicit vs. implicit) or to their diversification strategy (coverage vs. novelty). It is also unclear how much novelty actually contributes to the effectiveness of hybrid approaches, while penalising their efficiency. Although intuitive, novelty has yet to be shown effective for diversifying web search results. In particular, existing evidence of the effectiveness of novelty as a diversification strategy is based on either qualitative studies (Carbonell & Goldstein, 1998) or on curated corpora, such as Wikipedia (Rafiei et al., 2010) or newswire (Wang & Zhu, 2009).

To allow a thorough investigation of the role of novelty for search result diversification, in the next section, we adapt two existing novelty-based approaches to leverage explicit query aspect representations. Likewise, we produce coverage-only versions of two approaches that deploy a hybrid of coverage and novelty, including our xQuAD framework. By doing so, we bridge the gap between the diversification approaches in the literature and enable their evaluation in terms of the aspect representation and the diversification strategy dimensions.

8.2 Bridging the Gap

Although having the same goal of producing a diverse ranking, coverage and novelty-based approaches pursue this goal in rather distinct manners. In particular, purely coverage-based approaches ignore the set of already selected documents when scoring a given document. In turn, purely novelty-based approaches ignore the possible information needs underlying a query when comparing the contents of the retrieved documents. In practice, the distinct aspect representations leveraged by the existing approaches renders coverage and novelty not directly comparable. In this section, we describe our methodology to bridge the gap between these approaches and enable their direct comparison. Besides evaluating novelty in contrast to and in combination with coverage, our goal is to isolate these strategies from their underlying aspect representation, so as to provide a controlled setting for our investigation. To this end, in Section 8.2.1, we propose adaptations of two implicit novelty-based diversification approaches to leverage explicit aspect representations. Additionally, in Section 8.2.2, we deconstruct two explicit hybrid approaches to deploy a coverage-based strategy only.

8.2.1 Explicit Novelty-based Diversification

Existing novelty-based diversification approaches rely on an implicit aspect representation to estimate the diversity of a document with respect to the other retrieved documents (e.g., Carbonell & Goldstein, 1998; Zhai et al., 2003; Wang & Zhu, 2009). As a result, these approaches compare documents purely on the basis of their content, rather than based on how these documents satisfy the possible information needs underlying the query. Moreover, the resulting document representation (e.g., in the term-frequency space of a given corpus) is usually high-dimensional, which negatively impacts both the effectiveness and the efficiency of these approaches (Witten & Frank, 2005, Section 7.1). To counter these limitations and—more importantly for the investigation in this chapter—to enable a direct comparison of existing diversification approaches across both the aspect representation and the diversification strategy dimensions, we propose to leverage explicit aspect representations for estimating novelty. Besides providing a more expressive account of the relationship between documents and the aspects

they cover, this representation also has a considerable impact on efficiency, since the feature space is reduced from the size of the corpus vocabulary (millions) to the number of aspects underlying a query (around a dozen).

Given a query q with a set of aspects $\mathcal{A}$, with $|\mathcal{A}| = k$, we explicitly represent each retrieved document $d \in \mathcal{R}_q$ as a k-dimensional vector $\mathbf{d}$ over the aspects $\mathcal{A}$. In particular, the i-th dimension of the vector $\mathbf{d}$ is defined as:

$$\mathbf{d}_i = f(d, a_i), \tag{8.1}$$

where the function f estimates how well the document d satisfies the aspect $a_i \in \mathcal{A}$. As discussed in Section 5.2.1.3, different measures of the document-aspect association can be used, depending on how the aspects underlying the query are identified, e.g., based on reformulations mined from a query log or on categories derived from a classification taxonomy. Regardless of the particular mechanism used to identify the aspects of a query, an explicit representation of documents with respect to these aspects can be seamlessly integrated into existing novelty-based diversification approaches. In particular, to enable our analysis in Section 8.3, we derive explicit versions of two well-known novelty-based approaches in the literature, namely, Maximal Marginal Relevance (MMR; Carbonell & Goldstein, 1998) and Mean-Variance Analysis (MVA; Wang & Zhu, 2009).

Both MMR and MVA deploy the greedy diversification approach formalised in Algorithm 3.1. As discussed in Section 3.3.1, given an initial ranking $\mathcal{R}_q$ for the query q, these approaches iteratively build a diverse re-ranking $\mathcal{D}_q$. To this end, at each iteration, MMR (Equation (3.8)) instantiates the objective function $f(q, d, \mathcal{D}_q)$ in Algorithm 3.1 by estimating the similarity between each candidate document $d \in \mathcal{R}_q \setminus \mathcal{D}_q$ and its most dissimilar document $d_j \in \mathcal{D}_q$. Likewise, we devise xMMR (Explicit Maximal Marginal Relevance) to estimate novelty over explicit representations of the retrieved documents, according to:

$$f_{\text{xMMR}}(q, d, \mathcal{D}_q) = \lambda f_1(q, d) - (1 - \lambda) \max_{\mathbf{d}_j \in \mathcal{D}_q} f_2(\mathbf{d}, \mathbf{d}_j), \tag{8.2}$$

where $f_1(q, d)$ and $f_2(\mathbf{d}, \mathbf{d}_j)$ estimate the relevance of d with respect to the query q and its similarity to the documents already in $\mathcal{D}_q$, respectively. A balance be-

tween relevance (i.e., $f_1(q,d)$) and redundancy (i.e., $\max_{\mathbf{d}_j} f_2(\mathbf{d},\mathbf{d}_j)$, the opposite of novelty) is achieved through an appropriate setting of λ, as will be described in Section 8.3.1.3. In our experiments, $f_1(q,d)$ is estimated by a standard retrieval model. In order to estimate $f_2(\mathbf{d},\mathbf{d}_j)$, we compute the cosine between explicit representations of $\mathbf{d}$ and $\mathbf{d}_j$ over the set of aspects $\mathcal{A}$.

Analogously to MMR, MVA (Equation (3.11)) instantiates the objective function $f(q,d,\mathcal{D}_q)$ in Algorithm 3.1 by trading off relevance and redundancy. However, instead of computing the similarity between documents, MVA estimates the redundancy of a document based on how its relevance scores correlate to those of the other documents. Accordingly, we devise xMVA (Explicit Mean-Variance Analysis) to estimate these correlations based on how well the documents satisfy the explicitly represented query aspects. The objective function of xMVA is defined according to the following equation:

$$f_{\text{xMVA}}(q,d,\mathcal{D}_q) = \mu_d - b\, w_i\, \sigma_d^2 - 2\, b\, \sigma_d \sum_{d_j \in \mathcal{D}_q} w_j\, \sigma_{d_j}\, \rho_{\mathbf{d},\mathbf{d}_j}, \tag{8.3}$$

where μ_d and σ_d^2 are the mean and variance of the relevance estimates associated to document d, respectively, while the summation component estimates the redundancy of this document given the documents in $\mathcal{D}_q$. In particular, documents are compared in terms of their correlation $\rho_{\mathbf{d},\mathbf{d}_j}$. A balance between relevance, variance, and redundancy is achieved through the parameter b. Following Wang & Zhu (2009), μ_d is estimated by a standard retrieval model, with relevance scores normalised to yield a probability distribution, while σ_d is set as a constant for all documents. In our experiments, both σ and b are set through training, as will be described in Section 8.3.1.3. Finally, $\rho_{\mathbf{d},\mathbf{d}_j}$ is estimated as the Pearson's correlation between explicit representations of $\mathbf{d}$ and $\mathbf{d}_j$ over the aspects $\mathcal{A}$.

8.2.2 Explicit Coverage-based Diversification

Besides making coverage and novelty directly comparable by introducing explicit novelty-based diversification approaches (i.e., xMMR and xMVA), we want to be able to assess the effectiveness of novelty when combined with coverage. To this end, we deconstruct two hybrid diversification approaches, namely, IA-

Select (Agrawal et al., 2009) and our xQuAD framework, introduced in Chapter 4. Our ultimate goal is to produce directly comparable versions of these approaches, which should deploy coverage as their only strategy.

IA-Select (Equation (3.23)) was originally proposed to diversify the retrieved documents according to a predefined taxonomy, such as the one provided by the Open Directory Project (ODP). As a measure of novelty, IA-Select estimates the marginal utility $f(a_i|q, \mathcal{D}_q)$ of each query aspect $a_i \in \mathcal{A}$, represented by a taxonomy category, given the query q and the documents already in $\mathcal{D}_q$. The function $f(a_i|q, \mathcal{D}_q)$ incorporates both the relative importance of the aspect a_i in light of all aspects $\mathcal{A}$, as well as the utility of a_i, in light of the aspects already covered by the documents in $\mathcal{D}_q$. In essence, this function emulates a novelty component, by estimating how much the already selected documents satisfy each aspect of the query. To produce a coverage-only version of IA-Select, we assume that the query aspects do not lose their utility even if they are already covered by the documents in $\mathcal{D}_q$. In practice, this is achieved simply by dropping the term $\mathcal{D}_q$ in the marginal utility $f(a_i|q, \mathcal{D}_q)$, according to:

$$f_{\text{IA-Select}^*}(q, d, \mathcal{D}_q) = \sum_{a_i \in \mathcal{A}} f(a_i|q)\, f(d|q, a_i), \tag{8.4}$$

where $f(a_i|q)$ denotes the relative importance of the aspect a_i given the query q, and $f(d|q, a_i)$ denotes the extent to which this aspect is covered by the document d. To emphasise its difference from the standard formulation of IA-Select in Equation (3.23), we refer to this coverage-only version as IA-Select*.

Different from IA-Select, xQuAD (Equation (4.1)) implements the objective function $f(q, d, \mathcal{D}_q)$ in Algorithm 3.1 as a mixture of two probabilities: the probability $p(d|q)$ of the document d being relevant, and the probability $p(d, \bar{\mathcal{D}}_q|q)$ of d being diverse. The novelty component of xQuAD can be exposed by further expanding the latter probability, as demonstrated in Equations (4.2) through (4.6). In particular, xQuAD estimates the novelty of any document satisfying a given aspect $a_i \in \mathcal{A}$, represented as a sub-query, as the probability $p(\bar{\mathcal{D}}_q|q, a_i)$ that none of the already selected documents in $\mathcal{D}_q$ is relevant to this aspect. Analogously to our adaptation of IA-Select, we introduce a coverage-only version of xQuAD by assuming that all query aspects retain their utility, regardless of the documents

previously selected in $\mathcal{D}_q$. In practice, this is achieved simply by dropping the probability of novelty $p(\bar{\mathcal{D}}_q|q, a_i)$ from Equation (4.7), which produces xQuAD*:

$$f_{\text{xQuAD}^*}(q, d, \mathcal{D}_q) = (1 - \lambda)\, p(d|q) + \lambda \sum_{a_i \in \mathcal{A}} p(a_i|q)\, p(d|q, a_i), \tag{8.5}$$

where, similarly to IA-Select*, $p(a_i|q)$ and $p(d|q, a_i)$ denote the relative importance of the aspect a_i and the extent to which this aspect is covered by d, respectively. Note that, without a novelty component, the coverage-only objective functions of both IA-Select* (Equation (8.4)) and xQuAD* (Equation (8.5)) no longer require an iterative, greedy diversification strategy. In practice, as discussed in Section 3.3.2, in order to diversify the top τ documents from a ranking of n_q documents, we reduce the number of required evaluations of the objective function $f(q, d, \mathcal{D}_q)$ in Algorithm 3.1 from $\mathcal{O}(\tau n_q)$ to $\mathcal{O}(n_q)$.

In the next section, we assess the role of novelty as a diversification strategy for search result diversification. In particular, by contrasting the novelty-based diversification approaches introduced in Section 8.2.1 to the coverage-based approaches introduced in Section 8.2.2, we test novelty *in comparison* to coverage. By contrasting the coverage-based approaches introduced in Section 8.2.2 to their original hybrid formulation, we test novelty *in combination* with coverage.

8.3 Experimental Evaluation

In this section, we address the fourth claim from our thesis statement:

> *"By estimating the relevance of the retrieved documents to already well covered sub-queries, a high novelty can be attained."*

To address this claim, we investigate the role of novelty when deployed in isolation, as well as when combined with coverage in a hybrid strategy, by thoroughly evaluating all the approaches introduced in Section 8.2 under controlled settings. In particular, we aim to answer the following research questions:

Q1. Is novelty an effective diversification strategy?

Q2. How does novelty perform in comparison to coverage?

Q3. How does novelty perform in combination with coverage?

Q4. What is the role of novelty as a diversification strategy?

We address the first three research questions in Section 8.3.2. To answer Q1, we fix the diversification strategy to novelty, in order to evaluate the impact of different aspect representations. Conversely, to tackle Q2 and Q3, we measure the effectiveness of novelty in comparison to and in combination with coverage, respectively, across multiple aspect representations, which are held fixed. Finally, to provide further insights into the role of novelty as a diversification strategy, in Section 8.3.3, we address Q4, by thoroughly evaluating this strategy with simulated rankings of various quality. In the remainder of this section, we describe the experimental setup that supports the investigation of these questions.

8.3.1 Experimental Setup

In this section, we describe the setup that supports our investigation. In particular, we describe the test collections, the retrieval approaches, and the training procedure carried out for the experiments in Sections 8.3.2 and 8.3.3.

8.3.1.1 Test Collections

Our experiments are based on the WT09 and WT10 test collections. As described in Table 5.1, these test collections comprise 49 and 48 queries from the diversity task of the TREC 2009 and 2010 Web track (Clarke et al., 2009a, 2010), respectively. As a document corpus, we use the category B portion of ClueWeb09, described in Section 5.1.1. We index this corpus with Terrier (Macdonald et al., 2012a), after applying Porter's stemmer and removing stopwords.

8.3.1.2 Retrieval Approaches

To verify the consistency of our results, we experiment with several retrieval approaches and aspect representations. Firstly, as an adhoc retrieval approach, which does not perform diversification, we use the DPH model (Equation (2.31))

from the divergence from randomness framework. As discussed in Section 2.2.1.3, DPH is a parameter-free probabilistic model, and hence requires no training.

On top of DPH, we experiment with diversification approaches representative of the novelty and coverage strategies. In particular, as novelty-based approaches, we use MMR (Equation (3.8)) and MVA (Equation (3.11)), as well as their explicit variants, xMMR (Equation (8.2)) and xMVA (Equation (8.3)), introduced in Section 8.2.1. As coverage-based approaches, we consider our variants IA-Select* (Equation (8.4)) and xQuAD* (Equation (8.5)), from Section 8.2.2. Their standard versions, namely, IA-Select (Equation (3.23)) and xQuAD (Equation (4.7)), are used as hybrid approaches. To cope with the quadratic pairwise comparisons performed by novelty-based approaches (Gil-Costa, Santos, Macdonald & Ounis, 2011, 2013), both novelty, coverage, and hybrid approaches are deployed to diversify the top 100 documents retrieved by DPH. To analyse the impact of different aspect representations, in addition to a traditional implicit representation of documents in the space of the terms in the ClueWeb09 B corpus, we consider three explicit aspect representations. As described in Section 5.2.1.3, these representations include ODP categories (DZ), Bing suggestions (BS), and the official TREC Web track sub-topics (WT). The availability of relevance assessments for the WT aspects enables the evaluation of coverage and novelty using diversity estimates of various simulated quality, as we will show in Section 8.3.3.

8.3.1.3 Training and Evaluation Procedure

Most of the retrieval approaches considered in our investigation require some parameter tuning. The exceptions are DPH, IA-Select, and IA-Select*, which are parameter-free. In order to train the parameters of the other approaches (i.e., MMR and xMMR's λ, MVA and xMVA's b and σ, and xQuAD* and xQuAD's λ), we use the WT09 and WT10 queries as training and test sets, in a cross-year fashion—i.e., we train on WT09 and test on WT10, and vice versa. All parameters are optimised through simulated annealing (Kirkpatrick et al., 1983), in order to maximise ERR-IA@100 on the training queries. Accordingly, our results are reported on the union of the test queries from WT09 and WT10, using the evaluation metrics and significance symbols described in Section 5.2.1.2.

8.3.2 Experimental Results

In this section, we address the first three research questions stated in Section 8.3. In particular, Section 8.3.2.1 addresses Q1, in order to assess the effectiveness of novelty-based approaches across implicit and explicit aspect representations. Sections 8.3.2.2 and 8.3.2.3 address Q2 and Q3, by investigating how novelty performs in comparison to and in combination with coverage, respectively.

8.3.2.1 Implicit vs. Explicit Novelty

In order to answer research question Q1, we assess the effectiveness of novelty-based diversification approaches based on implicit and explicit aspect representations. In particular, we aim to investigate not only whether existing novelty-based approaches can be improved with a more refined aspect representation, but also whether any of these representations can improve over a standard, non-diversified baseline. Table 8.1 shows the diversification performances of MMR and MVA (as implicit novelty-based approaches), as well as their explicit counterparts (xMMR and xMVA, respectively) in terms of ERR-IA and α-nDCG. The latter approaches are deployed with the three explicit representations described in Section 5.2.1.3: ODP categories (DZ), Bing suggestions (BS), and the official TREC Web track sub-topics (WT). The performance of DPH is provided as a non-diversified baseline. Significance is verified using a paired t-test, as described in Section 5.1.2. In particular, for each diversification approach, a first significance symbol denotes a statistically significant difference (or lack thereof) compared to DPH. For explicit novelty-based approaches (i.e., xMMR and xMVA), a second symbol denotes significance with respect to their implicit counterpart (i.e., MMR and MVA, respectively). The best performing approach in each group is underlined, whereas the best overall approach is highlighted in bold.

From Table 8.1, we first observe that neither MMR nor MVA can consistently improve upon the non-diversified ranking produced by DPH. Indeed, as demonstrated by the number of affected queries, the positive impact observed for some queries is offset by the negative impact on other queries. These results corroborate our observations in Section 8.1, regarding the lack of empirical validation of novelty-based approaches for diversifying web search results in the literature.

Table 8.1: Diversification performance of novelty-based approaches with implicit (for MMR and MVA) and explicit (for xMMR and xMVA) aspect representations.

	$\mathcal{S}_q$	ERR-IA				α-nDCG			
		@20	−	=	+	@20	−	=	+
DPH		0.169				0.270			
+MMR		0.166°	22	47	28	0.270°	22	47	28
+xMMR	DZ	0.167$^{\circ\circ}$	17	69	11	0.269$^{\circ\circ}$	17	69	11
+xMMR	BS	0.169$^{\circ\circ}$	24	49	24	0.272$^{\circ\circ}$	22	50	25
+xMMR	WT	**0.180**$^{\circ\triangle}$	33	20	44	**0.290**$^{\triangle\triangle}$	36	19	42
+MVA		0.160$^{\triangledown}$	35	39	23	0.250$^{\blacktriangledown}$	39	37	21
+xMVA	DZ	0.169$^{\circ\triangle}$	24	55	18	0.272$^{\circ\blacktriangle}$	25	54	18
+xMVA	BS	0.150$^{\circ\circ}$	46	19	32	0.235$^{\triangledown\circ}$	45	19	33
+xMVA	WT	0.170$^{\circ\circ}$	39	14	44	0.274$^{\circ\circ}$	44	13	40

With respect to the different aspect representations, we observe that both xMMR and xMVA can significantly outperform their implicit counterparts in some settings, particularly when WT aspects are used for MMR, and DZ aspects are used for MVA. Nevertheless, xMMR and xMVA still cannot significantly improve upon DPH, which suggests that an explicit representation per se cannot guarantee an effective performance for novelty-based approaches. Recalling research question Q1, on the effectiveness of novelty as a diversification strategy, these results show that this strategy is generally ineffective when considered in isolation.

8.3.2.2 Explicit Coverage vs. Explicit Novelty

The observations in Section 8.3.2.1 suggest an inherent limitation of novelty as a diversification strategy, regardless of any particular aspect representation. To address Q2, we contrast the effectiveness of novelty and coverage-based approaches using the same representations. To this end, in Table 8.2, we compare the diversification performance of xMMR and xMVA (novelty-based) to that of IA-Select* and xQuAD* (coverage-based) across the DZ, BS, and WT explicit aspect representations. As in Table 8.1, a first significance symbol denotes significance compared to the DPH baseline. For IA-Select* and xQuAD*, two additional symbols denote significant differences from xMMR and xMVA, respectively.

Table 8.2: Diversification performance of novelty (xMMR and xMVA) and coverage-based (IA-Select* and xQuAD*) approaches for various explicit aspect representations.

	$\mathcal{S}_q$	ERR-IA				α-nDCG			
		@20	−	=	+	@20	−	=	+
DPH		0.169				0.270			
+xMMR	DZ	0.167°	17	69	11	0.269°	17	69	11
+xMVA	DZ	0.169°	24	55	18	0.272°	25	54	18
+IA-Select*	DZ	0.169$^{\circ\circ\circ}$	40	14	43	0.263$^{\circ\circ\circ}$	43	15	39
+xQuAD*	DZ	0.197$^{\triangle\triangle\triangle}$	35	13	49	0.297$^{\circ\circ\circ}$	37	14	46
+xMMR	BS	0.169°	24	49	24	0.272°	22	50	25
+xMVA	BS	0.150°	46	19	32	0.235$^{\triangledown}$	45	19	33
+IA-Select*	BS	0.201$^{\triangle\triangle\blacktriangle}$	41	14	42	0.299$^{\circ\circ\blacktriangle}$	40	14	43
+xQuAD*	BS	0.205$^{\blacktriangle\blacktriangle\blacktriangle}$	40	13	44	0.305$^{\triangle\triangle\blacktriangle}$	36	13	48
+xMMR	WT	0.180°	33	20	44	0.290$^{\triangle}$	36	19	42
+xMVA	WT	0.170°	39	14	44	0.274°	44	13	40
+IA-Select*	WT	**0.231**$^{\blacktriangle\blacktriangle\blacktriangle}$	36	10	51	**0.344**$^{\blacktriangle\blacktriangle\blacktriangle}$	31	10	56
+xQuAD*	WT	0.227$^{\blacktriangle\blacktriangle\blacktriangle}$	32	10	55	0.340$^{\blacktriangle\blacktriangle\blacktriangle}$	28	10	59

From Table 8.2, we observe that both coverage-based approaches substantially outperform the novelty-based ones in almost all settings, often significantly. The only exception is IA-Select* using the DZ aspect representation, which slightly underperforms, yet not significantly. As previously observed in Section 5.2.2.3, IA-Select (and, by extension, IA-Select*) tends to underperform when leveraging aspect representations that are uncorrelated with relevance, such as the DZ representation. Nevertheless, xQuAD* still outperforms both xMMR and xMVA in this scenario. Considering the other aspect representations, both xMMR and xMVA are significantly outperformed when using the BS (except for IA-Select* in terms of α-nDCG@20) and WT representations. In all cases, coverage-based approaches affect substantially more queries than do novelty-based approaches. Of the affected queries, in contrast to novelty-based approaches, coverage-based ones tend to improve more queries than they harm. Recalling Q2, on the effectiveness of novelty in comparison to coverage, these results show that, whenever the underlying aspect representation is held fixed, coverage provides an often significantly superior diversification strategy compared to novelty.

8.3.2.3 Explicit Coverage vs. Explicit Coverage+Novelty

The results in Section 8.3.2.2 show that novelty cannot improve against a pure coverage-based strategy. To address Q3, we investigate whether novelty can be effective in combination with coverage. To this end, Table 8.3 shows the diversification performance of IA-Select and xQuAD, which deploy hybrid diversification strategies, compared to their coverage-only versions, IA-Select* and xQuAD*, respectively. Once again, a first significance symbol denotes a statistically significant difference compared to DPH. For IA-Select and xQuAD, a second symbol denotes significance compared to IA-Select* and xQuAD*, respectively.

Table 8.3: Diversification performance of coverage (IA-Select* and xQuAD*) and hybrid (IA-Select and xQuAD) approaches for various explicit aspect representations.

	$\mathcal{S}_q$	ERR-IA				α-nDCG			
		@20	−	=	+	@20	−	=	+
DPH		0.169				0.270			
+IA-Select*	DZ	0.169°	40	14	43	0.263°	43	15	39
+IA-Select	DZ	$0.174^{\circ\circ}$	41	13	43	$0.270^{\circ\circ}$	42	14	41
+xQuAD*	DZ	$\underline{0.197}^{\triangle}$	35	13	49	$\underline{0.297}^{\circ}$	37	14	46
+xQuAD	DZ	$0.193^{\circ\circ}$	35	15	47	$0.295^{\circ\circ}$	37	15	45
+IA-Select*	BS	$0.201^{\triangle}$	41	14	42	0.299°	40	14	43
+IA-Select	BS	$\underline{0.209}^{\blacktriangle\circ}$	35	14	48	$\underline{0.311}^{\blacktriangle\circ}$	31	14	52
+xQuAD*	BS	$0.205^{\blacktriangle}$	40	13	44	$0.305^{\triangle}$	36	13	48
+xQuAD	BS	$0.204^{\blacktriangle\circ}$	40	13	44	$0.305^{\triangle\circ}$	36	13	48
+IA-Select*	WT	$\underline{\mathbf{0.231}}^{\blacktriangle}$	36	10	51	$\underline{\mathbf{0.344}}^{\blacktriangle}$	31	10	56
+IA-Select	WT	$0.228^{\blacktriangle\circ}$	32	11	54	$0.340^{\blacktriangle\circ}$	28	11	58
+xQuAD*	WT	$0.227^{\blacktriangle}$	32	10	55	$0.340^{\blacktriangle}$	28	10	59
+xQuAD	WT	$0.228^{\blacktriangle\circ}$	29	9	59	$0.341^{\blacktriangle\circ}$	26	9	62

From Table 8.3, despite generally harming fewer queries, we note that neither IA-Select nor xQuAD significantly improve upon their coverage-only versions. Recalling Q3, on the effectiveness of novelty in combination with coverage, this result shows that novelty does not significantly contribute to the effectiveness of hybrid approaches. Along with the results in Sections 8.3.2.2 and 8.3.2.3, this result raises further questions regarding the role of novelty as a diversification strategy, and the conditions (if any) under which this strategy could be effective.

8.3.3 Simulation Results

The results in Section 8.3.2 show that novelty performs ineffectively in comparison to and in combination with coverage, and even when compared to a non-diversified adhoc retrieval baseline. What remains unknown is why this is the case. Hence, in this section, we address research question Q4, by investigating the role of novelty as a diversification strategy. In particular, our ultimate goal is to identify the conditions (if any) under which novelty could be deployed effectively. To this end, we perform two complementary simulations. In Section 8.3.3.1, we analyse the impact of simulated relevance and diversity estimates on the effectiveness of novelty-based diversification approaches. In Section 8.3.3.2, we investigate how novelty is affected by the presence of non-relevant documents.

8.3.3.1 Relevance vs. Diversity

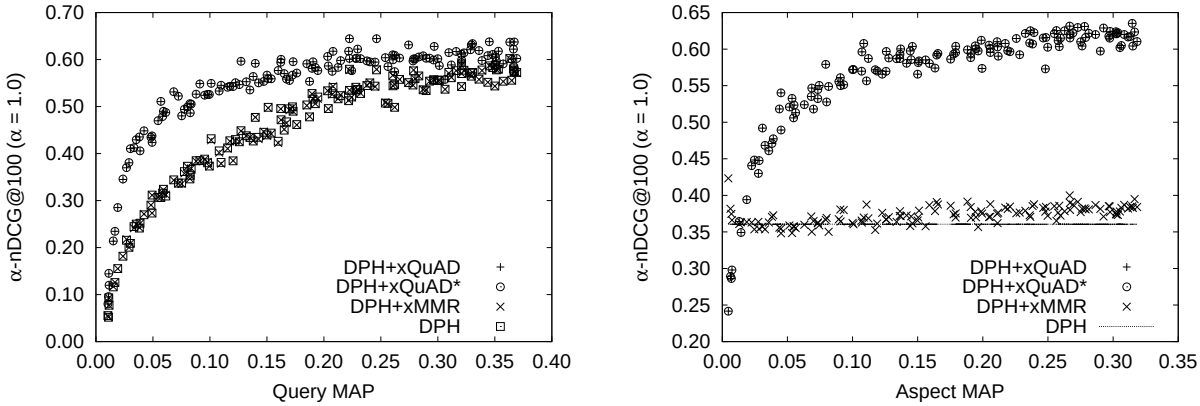
To address Q4 and ascertain the role of novelty as a diversification strategy, we analyse the effectiveness of novelty, coverage, and hybrid diversification approaches over a range of simulated relevance and diversity estimation performances. The first scenario (*simulated relevance*) simulates the application of these approaches over baseline rankings of various quality. The second scenario (*simulated diversity*) has different interpretations for different diversification approaches. For coverage-based and hybrid approaches, it represents a refined estimation of how well a document covers different aspects (i.e., $p(d|q, a_i)$ in Equations (8.5) and (4.7)). For explicit novelty-based approaches, it equates to a refined document representation in the space of the considered aspects (see Equation (8.1)), which allows for an improved identification of novel documents.

Following the procedure proposed by Turpin & Scholer (2006), we produce a range of relevance estimation performances by simulating re-rankings of the top 1000 documents retrieved by DPH for each query. In particular, each re-ranking seeks a different target *query* average precision (AP), by iteratively swapping randomly chosen pairs of relevant and irrelevant documents. For this simulation, we use the relevance assessments for the adhoc task of the TREC 2010 Web track (Clarke et al., 2010).[1] A similar procedure is used to simulate diversity

[1]The adhoc and diversity tasks share the same queries.

estimates. For this simulation, we use the TREC Web track sub-topics (WT in Table 5.5) as an aspect representation. As discussed in Section 8.3.1.2, this is the only available aspect representation with relevance assessments (i.e., those from the diversity task of the TREC 2010 Web track). Based on these "ground-truth" aspects and their corresponding relevance assessments, our simulation iteratively re-ranks the top 1000 documents retrieved by DPH for a given query with respect to each sub-topic of this query, until a target *aspect* AP performance is achieved.

As target relevance (for queries) and diversity (for query aspects) estimation performances, we split the range of possible AP values (i.e., $[0, 1]$) into 20 equally sized bins (i.e., each bin has size 0.05). Within the range of each bin, we randomly select 20 target AP values, making up a total of 400 simulated relevance and diversity estimation performances per query. To enable a comprehensive yet controlled analysis, we focus on xMMR, xQuAD*, and xQuAD as representative explicit novelty-based, coverage-based, and hybrid diversification approaches, respectively. These approaches are particularly suited for this analysis, as they tackle search result diversification as a bi-criterion optimisation problem, namely, that of balancing the trade-off between promoting relevance or diversity. As a result, they allow for a controlled experimentation, by varying relevance and diversity as two independent components. To avoid any bias towards either of these components, all approaches are applied with the standard setting of $\lambda = 0.5$.



(a) Simulated relevance (b) Simulated diversity

Figure 8.1: Diversification performance of novelty (xMMR), coverage (xQuAD*), and hybrid (xQuAD) approaches for a range of (a) relevance and (b) diversity performances.

The diversification performance of xMMR, xQuAD*, and xQuAD is shown in Figure 8.1(a) for a range of relevance estimation performances. Relevance performance (the x axis) is measured by mean average precision (MAP@100). Diversification performance (the y axis) is measured by α-nDCG@100 with $\alpha = 1.0$, so as to penalise redundancy the most heavily. Since all approaches are applied to diversify the top 100 documents, evaluation at rank cutoff 100 ensures that any observed improvements are due to removing redundancy with respect to the aspects already covered, rather than to covering additional aspects in the ranking. The performance of a standard DPH ranking is also included as a baseline.[2] From the figure, we first observe that the diversification performance of all approaches is highly correlated to their underlying relevance estimation performance. This is somewhat expected, since by improving relevance, the chance of satisfying at least one of the aspects of the query increases, as confirmed by the high correlation observed for the DPH baseline itself (Pearson's $\rho = 0.898$). As for the diversification approaches, xMMR is almost indistinguishable from DPH across the query MAP range. Likewise, xQuAD cannot be distinguished from xQuAD*. This further shows that novelty is a generally weak diversification strategy, both on its own, and when combined with coverage, corroborating the results in Section 8.3.2.

Figure 8.1(b) complements the results in Figure 8.1(a). In this second scenario, instead of varying the relevance estimations for the query, we simulate a range of diversity estimations. Once again, besides the diversification performance of xMMR, xQuAD*, and xQuAD over the range of simulated diversity estimations, we include DPH as an adhoc retrieval baseline. From Figure 8.1(b), we observe that the performance of xMMR remains indistinguishable from the performance of DPH, even with increasingly improved aspect relevance estimations, further confirming the limitation of novelty as a diversification strategy. In contrast, xQuAD* substantially improves as the underlying aspect relevance estimations improve. This shows that, besides being more robust, coverage can also benefit more from improved evidence of the association of documents to query aspects. More surprisingly, coverage proves to be a more effective strategy for promoting novelty (i.e., for reducing redundancy) than novelty itself, as

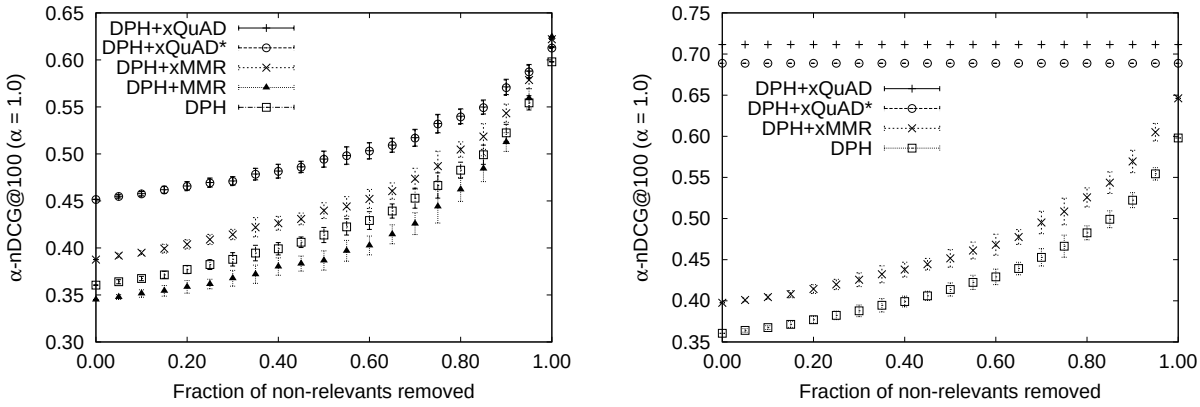
[2]Note that none of the diversification approaches attains a perfect MAP or α-nDCG, since their performance is limited by the performance of DPH.

shown by the striking superiority of xQuAD* compared to xMMR. However, the performance of xQuAD cannot be distinguished from that of xQuAD*, further confirming the limitations of novelty when combined with coverage.

8.3.3.2 Relevance vs. Non-Relevance

The results in Section 8.3.3.1 emphasise the limitations of novelty as a diversification strategy, based on a range of simulated relevance and diversity performance scenarios. Focusing on the relevance simulation scenario, for a fixed baseline ranking (i.e., a fixed relevance performance), a novelty-based diversification approach re-ranks documents on the basis of their differences from other documents, with no bearing on the likelihood of each document being relevant to a query aspect. In particular, Zhai et al. (2003) suggested that the gains in diversification performance attained by promoting novelty in the ranking may be offset by the corresponding losses due to also promoting non-relevant documents.

To fully investigate this intuition in a web search setting, we perform a complementary simulation to the one shown in Figure 8.1(a). In particular, while the previous simulation produced baseline rankings with various performances, these rankings still contained both relevant and non-relevant documents. Instead, we simulate a different scenario, where the baseline ranking is gradually improved by randomly removing non-relevant documents. This allows us to assess the impact of non-relevant documents on the performance of novelty-based diversification. In particular, Figure 8.2(a) shows the diversification performance of MMR, xMMR, xQuAD*, and xQuAD, as we increase the fraction of non-relevant documents removed from a baseline ranking produced by DPH. MMR (Equation (3.8)) is included so as to allow the analysis of the impact of non-relevant documents under an implicit novelty-based approach. The performance of DPH itself is also shown as a baseline. We test *removal fractions* from 0 to 1, in steps of 0.05. For instance, a removal fraction of 0 represents the original DPH ranking, while a fraction of 1 means that all non-relevant results have been removed from this ranking. For a given fraction, each random removal of non-relevant documents is repeated 20 times, and we report diversification performances averaged across these 20 repetitions, with error bars denoting standard deviations.



(a) Real diversity (b) Perfect diversity

Figure 8.2: Diversification performance of novelty (xMMR), coverage (xQuAD*), and hybrid (xQuAD) approaches as non-relevant documents are removed.

From Figure 8.2(a), we first note, as expected, that the performance of DPH improves as non-relevant documents are removed from its ranking. What we are interested to know, however, is whether a novelty strategy can take advantage of these gradually improving baseline performances. Looking at MMR, we observe that the performance of this implicit novelty-based approach is lower than that of DPH. Moreover, the gap between MMR and DPH remains almost unaltered as non-relevant documents are removed. A similar observation can be made for xMMR. Although it performs above DPH, the gap between the two approaches does not increase with the removal of non-relevant documents. Another important observation is that the hybrid combination of coverage and novelty implemented by xQuAD does not benefit from an improved baseline ranking when compared to xQuAD*—indeed, the performance of these two approaches is indistinguishable from one another in the figure. These results are surprising, as they show that, contrarily to the established intuition, a baseline ranking with only relevant documents is not sufficient to improve novelty-based diversification.

To investigate what could help improve novelty as a diversification strategy, we perform a similar simulation to the one presented in Figure 8.2(a), however under an extreme scenario. In particular, while the diversification approaches in Figure 8.2(a) leverage "real' aspect-document relevance estimates (i.e., those provided by DPH), we propose a scenario where these approaches are deployed

under ideal conditions, so as to stress their maximum potential. In this idealised scenario, all approaches are deployed with "perfect" aspect-document relevance estimates, based on the relevance assessments of the diversity task of the TREC 2010 Web track (Clarke et al., 2010). Moreover, all approaches are deployed to make full use of these perfect estimates. To achieve this, xMMR is deployed with $\lambda = 0$ (see Equation (8.2)), while xQuAD and xQuAD* are deployed with $\lambda = 1.0$ (see Equations (4.7) and (8.5)). Note that MMR is discarded from this simulation, as it cannot leverage aspect-document relevance estimates.

Figure 8.2(b) shows the results of this "perfect" simulation scenario. From the figure, we first observe that xMMR can consistently outperform DPH. However, as in Figure 8.2(a), the gap between xMMR and DPH remains roughly constant as non-relevant documents are removed. This surprising result shows that removing non-relevant documents from the baseline ranking does not necessarily improve novelty, even when novelty is deployed under idealised conditions.

In terms of absolute performance, although xMMR performs slightly better in contrast to its performance in the "real" scenario in Figure 8.2(a), the benefits of deploying novelty as a standalone strategy seem quite low. Indeed, while xMMR struggles to improve over DPH, xQuAD* largely outperforms both DPH and xMMR. To understand why this is the case, we can look at the right end of Figure 8.2(b). In particular, when there are only relevant documents to be diversified (i.e., when the fraction of non-relevants removed is 1), xQuAD* still outperforms xMMR. This is because, different from coverage, novelty does not take into account how well each *individual* document covers *multiple* query aspects. In contrast, coverage provides a much stronger diversification performance, by placing more emphasis on "highly diverse" documents (i.e., documents relevant to multiple aspects). Lastly, compared to xQuAD*—a purely coverage-based approach—the hybrid strategy deployed by xQuAD is ultimately shown to bring significant improvements. Recalling Q4, on the role of novelty as a diversification strategy, this result shows that, although rather limited as a standalone strategy, novelty can still play a role in combination with coverage, as a tie-breaking criterion—i.e., whenever two documents have similar coverage, the one that covers the least seen aspects (i.e., the most novel) should be ranked higher.

8.4 Summary

In this chapter, we have addressed the fourth claim of our thesis statement, by showing that an improved estimation of novelty can be attained with an improved estimation of the relevance of already selected documents with respect to each sub-query. However, we have also shown that an improved estimation of novelty does not necessarily result in an improved diversification performance.

To motivate our investigation, in Section 8.1, we questioned the lack of empirical evidence of the effectiveness of existing novelty-based approaches in a web search scenario. In order to ascertain the role of novelty for search result diversification, in Section 8.2, we proposed to bridge the gap between otherwise incomparable diversification approaches from the literature, by organising these approaches along the diversification strategy and aspect representation dimensions introduced in Section 3.3. In particular, to enable the assessment of each of these two dimensions independently of each other, we introduced four new diversification approaches, as an extension of existing novelty-based approaches, as well as a deconstruction of existing hybrid approaches.

By thoroughly evaluating the effectiveness of the introduced diversification approaches, in Section 8.3.2, we provided empirical evidence of the limitations of novelty-based diversification in a standard web search scenario. In particular, in Section 8.3.2.1, we evaluated the introduced novelty-based approaches using three distinct aspect representations. Contrary to the traditional view of novelty as an intuitive diversification strategy, we observed that none of the considered novelty-based approaches could consistently improve upon a non-diversified baseline, regardless of their leveraged aspect representation. In contrast, in Section 8.3.2.2, we showed that the introduced coverage-based approaches are substantially more effective than the novelty-based ones. In addition, in Section 8.3.2.3, we showed that the combination of coverage and novelty into a hybrid strategy does not significantly improve upon a purely coverage-based strategy.

In order to shed light on the limitations of novelty and its role as a diversification strategy, in Section 8.3.3, we performed a thorough simulation analysis. In particular, in Section 8.3.3.1, we further demonstrated the ineffectiveness of a pure novelty-based strategy when leveraging relevance estimates of various sim-

ulated performances, computed with respect to both the initial query and its sub-queries. In Section 8.3.3.2, by simulating baseline rankings with gradually fewer irrelevant documents, we observed a marginal improvement when promoting novelty. Finally, by analysing an extreme scenario considering only relevant documents, we showed that novelty plays a role at breaking the tie between documents with a similar coverage of the multiple aspects of the query, providing a further empirical justification for hybrid diversification strategies.

The experiments in this chapter showed that novelty can be an effective strategy for search result diversification, when deployed in combination with coverage in a hybrid strategy. Nevertheless, the inconsistent performance of novelty suggests that automatically detecting when such a criterion could be effectively exploited is key for the success of hybrid approaches. More generally, automatically determining how much to diversify the search results is of utmost importance for a robust integration of relevance, coverage, and novelty. Investigating such a robust diversification mechanism is the goal of the next chapter.

Chapter 9

Diversification Trade-Off

In Chapters 5 through 8, we validated the effectiveness of xQuAD in contrast to the current state-of-the-art and described effective instantiations for each of the components of the framework. In particular, we proposed effective mechanisms to generate sub-queries in Chapter 6, as well as to estimate the coverage and novelty of the retrieved documents with respect to each sub-query in Chapters 7 and 8. Throughout these chapters, we assumed that all queries were equally amenable to diversification. However, depending on how ambiguous they are, different queries may arguably benefit from more or less aggressive diversification strategies.

A more lenient or more aggressive diversification can be attained by appropriately setting the diversification trade-off λ, which balances relevance and diversity in the ranking, as described in Equation (4.1) of Chapter 4. In this chapter, we propose to selectively diversify the documents retrieved for different queries, by inferring an effective diversification trade-off for each individual query. As a result, not only do we predict when to diversify, but also by how much. To this end, we leverage a large range of query features from the literature and cast this problem as a regression task, namely, the task of predicting an effective trade-off.

In the remainder of this chapter, Section 9.1 overviews selective ranking approaches in the literature. Section 9.2 details our approach for predicting an effective diversification trade-off on a per-query basis. Our approach is thoroughly evaluated in Section 9.3. The results attest the effectiveness of our selective mechanism, with significant gains compared to a mechanism that optimises the trade-off uniformly for all queries, regardless of their predicted ambiguity.

9.1 Selective Web Search

Selective ranking approaches are relatively common in web search. A typical example is the adaptation of the produced ranking to account for the predicted intent of each query, as discussed in Section 7.1. Another classical example of selective ranking is the identification of queries more likely to benefit from query expansion. For instance, Yom-Tov et al. (2005) showed that the effectiveness of an expanded query is highly dependent on the effectiveness of the original query, since a poor first-pass retrieval may lead to the selection of irrelevant expansion terms. To overcome this problem, they proposed a selective mechanism to decide when to apply query expansion, based upon the predicted difficulty of the original query. Relatedly, Macdonald et al. (2005) hypothesised that using a high quality external resource, such as Wikipedia, could improve query expansion in cases where the local corpus would lead to decreased effectiveness. Accordingly, they proposed to leverage query performance predictors (Carmel & Yom-Tov, 2010) as features for choosing which corpus to use for expanding each query.

Plachouras (2006) introduced a Bayesian decision mechanism to select the retrieval approach most likely to be effective for a given query. Such a mechanism performed a density analysis, considering the observed effectiveness of multiple candidate approaches under multiple experimental conditions on a training set, and the likelihood of each experimental condition given a test query. Example experimental conditions included statistics of the available data, such as the distribution of terms, domains, and hyperlinks among the retrieved documents (Plachouras & Ounis, 2004). In the same vein, Peng & Ounis (2009) proposed a selective mechanism to choose the single most effective feature from a set of candidate query-independent features, such as those described in Section 2.2.2, to be integrated to a baseline ranking. In particular, the divergence between the score distributions of the documents retrieved for a query prior to and after the integration of a feature served as a mechanism to predict the effectiveness of this feature relatively to other candidates, given how well each candidate performed for training queries with a similar divergence. This approach was later extended to select any arbitrary ranking function from a pool of candidate functions to be applied for each individual query (Peng et al., 2010; Peng, 2010).

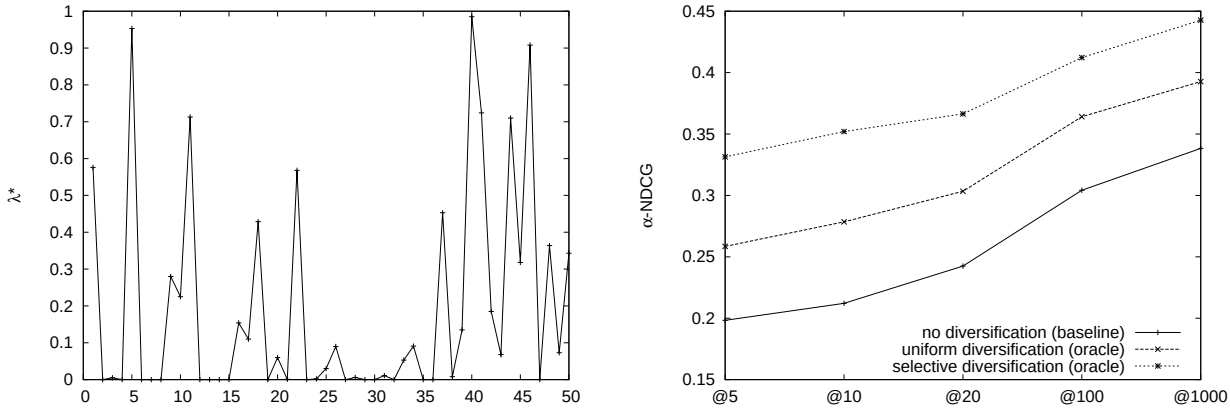
In a similar vein, Geng et al. (2008) proposed a selective approach to choose the most appropriate training examples to use for learning a ranking function on a per-query basis. Given a query, their approach deployed a ranking function (Joachims, 2002) learned on a selected subset of the available training queries, as opposed to the entire training set. Such a subset was identified either online, as the nearest neighbouring queries to the test query, or offline, by clustering the available training queries (Aha et al., 1991). For the nearest neighbour classification, the average score of each document feature across the top retrieved documents for a given query was used as a feature for this query.

Inspired by these approaches, we seek to learn an effective diversification trade-off for an unseen query based upon the optimal trade-offs observed for similar training queries. However, differently from these approaches, which relied on a single feature to identify neighbouring queries, we leverage a large pool of query features, inspired by different query understanding approaches in the literature. To the best of our knowledge, our approach constitutes the first attempt to tackle search result diversification as a query-dependent ranking problem.

9.2 Selective Diversification

Intuitively, maximising the satisfaction of the population of users issuing the same, ambiguous query involves trading off relevance for diversity in the ranking. On the one hand, a relevance-oriented ranking can focus on the most likely information need underlying the query (e.g., the most popular interpretation or aspect of the query). On the other hand, a diversity-oriented ranking can also cater for other plausible needs. As discussed in Section 4.2, these two strategies can be integrated as a bi-criteria ranking objective for improved effectiveness.

Several diversification approaches in the literature build upon this idea, with a parameter λ controlling the trade-off between relevance and diversity—e.g., MMR (Equation (3.8)), RM (Equation (3.9)), MVA (Equation (3.11)), ARW (Equation (3.13)), SSSD (Equation (3.14)), and our xQuAD framework (Equation (4.1)). Typically, this trade-off is uniformly optimised so as to maximise the diversification performance on a set of training queries. However, different queries may benefit from different diversification strategies, since not all queries



(a) Optimal trade-off per query. (b) Optimal diversification performance.

Figure 9.1: Optimal trade-off and diversification performance for the WT09 queries.

are equally ambiguous. For instance, while a query like "*bond*" could arguably benefit from a more aggressive diversification, a more lenient strategy could suffice for a less ambiguous query such as "*james bond*". In the extreme, a clear query like "*james bond skyfall website*" could attain an effective performance even without any diversification. To quantify this observation, Figure 9.1(a) shows the optimal trade-off λ^* for xQuAD for each of the 50 TREC 2009 Web track queries (Clarke et al., 2009a). From the figure, it is clear that different queries benefit from different trade-offs, and that any uniform choice of λ for all queries would be suboptimal. Indeed, Figure 9.1(b) shows that optimising this trade-off on a per-query basis substantially outperforms a uniform optimisation regime.

In this chapter, we propose to selectively diversify the documents retrieved for a given query, by predicting an effective trade-off between relevance and diversity for this query. To this end, we introduce a supervised selective diversification approach, aimed at inferring an effective trade-off for an unseen query, based on the optimal trade-off observed for similar training queries. Our approach is general and can be applied to improve any diversification approach, provided that it adheres to the aforementioned view of search result diversification as the problem of optimising the trade-off between promoting relevance or diversity in the ranking. In the following, Section 9.2.1 formalises our learning approach, while Section 9.2.2 describes the query features used to instantiate it.

9.2.1 Learning a Regression Model

Given an unseen query q, our goal is to learn an effective setting for the diversification trade-off λ, which maximises diversification performance according to a target evaluation metric. Following the standard discriminative learning framework described in Section 2.2.3.1, we aim to learn a hypothesis $h : \mathcal{X} \rightarrow \mathcal{Y}$, mapping the input space $\mathcal{X}$ into the output space $\mathcal{Y}$.

Our input space $\mathcal{X}$ encompasses a set $\mathbf{x} = \{\mathbf{x}_j\}_{j=1}^{m}$ of m learning instances, with each instance $\mathbf{x}_j = \Phi(q_j)$ conveying a vector representation of a query q_j, according to the feature extractor Φ. The actual features used in our investigation are described in Section 9.2.2. In turn, our output space $\mathcal{Y}$ comprises a set $\mathbf{y} = \{y_j\}_{j=1}^{m}$ of m learning labels, defined in the domain of the real numbers. In particular, a label y_j for a training query q_j corresponds to the optimal trade-off between relevance and diversity obtained for this query, according to a target diversity evaluation metric, such as any of the metrics described in Section 3.4.2. In principle, to obtain such an optimal trade-off, we could use any optimisation method. For simplicity, we perform a full scan over the range of possible λ values for the query q_j (i.e., $0 \leq \lambda \leq 1$), with steps of 0.001, and select the best value (according to the target evaluation metric) as the label y_j. Note that this process is entirely conducted offline, with no knowledge of unseen queries.

Lastly, to learn a hypothesis h, we could use different numeric prediction approaches, such as linear regression or model trees (Witten & Frank, 2005). In our investigation, we employ a k-nearest neighbour (k-NN) (Aha et al., 1991) algorithm. As an instance-based learning approach, k-NN does not have an explicit training phase. Instead, it stores the training data in memory and performs an online regression for each unseen query. During the online query processing, we predict an effective trade-off λ for a test query q as the mean of the λ_j^* values of the k nearest neighbouring training queries to q:

$$\lambda = h(\Phi(q)) = h(\mathbf{x}) = \frac{1}{k} \sum_{j|\mathbf{x}_j \in \Gamma_{\mathbf{x}}^k} \lambda_j^*, \tag{9.1}$$

where $\Gamma_{\mathbf{x}}^k$ comprises the k nearest training queries to q in the space of the considered features, according to a distance function, typically the Euclidian distance.

The main advantage of k-NN as a lazy learning approach is that a different and potentially more targeted hypothesis is learned based on the training neighbourhood of an unseen query, rather than on the entire training data. This reduces the complexity of the learning process by exploiting the locality of the data (Geng et al., 2008). Additionally, k-NN does not make strong assumptions about the underlying data distribution, as other regression approaches do (Aha et al., 1991). Despite its simplicity and effectiveness, two main concerns arise when employing an instance-based learning approach such as k-NN. Firstly, the cost of prediction can be significant, particularly when a large number of training instances is available. Fortunately, searching for the nearest neighbours of an unseen query can be done efficiently with the use of appropriate indexing structures to store training queries, such as ball trees (Omohundro, 1989). The second concern related to instance-based learning is the dimensionality of the feature space. In particular, k-NN considers all instance features when searching for the nearest neighbours. When the similarity between an unseen query and a true neighbour is determined by only a few features, these queries may be considered far from each other in light of the entire feature space, potentially compromising the accuracy of the prediction (Witten & Frank, 2005). To tackle this issue, we perform a feature selection ahead of the prediction step, as described in Section 9.3.1.

9.2.2 Query Features

A pool of meaningful features is crucial for the effectiveness of any learning process. As the goal of our particular task is to learn an effective trade-off between promoting relevance or diversity for a given query, a natural first direction is to look for features that capture the ambiguity of this query. Nevertheless, an effective setting for the diversification trade-off may depend not only on the ambiguity of the query itself, but also on how a particular diversification approach tackles such ambiguity, through its estimations of relevance and diversity.

In order to provide a rich query representation for our learning task, we devise a total of 952 features, as variants of 33 distinct feature classes, summarised in Table 9.1. Multiple variants are produced analogously to those described in Section 7.2.2.3. In particular, as described in the "variants" column of Table 9.1, doc-

Table 9.1: Query features used for trade-off prediction.

	Feature	Description	Variants	Total
QCI	AcronymSenses	Number of acronym senses		1
	DisambCount	Number of disambiguation pages	2m x 10c	20
	DisambSenses	Number of disambiguation senses	3m x 10c x 3s	90
	EntityCount	Number of entities in the query	4t	4
QID	DomainDistro	Number of documents per domain	5m x 3s	15
	HostDistro	Number of documents per host	5m x 3s	15
	URLComponents	Number of URL components	5m x 4t	20
	HomePage	Whether there is a homepage	2m	2
	MaxIncrement	Maximum score difference	2m	2
QLM	QueryFrequency	Number of occurrences		1
	ClickEntropy	URL-level click entropy		1
	HostEntropy	Host-level click entropy		1
	ResultCount	Examined documents per session	3s	3
	ClickCount	Clicked documents per session	3s	3
	ReformCount	Reformulations per session	3s	3
	SessionDuration	Session duration (in sec.)	3s	3
QPP	AvICTF	Pre-retrieval predictor		1
	AvIDF	Pre-retrieval predictor		1
	AvPMI	Pre-retrieval predictor		1
	EnIDF	Pre-retrieval predictor		1
	Gamma1	Pre-retrieval predictor		1
	Gamma2	Pre-retrieval predictor		1
	Terms	Pre-retrieval predictor		1
	Tokens	Pre-retrieval predictor		1
	ClarityScore	Post-retrieval predictor	5m x 10c	50
	QueryDifficulty	Post-retrieval predictor	2m x 10c	20
	QueryFeedback	Post-retrieval predictor	2m x 10c	20
QTC	CategoryCount	Number of categories	2m x 10c	20
	CategoryEntropy	Category entropy	2m x 10c	20
	CategoryCosine	Category pairwise cosine	2m x 10c x 3s	60
	ConceptCount	Number of concepts	5m x 3t x 10c	150
	ConceptEntropy	Concept entropy	5m x 3t x 10c	150
	ConceptCosine	Concept pairwise cosine	5m x 3t x 6c x 3s	270
	Grand total			952

ument features are computed based on five different ranking mechanisms (denoted "m" in the "variants" column of Table 9.1), namely, BM25 (Equation (2.13)), DPH (Equation (2.31)), and the APIs of the Bing, Google, and Yahoo! web search engines. For the latter three mechanisms, URLs not present in our target test collection are discarded. In addition, each of these features is computed at ten distinct rank cutoffs (denoted "c"): 1, 2, 3, 5, 10, 20, 50, 100, 500, and 1000. For entity-oriented features, up to four types (denoted "t") are considered: persons, organisations, products, and locations. Lastly, distributional features, such as the number of documents per domain or the pairwise distance between any two retrieved documents, are summarised using up to three summary statistics (denoted "s"): mean, standard deviation, and maximum. The devised features are organised into five groups, according to the tasks that motivated each feature: query concept identification (QCI), query intent detection (QID), query log mining (QLM), query performance prediction (QPP), and query topic classification (QTC). In the following, we describe each of these groups.

Query Concept Identification (QCI) A first sign of ambiguity is present at the word level (Sanderson, 2008). For instance, a query might contain multiple named entities, possibly representing a complex information need with multiple intents, as discussed in Section 7.2.2.3. Alternatively, a single query term can have multiple meanings according to a particular source, such as a dictionary or an encyclopedia. To capture these intuitions, we quantify the occurrence of named entities in the query, as well as of Wikipedia disambiguation pages in the ranking produced for this query. In addition, we further quantify the ambiguity of a query by detecting the presence of acronyms. To this end, instead of deploying sophisticated natural language processing techniques, we simply compute the number of interpretations returned by `all-acronyms.com` for single-term queries.[1]

Query Intent Detection (QID) Navigational queries are usually less ambiguous than informational ones (Welch et al., 2011), which suggests that useful query intent detection features might also be useful for predicting query ambiguity (Kang & Kim, 2003). With this in mind, we leverage several query intent

[1] We assume that acronyms in multi-term queries are disambiguated by the additional terms.

detection features proposed in the literature for our learning task. These include the distribution of host names, domain names, and other URL fragments among the top retrieved documents for a query, as well as the presence of a homepage among these documents. Additionally, we consider the maximum difference in relevance scores between any two retrieved documents as a strong indicator of the query intent. In particular, by analysing the score distribution for a query, this feature captures the intuition that navigational queries often have only a few relevant documents that have markedly larger scores.

Query Log Mining (QLM) Another promising direction for inferring the ambiguity of a query is to observe the past usage of this query in a query log (Silvestri, 2010). Inspired by previous research on query log mining for ambiguity detection, we deploy several query log features. For instance, queries often clicked for a single document are intuitively less ambiguous than queries with clicks spread over multiple distinct documents. We capture this intuition by computing the entropy of user clicks (Clough et al., 2009; Wang & Agichtein, 2010), at both the document URL and host levels. Additionally, for each query session, we compute the total number of documents displayed to the user, the total duration of the session in seconds, and the total number of query reformulations performed during the session (Clough et al., 2009). Finally, we also consider basic features such as the frequency of the query in the log and the total number of clicks it received.

Query Performance Prediction (QPP) As previously discussed in Section 9.2, our selective mechanism is agnostic to any particular diversification approach, and makes no assumption regarding how a given approach estimates the relevance and the diversity of a document with respect to a query. Since an optimal diversification trade-off clearly depends on the performance of these estimates, a promising direction is to leverage query performance prediction features within our learning approach (Carmel & Yom-Tov, 2010). In particular, we employ a range of both pre-retrieval and post-retrieval predictors. Pre-retrieval predictors estimate the performance of a query based on statistics derived from the target collection, such as the document frequency of individual query terms or the pointwise mutual information of pairs of query terms. Post-retrieval pre-

dictors, in turn, are based on the top retrieved documents for the query. For instance, they can estimate the query performance based on how cohesive these documents are, according to their language models (Cronen-Townsend et al., 2002) or relevance models built from them (Zhou & Croft, 2007).

Query Topic Classification (QTC) To further refine the prediction of an effective diversification trade-off for an unseen query, we consider more specialised features, which capture the distribution of topics among the documents retrieved for this query. These include the raw number of topics represented in the top retrieved documents for the query, the pairwise "topic" distance between any two retrieved documents for the query, and the "topic" entropy of the centroid of all retrieved documents (Song et al., 2009). Similarly to our query topic classification features discussed in Section 7.2.2.3, we consider topics related to multiple named entities, as well as to multiple categories, both derived from Wikipedia. In particular, our intuition is that documents sharing the same entities or the same categories tend to be more similar to one another, in which case the query for which they are retrieved tends to be less ambiguous.

9.3 Experimental Evaluation

In this section, we address the fifth claim from our thesis statement:

> *"By inferring the level of ambiguity of different queries, a balance between promoting relevance or diversity can be effectively attained."*

In order to address this claim, we investigate the effectiveness of our selective approach, which assigns an appropriate diversification trade-off on a per-query basis, in contrast to a uniform approach, which assigns the same trade-off for every query. This investigation aims to answer the following research questions:

Q1. How effective is our selective diversification approach?

Q2. What features constitute effective predictors of an optimal trade-off?

Q3. How robust is our approach to perturbations in the prediction accuracy?

In the following, Section 9.3.1 details the experimental setup that supports the investigation of these questions in Section 9.3.2.

9.3.1 Experimental Setup

In this section, we detail the test collection, topics, and evaluation metrics used in our experiments. Additionally, we describe the baseline diversification approaches and the different learning regimes considered in the evaluation of our selective diversification approach, introduced in Section 9.2.

9.3.1.1 Test Collection

Our experiments use the WT09 test collection, comprising 49 queries from the TREC 2009 Web track (Clarke et al., 2009a), as described in Table 5.1. The category B portion of ClueWeb09 is used as our document corpus. In particular, we index this corpus using Terrier (Macdonald et al., 2012a), after applying Porter's stemmer and removing standard English stopwords.

9.3.1.2 Diversification Approaches

In order to test the generality of our proposed approach, we deploy it to selectively predict the trade-off λ for two diversification approaches, namely, our xQuAD framework (Equation (4.1)) and MMR (Equation (3.8)). For MMR, following Carbonell & Goldstein (1998), we use the cosine distance as a similarity metric. For xQuAD, as a means to isolate the impact of any particular choice for representing query aspects, we use the official TREC Web track sub-topics (WT in Table 5.5) as sub-queries. Both MMR and xQuAD are deployed to diversify the top 100 documents retrieved by two ad-hoc retrieval baselines: BM25 (Equation (2.31)) and DPH (Equation (2.31)).

9.3.1.3 Training Regimes

In our evaluation, five distinct regimes are considered in order to set the parameter λ to control the diversification trade-off for both MMR and xQuAD:

1. Uni(base): a baseline uniform diversification regime, with a single λ value learned for all queries in each fold through a 5-fold cross validation.

2. Uni(ora): an upper-bound uniform diversification regime, with a single λ value selected to maximise the average performance across all queries.

3. Sel(rand): a baseline selective diversification regime, with a different λ value randomly sampled from the interval [0..1] on a per-query basis;

4. Sel(ora): an upper-bound selective diversification regime, with a different λ value selected on a per-query basis, so as to maximise the performance of each query individually.

5. Sel(k-nn): our proposed selective diversification regime, with a different λ value learned for each query through a 5-fold cross validation.

To set the k parameter for k-NN, a leave-one-out cross-validation is performed, by minimising mean absolute error (MAE; Witten & Frank, 2005). Additionally, given the large number of features described in Section 9.2.2, we investigate the impact of different feature selection mechanisms for the Sel(k-nn) regime. In particular, besides a baseline variant with no feature selection applied (Sel(k-nn,nofs)), we deploy two standard feature selection techniques:

- Sel(k-nn,pca) performs a principal component analysis (PCA) in order to reduce the dimensionality of the feature space (Pearson, 1901).

- Sel(k-nn,bfs) performs a greedy best-first search (BFS) in the space of feature combinations (Kohavi & John, 1997). To avoid converging on a local maximum, we allow negative improvements in the search for the next feature to be added to the current best combination. Hence, our stopping criterion becomes the maximum number of features to be selected: 100.

9.3.2 Experimental Results

In the remainder of this section, we thoroughly evaluate our proposed approach for selectively diversifying the documents retrieved for queries with different levels of ambiguity. In particular, in Section 9.3.2.1, we assess the effectiveness of our

approach at improving the diversification performance of MMR and xQuAD. In Section 9.3.2.2, we analyse the suitability of different groups of features for predicting an effective diversification trade-off on a per-query basis. Finally, in Section 9.3.2.3, we further investigate the robustness of our approach based on the impact of random perturbations on the prediction of this trade-off.

9.3.2.1 Diversification Effectiveness

In this experiment, we address research question Q1, regarding the effectiveness of our selective diversification approach. To this end, Table 9.2 shows the diversification performance of MMR and xQuAD, deployed on top of BM25 and DPH, under the several training regimes described in Section 9.3.1.3. These include Uni(base) as a baseline uniform regime, and Sel(rand) as a sanity check for the several variants of our selective regime, i.e., Sel(k-nn,•). Additionally, Uni(ora) and Sel(ora) provide upper-bound performances for both a uniform and a selective diversification regime, respectively. Diversification performance is given by ERR-IA (Equation (3.28)) and α-nDCG (Equation (3.29)). Significance is verified by a paired t-test, with the symbols previously introduced in Section 5.1.2 denoting significant differences (or lack thereof). For all instantiations of MMR and xQuAD, a first symbol denotes significant differences compared to BM25 or DPH. A second such symbol denotes significance with respect to Uni(base), while a third symbol denotes significance with respect to Uni(ora).

From Table 9.2, we first observe that, compared to the adhoc retrieval baselines, i.e., BM25 and DPH, MMR cannot improve significantly. On the other hand, xQuAD significantly improves upon both baselines in most settings, corroborating our findings in Chapter 8 regarding the superiority of a hybrid diversification strategy in contrast to a pure novelty-based strategy.

Contrasting the training regimes deployed by MMR and xQuAD, we note that Sel(k-nn,•) improves over Uni(base) in all cases for BM25+MMR, and in most cases for DPH+MMR, often significantly. For xQuAD, significant improvements over Uni(base) are observed for the Sel(k-nn,bfs) variant on top of both BM25 and DPH. Recalling research question Q1, on the effectiveness of our selective approach, these observations show that predicting an effective diversification

Table 9.2: Diversification performance under different training regimes.

	$\mathcal{S}_q$	λ	ERR-IA				α-nDCG			
			@20	−	=	+	@20	−	=	+
BM25			0.130				0.229			
+MMR		Uni(base)	$0.079^{\blacktriangledown}$	20	19	10	$0.149^{\blacktriangledown}$	20	19	10
+MMR		Uni(ora)	$\underline{0.131}^{\circ\blacktriangle}$	8	33	8	$\underline{0.228}^{\circ\blacktriangle}$	8	33	8
+MMR		Sel(rand)	$0.052^{\blacktriangledown\circ\blacktriangledown}$	31	7	11	$0.110^{\blacktriangledown\circ\blacktriangledown}$	33	7	9
+MMR		Sel(k-nn,nofs)	$0.113^{\triangledown\circ\triangledown}$	18	24	7	$0.205^{\triangledown\triangle\triangledown}$	18	24	7
+MMR		Sel(k-nn,pca)	$0.115^{\triangledown\triangle\triangledown}$	11	28	10	$0.207^{\circ\blacktriangle\circ}$	11	28	10
+MMR		Sel(k-nn,bfs)	$0.133^{\circ\blacktriangle\circ}$	9	32	8	$0.237^{\circ\blacktriangle\circ}$	10	32	7
+MMR		Sel(ora)	$\underline{0.140}^{\triangle\blacktriangle\triangle}$	5	25	19	$\underline{0.248}^{\triangle\blacktriangle\triangle}$	8	25	16
+xQuAD	WT	Uni(base)	$\underline{0.177}^{\circ}$	13	9	27	0.280°	12	9	28
+xQuAD	WT	Uni(ora)	$0.176^{\triangle\circ}$	10	11	28	$\underline{0.284}^{\triangle\circ}$	10	11	28
+xQuAD	WT	Sel(rand)	$0.173^{\circ\circ\circ}$	15	8	26	$0.271^{\circ\circ\circ}$	14	8	27
+xQuAD	WT	Sel(k-nn,nofs)	$0.167^{\circ\triangledown\circ}$	15	9	25	$0.267^{\circ\triangledown\circ}$	16	9	24
+xQuAD	WT	Sel(k-nn,pca)	$0.174^{\circ\circ\circ}$	15	10	24	$0.274^{\circ\circ\circ}$	15	10	24
+xQuAD	WT	Sel(k-nn,bfs)	$0.202^{\blacktriangle\triangle\circ}$	10	9	30	$0.305^{\blacktriangle\triangle\circ}$	9	9	31
+xQuAD	WT	Sel(ora)	$\underline{\mathbf{0.237}}^{\blacktriangle\blacktriangle\blacktriangle}$	1	13	35	$\underline{\mathbf{0.349}}^{\blacktriangle\blacktriangle\blacktriangle}$	1	13	35
DPH			0.143				0.243			
+MMR		Uni(base)	0.134°	8	33	8	0.231°	9	32	8
+MMR		Uni(ora)	$\underline{0.143}^{\circ\circ}$	0	49	0	$\underline{0.243}^{\circ\circ}$	0	49	0
+MMR		Sel(rand)	$0.047^{\blacktriangledown\blacktriangledown\blacktriangledown}$	37	8	4	$0.105^{\blacktriangledown\blacktriangledown\blacktriangledown}$	37	8	4
+MMR		Sel(k-nn,nofs)	$0.136^{\circ\circ\circ}$	6	37	6	$0.229^{\circ\circ\circ}$	6	37	6
+MMR		Sel(k-nn,pca)	$0.137^{\circ\circ\circ}$	8	36	5	$0.231^{\circ\circ\circ}$	9	35	5
+MMR		Sel(k-nn,bfs)	$0.144^{\circ\circ\circ}$	3	41	5	$0.244^{\circ\circ\circ}$	3	41	5
+MMR		Sel(ora)	$\underline{0.147}^{\triangle\triangle\triangle}$	3	35	11	$\underline{0.251}^{\triangle\triangle\triangle}$	4	34	11
+xQuAD	WT	Uni(base)	$0.201^{\triangle}$	18	10	21	$\underline{0.303}^{\triangle}$	18	10	21
+xQuAD	WT	Uni(ora)	$\underline{0.202}^{\triangle\circ}$	17	10	22	$\underline{0.303}^{\triangle\circ}$	17	10	22
+xQuAD	WT	Sel(rand)	$0.190^{\circ\triangledown\triangledown}$	20	8	21	$0.289^{\circ\triangledown\triangledown}$	19	8	22
+xQuAD	WT	Sel(k-nn,nofs)	$0.193^{\triangle\circ\circ}$	14	14	21	$0.294^{\triangle\circ\circ}$	15	14	20
+xQuAD	WT	Sel(k-nn,pca)	$0.197^{\triangle\circ\circ}$	16	11	22	$0.305^{\triangle\circ\circ}$	17	11	21
+xQuAD	WT	Sel(k-nn,bfs)	$0.204^{\blacktriangle\triangle\circ}$	13	13	23	$0.306^{\triangle\triangle\circ}$	15	13	21
+xQuAD	WT	Sel(ora)	$\underline{\mathbf{0.252}}^{\blacktriangle\blacktriangle\blacktriangle}$	1	18	30	$\underline{\mathbf{0.366}}^{\blacktriangle\blacktriangle\blacktriangle}$	1	18	30

trade-off on a per-query basis outperforms a uniform setting of this trade-off for all queries. Comparing the different variants of our approach, we note that feature

selection plays an important role in the identification of an effective diversification trade-off, particularly when such a large set of features as the one described in Section 9.2.2 is employed. In particular, SEL(k-NN,PCA), the variant based on principal component analysis, brings improvements over no feature selection (i.e., SEL(k-NN,NOFS)) in all cases. Further improvements are observed when a greedy best-first search feature selection approach is used (i.e., SEL(k-NN,BFS)). Moreover, this variant consistently outperforms the upper-bound uniform diversification baseline, given by the UNI(ORA) regime. Although not significant, these improvements are remarkable, given that the UNI(ORA) is deployed with an ideal setting of the diversification trade-off, optimised on all test queries.

Lastly, the non-triviality of these results is further attested by the superior performance of our selective approach compared to the SEL(RAND) regime, which randomly assigns a diversification trade-off on a per-query basis. Interestingly, while the performance of MMR is highly sensitive to a random assignment of the diversification trade-off, xQuAD still performs relatively well in this scenario. Nonetheless, such a resilient behaviour of xQuAD does not mean it would benefit less from our selective diversification regime. Indeed, the upper-bound performance of SEL(ORA) gives an encouraging room for further improvements.

9.3.2.2 Feature Analysis

The results in Section 9.3.2.1 attest the effectiveness of our proposed selective diversification approach, with significant improvements over a uniform diversification across multiple settings. These results are particularly promising given the simple techniques we deployed to select a subset of effective features from the large pool used in this work, as described in Section 9.2.2.

Although automatically finding an optimal subset of these features is beyond the scope of this thesis, in this section, we investigate the predictive power of different groups of features. In particular, we aim to answer research question Q2, concerning the usefulness of different features for our specific learning task. Inspired by our proposed classification of the features described in Section 9.2.2, we analyse the performance of our selective diversification approach using features from five different groups: query concept identification (QCI), query performance

prediction (QPP), query topic classification (QTC), query intent detection (QID), and query log mining (QLM) features. In particular, Table 9.3 shows the performance of our selective diversification approach for both MMR and xQuAD, with features grouped according to the aforementioned classification.

Table 9.3: Per-feature group performance in terms of α-nDCG@10.

	MMR		xQuAD	
	BM25	DPH	BM25	DPH
Sel(k-nn,nofs)	0.161	0.197	0.235	0.267
Sel(k-nn,qci)	0.166	0.159	0.239	0.261
Sel(k-nn,qlm)	0.174	0.182	0.240	0.246
Sel(k-nn,qpp)	0.168	0.170	0.245	0.257
Sel(k-nn,qtc)	0.172	0.203	0.238	0.284
Sel(k-nn,qid)	0.164	0.182	0.251	0.249
Pearson's ρ		0.53		-0.52

From Table 9.3, we first observe that, for each of the possible combinations of adhoc (i.e., BM25 or DPH) and diversity (MMR or xQuAD) baselines, there is at least one feature group that can improve the effectiveness of our selective regime, compared to using all the available features (i.e., the Sel(k-nn,nofs) variant). Recalling research question Q2, on the relative effectiveness of different features, we observe that our query topic classification (QTC) features constitute the most robust group of all features considered in this work, with improvements across all different baselines.[2] Another interesting observation relates to how different diversification approaches leverage different feature groups. For instance, while MMR shows a positive correlation ($\rho = 0.53$) between its performances on top of BM25 and DPH across different groups, xQuAD favours different groups of features depending on the underlying ranking approach deployed to produce its relevance and coverage estimates ($\rho =$ -0.52). Although anecdotal, these observations illustrate the challenge of selecting a suitable subset of features for learning an effective diversification trade-off for different diversification approaches.

[2]According to our greedy best-first search feature selection approach, the most effective features in this group are CategoryCount and CategoryCosine, i.e., the total number of top-level categories among Wikipedia articles retrieved for a query, and the average distance between these articles in the space of categories, respectively.

9.3.2.3 Prediction Robustness

In Section 9.3.2.2, we have shown that our approach can be effective even when deploying relatively simple feature selection techniques to reduce the dimensionality of our feature space. In this section, we investigate the reasons for such a robust behaviour. More precisely, we aim to answer research question Q3, regarding the sensitivity of our approach to perturbations in the underlying regression accuracy. To this end, we propose a simple perturbation criterion, which introduces randomness in the regression process. In particular, we predict a diversification trade-off λ for a query q according to a linear combination:

$$\lambda = (1 - \phi)\lambda^* + \phi\lambda^{\text{rnd}}, \tag{9.2}$$

where λ^* is the optimal trade-off for the query q, obtained as described in Section 9.2.1, and λ^{rnd} is a random number in the interval [0,1]. The interpolation parameter ϕ represents the perturbation level. When $\phi = 0$, we have a perfect prediction accuracy, equivalent to our upper-bound regime Sel(ora). On the other extreme, when $\phi = 1$, we have a completely random prediction accuracy, equivalent to our baseline regime Sel(rand). Figure 9.2 shows the diversification performance of Sel(ora) for different levels of prediction perturbation, using DPH+xQuAD. As a baseline, we also include the performance of the Uni(ora) regime, which represents the upper-bound for a uniform diversification.

Regarding research question Q3, the results in Figure 9.2 attest the robustness of our selective approach to perturbations in regression accuracy. In particular, our approach can outperform the upper-bound uniform diversification even with up to 50% of accuracy perturbation (i.e., $\phi = 0.5$). This is remarkable, and confirms the effectiveness of our approach, despite the inherent difficulty of the prediction task. A second observation relates to how close to the upper-bound performance we can expect to be in a realistic scenario. From Figure 9.2, we observe that gradual improvements are attained as the level of perturbation drops. However, after a certain level ($\phi \approx 0.05$), further improvements seem unlikely, as they would require a near-perfect regression accuracy. In this example, we could expect the upper-bound performance of DPH+xQuAD in terms of α-nDCG@10 to lie in between 0.31 and 0.32 in a more realistic scenario.

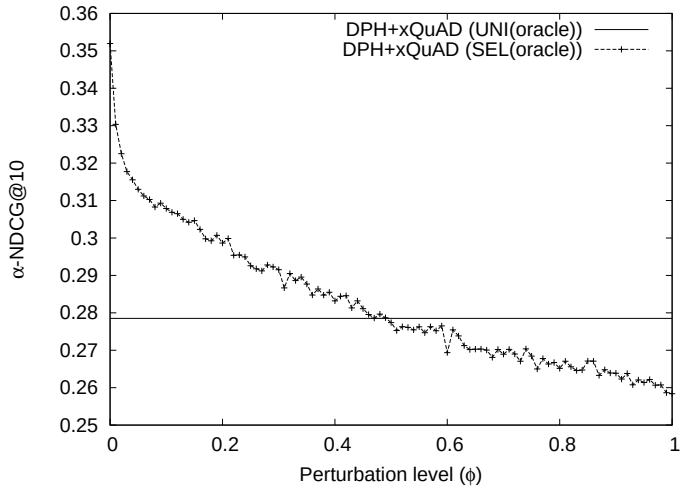


Figure 9.2: Diversification performance under an increasing prediction perturbation.

9.4 Summary

In this chapter, we have addressed the fifth claim from our thesis statement, by showing that an effective balance between promoting relevance or diversity in the ranking can be attained by inferring the level of ambiguity of each individual query. To perform such an inference, we have introduced a supervised machine learning approach based upon nearest neighbour regression to predict an effective setting for the diversification trade-off λ on a per-query basis.

In Section 9.1, we provided an overview of related approaches from the literature that selectively choose how to best rank the documents retrieved for a given query, according to some characteristic of this query, such as its predicted difficulty. In Section 9.2, we introduced our selective diversification approach, motivated by the observation that not all queries are equally ambiguous, which renders any uniform diversification strategy suboptimal. To operationalise our approach, we leveraged a large range of query features inspired by different query understanding tasks in the literature within a nearest neighbour regression task. Given a query, our approach infers an effective setting for the diversification trade-off λ according to the optimal setting observed for similar training queries. As a result, we predict not only whether a particular query could benefit from diversification, but also to what extent its retrieved documents should be diversified.

In Section 9.3, we thoroughly validated our proposed approach for improving the diversification performance of MMR and our xQuAD framework, as representatives of implicit and explicit diversification approaches. In order to validate our selective diversification regime, we contrasted it to a uniform regime, in which every query was assigned the same trade-off, learned from training data, regardless of the predicted ambiguity of each individual query. The effectiveness of selectively diversifying the retrieved documents on a per-query basis was demonstrated in Section 9.3.2.1, with significant improvements compared to deploying a uniform regime. In Section 9.3.2.2, we assessed the effectiveness of different groups of features for predicting a suitable trade-off. In particular, the query topic classification features were shown to perform particularly well across different settings. Finally, in Section 9.3.2.3, we assessed the robustness of our approach to random perturbations in its prediction accuracy. As a result, we found that our approach is extremely robust, showing improvements compared to a uniform diversification for trade-off predictions with as much as 50% of randomness.

With this chapter, we conclude the experimental validation of the xQuAD framework, as proposed in Chapter 4. In the next chapter, we recap on the contributions of this thesis, and discuss several directions for extending the xQuAD framework and its various components, as opportunities for future research.

Chapter 10

Conclusions and Future Work

Web search has grown in complexity with the growth of the Web itself. Besides the efficiency challenges posed by the ever-increasing rates of information production and consumption (Cutts, 2012), web search engines must also strive to improve their effectiveness. In particular, while the Web keeps growing, the typical length of web search queries remains short (Jansen et al., 2000). As an immediate result, queries submitted to a web search engine are often ambiguous or underspecified to some extent (Song et al., 2009). In this scenario, understanding the information need underlying each submitted query becomes a challenging task.

In this thesis, we proposed to tackle the ambiguity of a query by diversifying the documents retrieved for this query. In particular, with the multitude of users searching the Web, we argued that an ambiguous query should be viewed as representing not one, but multiple possible information needs—i.e., the needs of the different users issuing this query. By diversifying the retrieved documents with respect to these needs, the chance that the population of users will be satisfied can be improved. To this end, we introduced a novel probabilistic framework aimed to diversify the documents retrieved for an ambiguous query, by explicitly accounting for the possible information needs underlying this query.

Throughout this thesis, we thoroughly described and validated the proposed framework in light of the current literature. In the remainder of this chapter, Sections 10.1 and 10.2 summarise our main contributions and the conclusions drawn from the previous chapters, respectively. In Section 10.3, we lay out several directions for future research, directly stemming from the results of this thesis.

10.1 Summary of Contributions

In the following, we summarise the main contributions of this thesis.

A taxonomy of diversification approaches In Chapter 3, we proposed a taxonomy of existing search result diversification approaches, according to two orthogonal dimensions: aspect representation and diversification strategy. The first dimension determines how the multiple information needs underlying a query are represented as query aspects, whereas the second dimension determines how a diversification approach leverages the represented aspects in order to diversify the retrieved documents. In Section 3.3, we described the most prominent diversification approaches in the literature under this unified taxonomy, in order to enable their systematic comparison across the two considered dimensions.

A probabilistic diversification framework In Chapter 4, we introduced xQuAD, a novel framework for search result diversification. As discussed in Section 4.1, different from implicit approaches in the literature, xQuAD adopts an explicit aspect representation. In turn, different from most other explicit approaches, xQuAD's user-centric aspect representation directly represents the multiple possible information needs underlying a query, in the form of sub-queries associated with the initial query. Finally, xQuAD's ranking objective is formally defined in probabilistic terms, as demonstrated in Section 4.2. Such a theoretically sound formulation is also general, as it naturally encompasses effective features of previous approaches, as discussed in Section 4.4.

A thorough validation of the proposed framework In Chapter 5, we thoroughly validated the xQuAD framework in contrast to effective representatives of the multiple families of diversification approaches in the literature. In addition to validating the framework as a whole in Section 5.2.2.1, we also validated its key pillars. In particular, Section 5.2.2.2 validated xQuAD's hybrid diversification strategy, promoting both coverage and novelty in the ranking. In turn, Section 5.2.2.3 validated xQuAD's user-driven aspect representation in contrast to representations deployed by other approaches in the literature.

A mechanism for generating sub-queries from a query log In Chapter 6, we proposed to mine effective sub-queries from a query log. In particular, in Section 6.2, we introduced a learning to rank approach to identify effective query suggestions as potential sub-queries. In addition, in Section 6.3, we proposed an evaluation framework to quantitatively assess the effectiveness of a suggestion mechanism. Under the proposed evaluation, in Section 6.4, we validated our learning approach in contrast to a state-of-the-art suggestion mechanism, as well as to the suggestions produced by the API of a commercial web search engine.

An intent-aware mechanism for estimating coverage In Chapter 7, we proposed to improve the estimation of the probability that a document covers each sub-query. To this end, in Section 7.2, we introduced an intent-aware approach to perform such estimations. Our approach learns both the likelihood of multiple intents given a sub-query, as well as the score of the retrieved documents with respect to each intent. In Section 7.3, we thoroughly evaluated our approach in comparison to coverage estimates computed by an intent-agnostic approach.

A thorough assessment of the role of novelty In Chapter 8, we performed the first empirical investigation of the role of novelty as a diversification strategy. To this end, in Section 8.2, we proposed adaptations of existing diversification approaches in the literature, in order to enable a fair comparison of coverage and novelty in isolation from any particular aspect representation. In Section 8.3, we thoroughly evaluated novelty in comparison to and in combination with a coverage-based strategy, under a range of empirical and simulated scenarios.

A selective approach for setting the diversification trade-off In Chapter 9, we proposed a selective diversification approach, aimed at automatically determining how much to diversify the retrieved documents on a per-query basis. In Section 9.2, we formalised our approach as a nearest neighbour regression, by learning an effective trade-off between promoting relevance or diversity for an unseen query, given the optimal trade-off observed for similar training queries. In Section 9.3, we thoroughly evaluated our proposed approach in contrast to a mechanism that uniformly selects a single effective trade-off for every query.

10.2 Summary of Conclusions

In this section, we summarise the main conclusions drawn from the thorough and comprehensive evaluation of the xQuAD framework and each of its components throughout this thesis. In particular, these conclusions fully validate the statement of this thesis, as presented in Section 1.1.

On the effectiveness of xQuAD In Section 5.2.2.1, we contrasted xQuAD to effective representatives of novelty-based, coverage-based, and hybrid diversification approaches in the literature. The results of this investigation showed that xQuAD compares favourably to these approaches, with significant gains in many instances. Indeed, not only does xQuAD bring larger improvements on top of a relevance-oriented baseline, but it also performs more robustly than the considered diversification approaches, in terms of the number of queries positively and negatively affected. To understand the reasons for such a superior performance, we performed a breakdown analysis of the effectiveness of xQuAD across the diversification strategy and aspect representation dimensions. In particular, in Section 5.2.2.2, we showed that xQuAD's hybrid diversification strategy consistently outperforms the strategies deployed by the considered diversification approaches across multiple (fixed) aspect representations. In addition, in Section 5.2.2.3, we showed that, while different diversification approaches benefit more or less from different aspect representations, the user-driven representation adopted by xQuAD based on query suggestions is consistently effective for all the considered approaches. Moreover, in contrast to other diversification approaches, by incorporating a probability of relevance as part of its ranking objective, xQuAD is able to successfully leverage aspect representations that have no apparent bearing on topical relevance, such as query categories.

On the effectiveness of xQuAD's sub-query generation As demonstrated in Section 5.2.2.3, xQuAD's ranking objective is general and can be successfully deployed using different mechanisms for generating sub-queries as an aspect representation. Nevertheless, to better understand the characteristics of effective sub-queries, in Chapter 6, we introduced a learning to rank approach for generat-

ing sub-queries, identified as query suggestions from a query log. In Section 6.4.2, we quantitatively evaluated the suggestions produced by our supervised approach in comparison to those produced by a state-of-the-art query suggestion mechanism from the literature. In Sections 6.4.2.1 and 6.4.2.2, we showed that our approach significantly outperforms this baseline at producing suggestions that serve as effective replacements for the user's original query and as sub-queries for an effective diversification using xQuAD, respectively. In the latter scenario, our produced suggestions were also statistically comparable to those produced by a commercial web search engine, which arguably leverages much larger query logs than the one-month sample log used in our investigation. Another benefit of our approach, as demonstrated in Section 6.4.2.3, is its resilience to data sparsity. Indeed, our approach is able to produce effective suggestions even for a previously unseen query, provided that this query shares at least one term with relevant suggestions (or other queries related to these suggestions) in the log. Regarding the representation of candidate suggestions, our investigation in Section 6.4.2.4 showed that features dependent on the input query (computed using terms from the suggestion itself, as well as those from other queries with a common session or click with the suggestion) are the most effective descriptors of effective suggestions, denoting the topical nature of this task. Nevertheless, query-independent features reflecting lexical characteristics of a suggestion (e.g., its length) or its usage history (e.g., the amount of clicks it received across sessions) were also effective. Finally, a comprehensive analysis in Section 6.4.2.5 showed the robustness of our proposed evaluation methodology for quantifying suggestion effectiveness in light of missing document relevance assessments.

On the effectiveness of xQuAD's coverage estimates In Chapter 7, we investigated another pillar for the effectiveness of the xQuAD framework, namely, its underlying estimation of the coverage of a document with respect to multiple sub-queries. In practice, this coverage estimation can be performed as a standard estimation of the relevance of this document with respect to each individual sub-query. To this end, we built upon a relevance estimation approach traditionally deployed for web search, by exploiting the intent (e.g., informational or navigational) underlying each identified sub-query. As discussed in Section 7.2,

our proposed intent-aware diversification approach estimates the relevance of a document with respect to a sub-query by applying ranking models suitable for the predicted intents of this sub-query. In particular, in Section 7.2.2.1, we proposed two supervised intent prediction regimes within our approach: model selection, which chooses a single ranking model corresponding to the most likely intent for each sub-query, and model merging, which mixes the scores produced by multiple ranking models for each sub-query, proportionally to the likelihood of its different intents. In Section 7.3.2.1, our thorough experiments showed that the model selection regime, choosing between an informational and a navigational ranking models on a per-sub-query basis, significantly outperforms the uniform application of either of these models for all sub-queries regardless of their predicted intent. In addition, in Section 7.3.2.2, we showed that the model merging regime, which mixes the scores produced by the informational and the navigational models, performs at least as effectively as the model selection regime.

On the role of novelty as a diversification strategy In Chapter 8, we thoroughly investigated the role of novelty for search result diversification. Our goal was to assess the effectiveness of this strategy when deployed in isolation, as well as when combined with coverage into a hybrid strategy. To this end, in Section 8.3.2.1, we evaluated two novelty-based approaches using three distinct aspect representations. Contrary to the traditional view of novelty as an intuitive diversification strategy, we observed that none of the considered novelty-based approaches could consistently improve upon a non-diversified baseline, regardless of their leveraged aspect representation. In contrast, in Section 8.3.2.2, we showed that coverage-based approaches are substantially more effective than novelty-based ones. In addition, in Section 8.3.2.3, we showed that the combination of coverage and novelty into a hybrid strategy does not significantly improve upon a purely coverage-based strategy. Hence, to analyse the conditions (if any) under which novelty could be an effective strategy, we performed a thorough simulation analysis. In particular, in Section 8.3.3.1, we further demonstrated the ineffectiveness of a pure novelty-based strategy when leveraging relevance estimates of various simulated performances, computed with respect to both the initial query and its sub-queries. In Section 8.3.3.2, by simulating baseline rankings

with gradually fewer irrelevant documents, we observed a marginal improvement when promoting novelty. Finally, by analysing an extreme scenario considering only relevant documents, we showed that novelty plays a role at breaking the tie between documents with a similar coverage of the multiple aspects of the query, providing a further empirical justification for hybrid diversification strategies.

On the effectiveness of balancing the diversification trade-off In Chapter 9, we argued that the trade-off between promoting relevance or diversity in the ranking should be set on a per-query basis, according to the predicted ambiguity of each query. To this end, as discussed in Section 9.2, we introduced a selective approach for predicting an effective trade-off for an unseen query, based upon the optimal trade-off observed for similar training queries. In order to validate our proposed selective diversification regime, we contrasted it to a uniform regime, in which every query was assigned the same trade-off, learned from training data, regardless of the predicted ambiguity of each individual query. The effectiveness of selectively diversifying the retrieved documents on a per-query basis was demonstrated in Section 9.3.2.1, with significant improvements compared to deploying a uniform regime. In Section 9.3.2.2, we assessed the effectiveness of different groups of features for predicting a suitable trade-off. In particular, the query topic classification features were shown to perform particularly well across different settings. Finally, in Section 9.3.2.3, we assessed the robustness of our approach to random perturbations in its prediction accuracy. We found that our approach is extremely robust, showing improvements compared to a uniform diversification for trade-off predictions with as much as 50% of randomness.

10.3 Directions for Future Research

In this section, we discuss possible directions for future research, directly inspired by or stemming from the results of this thesis. These directions are further organised in the broad themes of estimation and modelling.

10.3.1 Estimation

As a generative probabilistic framework, the effectiveness of xQuAD is also dependent on the effectiveness of the estimation of its components. In the following, we propose directions for further improving the estimation of these components.

Multi-Source Sub-Queries As shown in Chapter 6, query suggestions are an effective representation of the multiple possible information needs underlying a query. However, our xQuAD framework is not tied to a particular sub-query generation mechanism. Indeed, as shown in Section 5.2.2.3, it can effectively leverage sub-queries produced from sources other than a query log, including non-keyword-based representations, such as categories from a topic taxonomy like ODP. In fact, xQuAD has already inspired initiatives towards exploiting alternative sub-query generation mechanisms. For instance, Plakhov (2011) proposed to leverage manually identified terms that commonly occur in queries of the same category as the user's query (e.g., "symptoms" and "treatment" are common terms for queries that fall in the category "diseases"). Relatedly, Zheng et al. (2012) proposed to exploit a hierarchical classification of the concepts in the user's query, in order to identify potential aspects belonging to different sub-categories in this hierarchy (e.g., "computer security" and "animals" are disjoint categories related to the query "worm"). More generally, Dou et al. (2011) proposed to leverage multiple sources in order to identify effective sub-queries, including anchor-text strings matching the user's query, query reformulations mined from a query log, key phrases extracted from the top retrieved documents (Zeng et al., 2004), and virtual aspects induced by grouping these documents by the domain part of their associated URL. An important question for all these approaches is how to weigh the relative importance of sub-queries derived from distinct sources, while still achieving the goal of satisfying the information needs of the user population.

Another key open problem is the identification of effective sub-queries generated implicitly, i.e., from the top retrieved documents themselves. While a pure implicit approach that performs effectively is still missing from the literature, a hybrid aspect representation was investigated by He et al. (2012). In particular, they leveraged explicit query aspects mined from multiple sources in order

to regularise an implicit aspect representation based on topic models estimated from the top retrieved documents for the query. A promising direction towards a pure implicit sub-query generation is a supervised approach aimed at learning the characteristics of effective sub-queries given only the top retrieved documents.

Non-IID Sub-Queries and Coverage Estimates A common assumption made by explicit diversification approaches is that the identified query aspects are independent and identically distributed (IID) with respect to one another. On the one hand, such an assumption greatly simplifies the underlying mathematics of these approaches and their practical instantiation. On the other hand, this assumption is unlikely to hold in a real scenario. As an example, the rankings produced for different suggestions of the same query tend to be highly correlated, particularly since these suggestions often share common terms (Ma et al., 2010). While such an assumption has not precluded the effectiveness of explicit diversification approaches such as our xQuAD framework, a promising direction for further improving their effectiveness is to relax this assumption while generating sub-queries. For instance, such sub-queries could be selected based upon the top retrieved documents (Song et al., 2011b) or the clicks that these documents received in the past (Radlinski et al., 2010a). From a statistical machine learning perspective, one possibility is to account for partial dependencies between candidate sub-queries associated with the same query (Dundar et al., 2007). In this vein, learning to rank *sets* of effective sub-queries is an open direction.

The non-IID nature of sub-queries can be also exploited by the mechanisms deployed to estimate the coverage of each retrieved document with respect to these sub-queries. For instance, in Chapter 7, we inferred the likelihood of multiple intents for a given sub-query regardless of the intents of other sub-queries identified for the same query. One possibility for further improving this and other selective approaches for coverage estimation is to once again take into account partial dependencies between these sub-queries in order to predict an assignment of intents to an entire set of sub-queries (Dundar et al., 2007). Another possibility is to leverage query (as opposed to sub-query) features for this purpose (Macdonald, Santos & Ounis, 2012b). As an intuitive example, sub-queries identified for a navigational query could be more likely to be themselves navigational.

Holistic Diversification Thus far, we have discussed aspect representation and diversification strategy as two orthogonal dimensions. In particular, in Chapters 6 and 7, we introduced sub-query generation and coverage estimation techniques that make no assumptions about each other. While such an independence contributes to the generality of xQuAD, an interesting direction for further improving the effectiveness of the framework is to generate sub-queries and to infer the coverage of documents in an integrated process. Such a process could be seen as a holistic diversification, aimed to produce at the same time diverse sub-queries and a diverse document ranking. To this end, a natural direction is to exploit random walks in a query-click graph, so as to directly account for the sub-queries covered by different documents. However, this approach could suffer from data sparsity, particularly since typically only a few documents are clicked by web search users (Jansen et al., 2000). To overcome this limitation, bipartite graphs could be inferred from content-enriched representations of sub-queries (e.g., using the structured virtual document representation proposed in Section 6.2.1) and of the top retrieved (as opposed to the clicked) documents for the query.

10.3.2 Modelling

Besides improving the estimation of the various components of xQuAD, another possible direction for future research is on extending the framework for search result diversification in specialised search scenarios, as we discuss in this section.

Aggregated Diversification Existing diversification approaches have been deployed mostly in the context of web (e.g., Agrawal et al., 2009; Rafiei et al., 2010; Santos et al., 2010a) and newswire (e.g., Carbonell & Goldstein, 1998; Chen & Karger, 2006; Zhai et al., 2003; Wang & Zhu, 2009) search, but there have also been approaches dedicated to diversifying image (e.g., Paramita et al., 2009; van Leuken et al., 2009), product (e.g., Vee et al., 2008; Gollapudi & Sharma, 2009), and blog (Demartini, 2011; Santos et al., 2012a) search results. Our initial analysis of ambiguous queries using searching behaviour data from four Google verticals (web, image, news, and product search) showed that the ambiguity of a single query varies considerably across different verticals (Santos et al., 2011a). Such a

variation is observed not only in terms of the aspects underlying a query in different verticals, but also in terms of the likelihood of these different aspects. With the prevalence of aggregated search interfaces in modern web search (Murdock & Lalmas, 2008; Diaz et al., 2010), an open question faced by web search engines is how to tackle query ambiguity across multiple search verticals.

In this vein, we have proposed an extension of xQuAD to tackle the aggregated search result diversification problem (Santos et al., 2011a). Following the greedy approach in Algorithm 3.1, given a query q and an initial ranking $\mathcal{R}_q$, comprising search results from multiple verticals $\vartheta \in \Theta$ triggered by this query, and a set $\mathcal{D}_q \subseteq \mathcal{R}_q$ with the search results selected in the previous iterations of the algorithm, we defined the ranking objective of xQuAD^{agg} as follows:

$$f_{\text{xQuAD}^{agg}}(q, d, \mathcal{D}_q) = \sum_{\vartheta \in \Theta} p(\vartheta|q) \Big[(1 - \lambda_\vartheta)\, p(d|q, \vartheta) + \lambda_\vartheta\, p(d, \bar{\mathcal{D}}_{q|\vartheta}|q, \vartheta) \Big], \quad (10.1)$$

where $p(\vartheta|q)$ is the probability of selecting the vertical ϑ for the query q, while $p(d|q, \vartheta)$ and $p(d, \bar{\mathcal{D}}_{q|\vartheta}|q, \vartheta)$ are vertical-specific instantiations of xQuAD's relevance and diversity probabilities in Equation (4.1). Accordingly, λ_ϑ is the vertical-specific diversification trade-off, denoting the expected ambiguity of the query q in the scope of the vertical ϑ. Besides open questions regarding the estimation of the various components that emerge from this extended formulation (i.e., vertical-specific sub-query generation, coverage, and novelty), an interesting modelling question also arises. In particular, the extended formulation in Equation (10.1) takes a *local* approach, by diversifying the search results within each vertical, and then aggregating the rankings produced from the various verticals. As a result, redundancy is penalised only within each vertical, but not across different verticals. In practice, we assume that similar search results of the same type (e.g., two videos about the same event) may be redundant, but similar results of different types (e.g., a video and a news story covering the same event) may be actually complementary. Another plausible formulation could take a *global* approach, namely, by aggregating the search results from multiple verticals first, and only then performing a diversification. Given the lack of a shared test collection for aggregated search evaluation, the empirical validation of these proposed complementary approaches is also left as an open direction for investigation.

Personalised Diversification The experiments in this thesis assumed a conservative search scenario, in which a query is the only evidence of a particular user's information need available to the search engine. Nonetheless, web search engines are increasingly gathering additional information about individual search users, including their previously issued queries and clicked documents in the session, their recently browsed websites, and their profile across multiple social networking websites. Such evidence could be directly incorporated within our xQuAD framework in order to perform a personalised search result diversification. Indeed, Vallet & Castells (2012) proposed one such extension, denoted personalised xQuAD, which can be defined as follows:

$$f_{\text{xQuAD}^{per}}(u, q, d, \mathcal{D}_q) = (1 - \lambda_u)\, p(d|q, u) + \lambda_u\, p(d, \bar{\mathcal{D}}_q|q, u), \tag{10.2}$$

where $p(d|q, u)$ and $p(d, \bar{\mathcal{D}}_q|q, u)$ are the probabilities of relevance and diversity conditioned on the user u, respectively. In turn, λ_u represents the user-specific diversification trade-off, denoting how ambiguous the query q is for the user u. This extended formulation of xQuAD's ranking objective (Equation (4.1)) opens up interesting directions regarding the estimation of the several components of the framework within the universe of a single user. In particular, inferring how ambiguous a given query is and which sub-queries are plausible given the current user's profile can be challenging tasks, primarily since this profile is typically sparse (Shahabi & Chen, 2003). A possible solution in this direction is to augment the user's profile by leveraging the preferences of similar users that issued the same query. For instance, such a group-based personalised diversification could be performed by exploiting the user's social circle (Carmel et al., 2009).

Discriminative Diversification Throughout this thesis, we have used machine learning to improve the estimation of several components of the xQuAD framework. For instance, in Chapter 5, we used learning to rank for an improved estimation of relevance and coverage, while in Chapter 6 learning to rank was used to generate more effective sub-queries. In turn, Chapters 7 and 9 deployed classification and regression techniques to infer the likelihood of different sub-query intents and the ambiguity of different queries, respectively. Despite having

been shown to be effective at estimating the various components of xQuAD, these approaches were deployed independently of one another. While such a componentised approach contributes to the generality of the framework, by allowing for alternative estimation techniques to be easily deployed, it would be desirable to have a unified process for learning to diversify the retrieved documents given suitable training data. As described in Section 3.3, existing approaches for learning to diversify are either based on implicit aspect representations (Yue & Joachims, 2008; Raman et al., 2012) or leverage an explicit aspect representation in an online learning setting (Radlinski et al., 2008a; Slivkins et al., 2010).

We are currently investigating an extension of xQuAD for learning to rank for diversity in a traditional offline setting, which is a recognised open challenge in learning to rank (Chapelle & Chang, 2011). In particular, we have devised a meta learning framework that directly leverages existing learning to rank algorithms—such as those introduced in Section 2.2.3.2—in order to produce effective ranking models that reward diversity. To this end, our proposed framework extends a standard feature space for learning to rank comprising, e.g., query-dependent and query-independent document features, into a space augmented towards two orthogonal axes. On the *coverage* axis, the feature space is augmented to include features that estimate the relevance of a document with respect to multiple sub-queries. On the *novelty* axis, the space is augmented by taking into account the diminishing relevance of each document in light of the selection of other documents to compose a diverse ranking. In practice, we can employ xQuAD's diversity component from Equation (4.6) as a mechanism for generating sub-query-dependent features that leverage these augmented axes. Precisely, a sub-query-dependent feature ϕ can be computed according to:

$$p_\phi(d|q) = \Psi_{s \in \mathcal{S}}\, p(s|q)\, p_\phi(d|q,s) \prod_{d_j \in \mathcal{D}_q} p_\phi(\bar{d}_j|q,s), \tag{10.3}$$

where $p_\phi(d|q,s)$ denotes the estimated coverage of the document d with respect to the sub-query s, according to the feature ϕ, which could be any of the query-dependent ranking approaches described in Section 2.2.1, such as BM25 (Equation (2.13)) or DPH (Equation (2.31)). The coverage probabilities produced for all sub-queries $s \in \mathcal{S}_q$ using the feature ϕ are then aggregated using a function

Ψ, which in turn could be any summary statistic, such as mean or maximum. Our initial results for this approach in TREC 2012 are promising (Limsopatham, McCreadie, Albakour, Macdonald, Santos & Ounis, 2012).

10.4 Final Remarks

This thesis contributed a novel framework for the search result diversification problem. As demonstrated throughout the thesis, the principles underlying the xQuAD framework are general, sound, and effective. From a research perspective, the generality of the framework enabled the investigation of several dimensions of the diversification problem, including how to best represent the possible aspects underlying a query, how to estimate the relevance of a document with respect to multiple aspects, and how to tailor the diversification for the level of ambiguity of different queries. These investigations led to the publication of 11 peer-reviewed research papers and 5 evaluation forum reports directly related to this thesis. Moreover, as discussed in Section 10.3, this thesis opened up directions for other researchers, who deployed and extended the xQuAD framework for different applications. From a practical perspective, xQuAD has been subjected to scrutiny from the research community as a regular contender in internationally renown evaluation forums, such as TREC and NTCIR. As the winning entry in all editions of the diversity task of the TREC Web track,[1] we believe that the xQuAD framework has secured its place in the state-of-the-art.

[1]Best cat. B submission in TREC 2009 (Clarke et al., 2009b) and 2010 (Clarke et al., 2010); best overall submission in TREC 2011 (Clarke et al., 2011b) and 2012 (Clarke et al., 2012).

References

Adar, E., Teevan, J., Dumais, S.T. & Elsas, J.L. (2009). The Web changes everything: Understanding the dynamics of web content. In *Proceedings of the 2nd ACM International Conference on Web Search and Data Mining*, 282–291, ACM, Barcelona, Spain. 13

Agrawal, R., Gollapudi, S., Halverson, A. & Ieong, S. (2009). Diversifying search results. In *Proceedings of the 2nd ACM International Conference on Web Search and Data Mining*, 5–14, ACM, Barcelona, Spain. 2, 60, 63, 64, 73, 78, 79, 85, 87, 90, 97, 107, 109, 110, 173, 217

Aha, D.W., Kibler, D.F. & Albert, M.K. (1991). Instance-based learning algorithms. *Machine Learning*, **6**, 37–66. 150, 191, 193, 194

Ahmad, F. & Kondrak, G. (2005). Learning a spelling error model from search query logs. In *Proceedings of the 2005 Conference on Empirical Methods in Natural Language Processing*, 955–962, ACL, Vancouver, BC, Canada. 16

Alonso, O., Rose, D.E. & Stewart, B. (2008). Crowdsourcing for relevance evaluation. *SIGIR Forum*, **42**, 9–15. 48, 146

Alpert, J. & Hajaj, N. (2008). We knew the Web was big... `http://googleblog.blogspot.com/2008/07/we-knew-web-was-big.html` (August 2012). 16

Amati, G. (2003). *Probability models for information retrieval based on Divergence From Randomness*. Ph.D. thesis, University of Glasgow. 31, 32, 33, 34, 105

Amati, G. (2006). Frequentist and Bayesian approach to information retrieval. In *Proceedings of the 28th European Conference on IR Research on Advances in Information Retrieval*, 13–24, Springer, London, UK. 31, 34

Amati, G. & van Rijsbergen, C.J. (2002). Probabilistic models of information retrieval based on measuring the divergence from randomness. *ACM Transactions on Information Systems*, **20**, 357–389. 31, 33

Amati, G., Carpineto, C., Romano, G. & Bordoni, F.U. (2004). Query difficulty, robustness and selective application of query expansion. In *Proceedings of the 26th European Conference on IR Research on Advances in Information Retrieval*, 127–137, Springer. 157

Amati, G., Ambrosi, E., Bianchi, M., Gaibisso, C. & Gambosi, G. (2007). FUB, IASI-CNR and University of Tor Vergata at TREC 2007 Blog track. In *Proceedings of the 16th Text REtrieval Conference*, Gaithersburg, MD, USA. 34, 105

Arasu, A., Cho, J., Garcia-Molina, H., Paepcke, A. & Raghavan, S. (2001). Searching the Web. *ACM Transactions on Internet Technology*, **1**, 2–43. 10, 16

Ashkan, A. & Clarke, C.L.A. (2011). On the informativeness of cascade and intent-aware effectiveness measures. In *Proceedings of the 20th International Conference on World Wide Web*, 407–416, ACM, Hyderabad, India. 82

Aslam, J.A., Yilmaz, E. & Pavlu, V. (2005). The maximum entropy method for analyzing retrieval measures. In *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 27–34, ACM, Salvador, Brazil. 82

Baeza-Yates, R., Hurtado, C. & Mendoza, M. (2004). Query recommendation using query logs in search engines. In *Proceedings of the 9th International Conference on Current Trends in Database Technology*, 588–596, Springer-Verlag, Heraklion, Greece. 2, 121, 122

Baeza-Yates, R.A. & Ribeiro-Neto, B. (2011). *Modern Information Retrieval*. Pearson Education Ltd., Harlow, UK, 2nd edn. 9, 13, 16, 20

Becchetti, L., Castillo, C., Donato, D., Leonardi, S. & Baeza-Yates, R. (2006). Link-based characterization and detection of web spam. In *Proceedings of the 2nd International Workshop on Adversarial Information Retrieval on the Web*, Seattle, WA, USA. 158

Beitzel, S.M., Jensen, E.C., Frieder, O., Grossman, D., Lewis, D.D., Chowdhury, A. & Kolcz, A. (2005). Automatic web query classification using labeled and unlabeled training data. In *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 581–582, ACM, Salvador, Brazil. 17

Bendersky, M., Croft, W.B. & Diao, Y. (2011). Quality-biased ranking of web documents. In *Proceedings of the 4th ACM International Conference on Web Search and Data Mining*, 95–104, ACM, Hong Kong, China. 36, 38

Bennett, P.N., Carterette, B., Chapelle, O. & Joachims, T. (2008). Beyond binary relevance: Preferences, diversity, and set-level judgments. *SIGIR Forum*, **42**, 53–58. 58

Berger, A. & Lafferty, J. (1999). Information retrieval as statistical translation. In *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 222–229, ACM, Berkeley, CA, USA. 26

Bergman, M.K. (2001). The deep Web: Surfacing hidden value. *Journal of Electronic Publishing*, **7**. 12

Bergsma, S. & Wang, Q.I. (2007). Learning noun phrase query segmentation. In *Proceedings of the 2007 Conference on Empirical Methods in Natural Language Processing*, 819–826, ACL, Prague, Czech Republic. 17

Berners-Lee, T. (1989). Information management: A proposal. Tech. rep., CERN, Genf. 10

Berry, D. & Fristedt, B. (1985). *Bandit problems: Sequential allocation of experiments*. Chapman and Hall. 71

Blei, D.M., Ng, A.Y. & Jordan, M.I. (2003). Latent dirichlet allocation. *The Journal of Machine Learning Research*, **3**, 993–1022. 69

Blondel, V.D., Guillaume, J.L., Lambiotte, R. & Lefebvre, E. (2008). Fast unfolding of communities in large networks. *Journal of Statistical Mechanics*, **2008**, P10008+. 76

Boldi, P., Bonchi, F., Castillo, C., Donato, D., Gionis, A. & Vigna, S. (2008). The query-flow graph: model and applications. In *Proceedings of the 17th ACM Conference on Information and Knowledge Management*, 609–618, ACM, Napa Valley, CA, USA. 122

Boldi, P., Bonchi, F., Castillo, C., Donato, D. & Vigna, S. (2009a). Query suggestions using query-flow graphs. In *Proceedings of the 2009 Workshop on Web Search Click Data*, 56–63, ACM. 122, 123, 135

Boldi, P., Bonchi, F., Castillo, C. & Vigna, S. (2009b). From "Dango" to "Japanese cakes": Query reformulation models and patterns. In *Proceedings of the 2009 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology*, 183–190, IEEE Computer Society, Milan, Italy. 70

Bookstein, A. & Swanson, D.R. (1974). Probabilistic models for automatic indexing. *Journal of the American Society for Information Science*, **25**, 312–316. 31

Borlund, P. (2003). The concept of relevance in IR. *Journal of the American Society for Information Science and Technology*, **54**, 913–925. 47, 48

Borlund, P. & Ingwersen, P. (1997). The development of a method for the evaluation of interactive information retrieval systems. *Journal of Documentation*, **53**, 225–250. 48

Box, G.E.P., Hunter, J.S. & Hunter, W.G. (2005). *Statistics for Experimenters: Design, Innovation, and Discovery*. Wiley-Interscience, 2nd edn. 101

Boyce, B.R. (1982). Beyond topicality: A two stage view of relevance and the retrieval process. *Information Processing and Management*, **18**, 105–109. 36

Brin, S. & Page, L. (1998). The anatomy of a large-scale hypertextual web search engine. *Computer Networks and ISDN Systems*, **30**, 107–117. 41

Broccolo, D., Marcon, L., Nardini, F.M., Perego, R. & Silvestri, F. (2012). Generating suggestions for queries in the long tail with an inverted index. *Information Processing and Management*, **48**, 326–339. xiii, 122, 123, 125, 126, 127, 135, 136, 137, 138, 139, 140, 141, 142, 143

Broder, A. (2002). A taxonomy of web search. *SIGIR Forum*, **36**, 3–10. 9, 36, 47, 56, 76, 78, 149, 150, 155

Broder, A., Kumar, R., Maghoul, F., Raghavan, P., Rajagopalan, S., Stata, R., Tomkins, A. & Wiener, J. (2000). Graph structure in the Web. *Computer Networks*, **33**, 309–320. 10, 11

Burges, C., Shaked, T., Renshaw, E., Lazier, A., Deeds, M., Hamilton, N. & Hullender, G. (2005). Learning to rank using gradient descent. In *Proceedings of the 22nd International Conference on Machine Learning*, 89–96, ACM, Bonn, Germany. 51

Burges, C.J.C., Ragno, R. & Le, Q.V. (2006). Learning to rank with nonsmooth cost functions. In *Proceedings of the 20th Annual Conference on Neural Information Processing Systems*, 193–200, MIT Press, Vancouver, BC, Canada. 46

Calderón-Benavides, L. (2011). *Unsupervised identification of the user's query intent in web search*. Ph.D. thesis, Universitat Pompeu Fabra. 150

Cambazoglu, B.B., Zaragoza, H., Chapelle, O., Chen, J., Liao, C., Zheng, Z. & Degenhardt, J. (2010). Early exit optimizations for additive machine learned ranking systems. In *Proceedings of the 3rd ACM International Conference on Web Search and Data Mining*, 411–420, ACM, New York, NY, USA. 17, 19

Cao, G., Nie, J.Y. & Bai, J. (2005). Integrating word relationships into language models. In *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 298–305, ACM, Salvador, Brazil. 27

Capannini, G., Nardini, F.M., Perego, R. & Silvestri, F. (2011). Efficient diversification of web search results. *Proceedings of the VLDB Endowment*, **4**, 451–459. 64, 70, 85

Carbonell, J. & Goldstein, J. (1998). The use of MMR, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval*, 335–336, ACM, Melbourne, Australia. 2, 63, 64, 65, 85, 97, 107, 110, 169, 170, 171, 199, 217

Carmel, D. & Yom-Tov, E. (2010). Estimating the query difficulty for information retrieval. *Synthesis Lectures on Information Concepts, Retrieval, and Services*, **2**, 1–89. 157, 190, 197

Carmel, D., Zwerdling, N., Guy, I., Ofek-Koifman, S., Har'el, N., Ronen, I., Uziel, E., Yogev, S. & Chernov, S. (2009). Personalized social search based on the user's social network. In *Proceedings of the 18th ACM Conference on Information and Knowledge Management*, 1227–1236, ACM, Hong Kong, China. 219

Carpineto, C. & Romano, G. (2012). A survey of automatic query expansion in information retrieval. *ACM Computing Surveys*, **44**, 1:1–1:50. 17

Carterette, B. (2009). An analysis of NP-completeness in novelty and diversity ranking. In *Proceedings of the 2nd International Conference on Theory of Information Retrieval*, 200–211, Springer-Verlag, Cambridge, UK. 2, 82

Carterette, B. & Chandar, P. (2009). Probabilistic models of ranking novel documents for faceted topic retrieval. In *Proceedings of the 18th ACM Conference on Information and Knowledge Management*, 1287–1296, ACM, Hong Kong, China. 63, 64, 68, 85, 90

Carterette, B., Allan, J. & Sitaraman, R. (2006). Minimal test collections for retrieval evaluation. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 268–275, ACM, Seattle, WA, USA. 146

Carterette, B., Pavlu, V., Kanoulas, E., Aslam, J.A. & Allan, J. (2009a). If I dad a million queries. In *Proceedings of the 31th European Conference on IR Research on Advances in Information Retrieval*, 288–300, Springer-Verlag, Toulouse, France. 146

Carterette, B., Pavluz, V., Fangx, H. & Kanoulas, E. (2009b). Million Query track 2009 overview. In *Proceedings of the 18th Text REtrieval Conference*, Gaithersburg, MD, USA. 158

Carvalho, V.R., Lease, M. & Yilmaz, E. (2011). Crowdsourcing for search evaluation. *SIGIR Forum*, **44**, 17–22. 48

Castillo, C. (2004). *Effective web crawling*. Ph.D. thesis, University of Chile. 11, 12

Castillo, C. & Davison, B.D. (2011). Adversarial web search. *Foundations and Trends in Information Retrieval*, **4**, 377–486. 36, 38, 40

Castillo, C., Donato, D., Gionis, A., Murdock, V. & Silvestri, F. (2007). Know your neighbors: Web spam detection using the web topology. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 423–430, ACM. 40

Chakrabarti, D., Kumar, R. & Punera, K. (2007). Page-level template detection via isotonic smoothing. In *Proceedings of the 16th International Conference on World Wide Web*, 61–70, ACM, Banff, AB, Canada. 14

Chapelle, O. & Chang, Y. (2011). Yahoo! Learning to Rank Challenge overview. *Journal of Machine Learning Research, Proceedings Track*, **14**, 1–24. 45, 220

Chapelle, O., Metlzer, D., Zhang, Y. & Grinspan, P. (2009). Expected reciprocal rank for graded relevance. In *Proceedings of the 18th ACM Conference on*

Information and Knowledge Management, 621–630, ACM, Hong Kong, China. 52, 80

Chapelle, O., Ji, S., Liao, C., Velipasaoglu, E., Lai, L. & Wu, S.L. (2011). Intent-based diversification of web search results: Metrics and algorithms. *Information Retrieval*, **14**, 572–592. 80, 82

Chapelle, O., Joachims, T., Radlinski, F. & Yue, Y. (2012). Large-scale validation and analysis of interleaved search evaluation. *ACM Transactions on Information Systems*, **30**, 1–41. 48

Chen, H. & Karger, D.R. (2006). Less is more: Probabilistic models for retrieving fewer relevant documents. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 429–436, ACM, Seattle, WA, USA. 2, 54, 64, 66, 217

Cho, J., Garcia-Molina, H. & Page, L. (1998). Efficient crawling through URL ordering. In *Proceedings of the 7th International Conference on World Wide Web*, 161–172, Elsevier, Brisbane, Australia. 12

Clarke, C.L.A., Kolla, M., Cormack, G.V., Vechtomova, O., Ashkan, A., Büttcher, S. & MacKinnon, I. (2008). Novelty and diversity in information retrieval evaluation. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 659–666, ACM, Singapore, Singapore. 1, 2, 55, 58, 80, 81, 90

Clarke, C.L.A., Craswell, N. & Soboroff, I. (2009a). Overview of the TREC 2009 Web track. In *Proceedings of the 18th Text REtrieval Conference*, Gaithersburg, MD, USA. 3, 6, 76, 82, 101, 102, 103, 118, 134, 135, 144, 161, 175, 192, 199

Clarke, C.L.A., Kolla, M. & Vechtomova, O. (2009b). An effectiveness measure for ambiguous and underspecified queries. In *Proceedings of the 2nd International Conference on Theory of Information Retrieval*, 188–199, Springer-Verlag, Cambridge, UK. 81, 221

Clarke, C.L.A., Craswell, N., Soboroff, I. & Cormack, G.V. (2010). Preliminary overview of the TREC 2010 Web track. In *Proceedings of the 19th Text RE-*

trieval Conference, Gaithersburg, MD, USA. 3, 6, 76, 101, 102, 103, 118, 134, 135, 144, 161, 175, 181, 186, 221

Clarke, C.L.A., Craswell, N., Soboroff, I. & Ashkan, A. (2011a). A comparative analysis of cascade measures for novelty and diversity. In *Proceedings of the 4th ACM International Conference on Web Search and Data Mining*, 75–84, ACM, Hong Kong, China. 80, 82, 104

Clarke, C.L.A., Craswell, N., Soboroff, I. & Voorhees, E.M. (2011b). Overview of the TREC 2011 Web track. In *Proceedings of the 20th Text REtrieval Conference*, Gaithersburg, MD, USA. 3, 6, 76, 101, 102, 103, 118, 134, 141, 144, 221

Clarke, C.L.A., Craswell, N. & Voorhees, E.M. (2012). Overview of the TREC 2012 Web track. In *Proceedings of the 21st Text REtrieval Conference*, Gaithersburg, MD, USA. 6, 76, 221

Cleverdon, C. (1967). The Cranfield tests on index language devices. *Aslib Proceedings*, **19**, 173–194. 48

Cleverdon, C.W. (1991). The significance of the Cranfield tests on index languages. In *Proceedings of the 14th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 3–12, ACM, Chicago, IL, USA. 74

Cleverdon, C.W. & Keen, M. (1962). Report on the testing and analysis of an investigation into the comparative efficiency of indexing systems. *ASLIB Cranfield Research Project*, **2**. 50

Cleverdon, C.W. & Keen, M. (1966). Factors determining the performance of indexing systems. *ASLIB Cranfield Research Project*, **2**. 50

Clough, P., Sanderson, M., Abouammoh, M., Navarro, S. & Paramita, M. (2009). Multiple approaches to analysing query diversity. In *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 734–735, ACM, Boston, MA, USA. 56, 157, 197

Codd, E.F. (1970). A relational model of data for large shared data banks. *Communications of the ACM*, **13**, 377–387. 9

Cooper, W.S. (1971). The inadequacy of probability of usefulness as a ranking criterion for retrieval system output. Tech. rep., University of California, Berkeley, Berkeley, CA, USA. 2, 21, 54, 57, 67

Cooper, W.S. (1995). Some inconsistencies and misidentified modeling assumptions in probabilistic information retrieval. *ACM Transactions on Information Systems*, **13**, 100–111. 57

Cormack, G.V., Smucker, M.D. & Clarke, C.L.A. (2011). Efficient and effective spam filtering and re-ranking for large web datasets. *Information Retrieval*, **14**, 441–465. 39

Cormen, T.H., Leiserson, C.E., Rivest, R.L. & Stein, C. (2001). *Introduction to Algorithms*. The MIT Press, 2nd edn. 62

Craswell, N. & Hawking, D. (2004). Overview of the TREC 2004 Web track. In *Proceedings of the 13th Text REtrieval Conference*, Gaithersburg, MD, USA. 150, 154

Craswell, N. & Szummer, M. (2007). Random walks on the click graph. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 239–246, ACM, Amsterdam, The Netherlands. 76

Craswell, N., Hawking, D. & Robertson, S. (2001). Effective site finding using link anchor information. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 250–257, ACM, New Orleans, LA, USA. 16

Craswell, N., Zoeter, O., Taylor, M. & Ramsey, B. (2008). An experimental comparison of click position-bias models. In *Proceedings of the 1st International Conference on Web Search and Data Mining*, 87–94, ACM. 47, 52, 58, 80, 104

Croft, W.B., Metzler, D. & Strohman, T. (2009). *Search Engines: Information Retrieval in Practice*. Addison-Wesley Publishing Company, USA, 1st edn. 14

Cronen-Townsend, S. & Croft, W.B. (2002). Quantifying query ambiguity. In *Proceedings of the 2nd International Conference on Human Language Technology Research*, 104–109, Morgan Kaufmann Publishers Inc., San Diego, CA, USA. 55

Cronen-Townsend, S., Zhou, Y. & Croft, W.B. (2002). Predicting query performance. In *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 299–306, ACM, Tampere, Finland. 157, 198

Cui, H., Wen, J.R., Nie, J.Y. & Ma, W.Y. (2002). Probabilistic query expansion using query logs. In *Proceedings of the 11th International Conference on World Wide Web*, 325–332, ACM, Honolulu, HI, USA. 17

Cutts, M. (2012). Spotlight keynote. In *Proceedings of Search Engine Strategies*, San Francisco, CA, USA. 1, 10, 12, 208

Damerau, F.J. (1965). An experiment in automatic indexing. *American Documentation*, **16**, 283–289. 31

Dang, V., Bendersky, M. & Croft, W.B. (2010). Learning to rank query reformulations. In *Proceedings of the 33rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 807–808, ACM, Geneva, Switzerland. 123

Das, A. & Jain, A. (2012). *Indexing the World Wide Web: The journey so far*, chap. 1, 1–28. IGI Global. 16

Dean, J. (2009). Challenges in building large-scale information retrieval systems. In *Proceedings of the 2nd ACM International Conference on Web Search and Data Mining*, 1, ACM, Barcelona, Spain. 13, 125

Demartini, G. (2011). ARES: a retrieval engine based on sentiments sentiment-based search result annotation and diversification. In *Proceedings of the 33rd European Conference on IR Research on Advances in Information Retrieval*, 772–775, Springer-Verlag, Dublin, Ireland. 217

Diaz, F., Lalmas, M. & Shokouhi, M. (2010). From federated to aggregated search. In *Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 910. 218

Dirac, P.A.M. (1930). *The Principles of Quantum Mechanics*. Clarendon Press, Oxford, UK. 67

Dou, Z., Hu, S., Chen, K., Song, R. & Wen, J.R. (2011). Multi-dimensional search result diversification. In *Proceedings of the fourth ACM international Conference on Web Search and Data Mining*, 475–484, ACM, Hong Kong, China. 215

Downey, D., Dumais, S. & Horvitz, E. (2007). Heads and tails: studies of web search with common and rare queries. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 847–848, ACM, Amsterdam, The Netherlands. 122

Dundar, M., Krishnapuram, B., Bi, J. & Rao, R.B. (2007). Learning classifiers when the training data is not IID. In *Proceedings of the 20th International Joint Conference on Artifical Intelligence*, 756–761, Morgan Kaufmann Publishers Inc., Hyderabad, India. 216

Edwards, J., McCurley, K. & Tomlin, J. (2001). An adaptive model for optimizing performance of an incremental web crawler. In *Proceedings of the 10th International Conference on World Wide Web*, 106–113, ACM, Hong Kong, Hong Kong. 13

Evert, S. (2008). A lightweight and efficient tool for cleaning web pages. In *Proceedings of the 6th International Conference on Language Resources and Evaluation*, Marrakech, Morocco. 14

Feige, U. (1998). A threshold of $\ln(n)$ for approximating set cover. *Journal of the ACM*, **45**, 634–652. 62, 86

Feller, W. (1968). *An Introduction to Probability Theory and its Applications*, vol. 1. Wiley. 33, 34

Fetterly, D., Manasse, M., Najork, M. & Wiener, J.L. (2004). A large-scale study of the evolution of web pages. *Software: Practice and Experience*, **34**, 213–237. 13, 24

Fisher, R.A. (1922). On the mathematical foundations of theoretical statistics. *Philosophical Transactions of the Royal Society of London Series A*, **222**, 309–368. 29

Fonseca, B.M., Golgher, P.B., De Moura, E.S., Pôssas, B. & Ziviani, N. (2003). Discovering search engine related queries using association rules. *Journal of Web Engineering*, **2**, 215–227. 121

Friedman, J.H. (2001). Greedy function approximation: A gradient boosting machine. *The Annals of Statistics*, **29**, 1189–1232. 46

Fuhr, N. (1989). Optimum polynomial retrieval functions based on the probability ranking principle. *ACM Transactions on Information Systems*, **7**, 183–204. 42

Gao, J., Nie, J.Y., Wu, G. & Cao, G. (2004). Dependence language model for information retrieval. In *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 170–177, ACM. 27

Geng, X., Liu, T.Y., Qin, T., Arnold, A., Li, H. & Shum, H.Y. (2008). Query dependent ranking using k-nearest neighbor. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 115–122, ACM, Singapore, Singapore. 149, 150, 151, 191, 194

Gil-Costa, V., Santos, R.L.T., Macdonald, C. & Ounis, I. (2011). Sparse spatial selection for novelty-based search result diversification. In *Proceedings of the 18th International Symposium on String Processing and Information Retrieval*, 344–355, Springer, Pisa, Italy. 5, 64, 67, 68, 85, 176

Gil-Costa, V., Santos, R.L.T., Macdonald, C. & Ounis, I. (2013). Modelling efficient novelty-based search result diversification in metric spaces. *Journal of Discrete Algorithms*. 5, 64, 68, 85, 176

Goffman, W. (1964). On relevance as a measure. *Information Storage and Retrieval*, **2**, 201–203. 2, 9, 57

Gollapudi, S. & Sharma, A. (2009). An axiomatic approach for result diversification. In *Proceedings of the 18th International Conference on World Wide Web*, 381–390, ACM, Madrid, Spain. 58, 217

Good, I.J. (1950). *Probability and the weighing of evidence*. C. Griffin, London, UK. 88, 90

Goodman, J. & tau Yih, W. (2006). Online discriminative spam filter training. In *Proceedings of the 3rd Conference on Email and Anti-Spam*, Mountain View, CA, USA. 39

Gordon, M.D. & Lenk, P. (1991). A utility theoretic examination of the probability ranking principle in information retrieval. *Journal of the American Society for Information Science and Technology*, **42**, 703–714. 57

Gordon, M.D. & Lenk, P. (1992). When is the probability ranking principle suboptimal? *Journal of the American Society for Information Science and Technology*, **43**, 1–14. 54, 57

Gudivada, V.N., Raghavan, V.V., Grosky, W.I. & Kasanagottu, R. (1997). Information retrieval on the World Wide Web. *IEEE Internet Computing*, **1**, 58–68. 17

Guo, J., Xu, G., Cheng, X. & Li, H. (2009). Named entity recognition in query. In *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 267–274, ACM, Boston, MA, USA. 17

Harman, D. (1993). Overview of the second Text REtrieval Conference (TREC-2). In *Proceedings of the 2nd Text REtrieval Conference*, Gaithersburg, MD, USA. 51

Harter, S.P. (1975a). A probabilistic approach to automatic keyword indexing. Part I: On the distribution of specialty words in a technical literature. *Journal*

of the American Society for Informaiton Science, **26**, 197–206. 21, 23, 24, 31, 54

Harter, S.P. (1975b). A probabilistic approach to automatic keyword indexing. Part II: An algorithm for probabilistc indexing. *Journal of the American Society for Informaiton Science*, **26**, 280–289. 21, 23, 24, 31, 54

Hauff, C., Kelly, D. & Azzopardi, L. (2010). A comparison of user and system query performance predictions. In *Proceedings of the 19th ACM international Conference on Information and Knowledge Management*, 979–988, ACM, Toronto, ON, Canada. 123, 131

He, B. & Ounis, I. (2003). A study of parameter tuning for term frequency normalization. In *Proceedings of the 12th International Conference on Information and Knowledge Management*, 10–16, ACM, New Orleans, LA, USA. 33

He, B. & Ounis, I. (2006). Query performance prediction. *Information Systems*, **31**, 585–594. 157

He, B. & Ounis, I. (2007). Combining fields for query expansion and adaptive query expansion. *Information Processing and Management*, **43**, 1294–1307. 17

He, J., Meij, E. & de Rijke, M. (2011). Result diversification based on query-specific cluster ranking. *Journal of the American Society for Information Science and Technology*, **62**, 550–571. 63, 64, 69, 85

He, J., Hollink, V. & de Vries, A. (2012). Combining implicit and explicit topic representations for result diversification. In *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 851–860, ACM, Portland, OR, USA. 215

Hearst, M.A. (2009). *Search User Interfaces*. Cambridge University Press. 2

Hersh, W.R. & Over, P. (1999). Trec-8 interactive track report. In *Proceedings of the 8th Text REtrieval Conference*, Gaithersburg, MD, USA. 75

Hiemstra, D. (1998). A linguistically motivated probabilistic model of information retrieval. In *Proceedings of the 2nd European Conference on Research and Advanced Technology for Digital Libraries*, 569–584, Springer, Heraklion, Greece. 26

Hintikka, J. & Suppes, P. (1970). *Information and Inference*. Synthese Library, Reidel. 34

Hochbaum, D.S., ed. (1997). *Approximation algorithms for NP-hard problems*. PWS Publishing Co., Boston, MA, USA. 60, 87

Huang, J. & Efthimiadis, E.N. (2009). Analyzing and evaluating query reformulation strategies in web search logs. In *Proceedings of the 18th ACM Conference on Information and Knowledge Management*, 77–86, ACM, Hong Kong, China. 16

Ingwersen, P. & Järvelin, K. (2005). *The Turn: Integration of Information Seeking and Retrieval in Context (The Information Retrieval Series)*. Springer, Secaucus, NJ, USA. 48

Jain, A., Cucerzan, S. & Azzam, S. (2007). Acronym-expansion recognition and ranking on the Web. In *Proceedings of the 2007 IEEE International Conference on Information Reuse and Integration*, 209–214. 16

Jansen, B.J., Spink, A., Bateman, J. & Saracevic, T. (1998). Real life information retrieval: A study of user queries on the Web. *SIGIR Forum*, **32**, 5–17. 104, 121, 136

Jansen, B.J., Spink, A. & Saracevic, T. (2000). Real life, real users, and real needs: A study and analysis of user queries on the Web. *Information Processing and Management*, **36**, 207–227. 1, 16, 18, 54, 208, 217

Järvelin, K. & Kekäläinen, J. (2002). Cumulated gain-based evaluation of IR techniques. *ACM Transactions on Information Systems*, **20**, 422–446. 51, 79, 80

Joachims, T. (2002). Optimizing search engines using clickthrough data. In *Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 133–142, ACM, Edmonton, AB, Canada. 40, 41, 47, 191

Joachims, T., Granka, L., Pan, B., Hembrooke, H. & Gay, G. (2005). Accurately interpreting clickthrough data as implicit feedback. In *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 154–161, ACM. 48

Jones, R., Rey, B., Madani, O. & Greiner, W. (2006). Generating query substitutions. In *Proceedings of the 15th international conference on World Wide Web*, 387–396, ACM, Edinburgh, UK. 121

Kang, I.H. & Kim, G. (2003). Query type classification for web document retrieval. In *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval*, 64–71, ACM, Toronto, Canada. 149, 150, 151, 196

Kanungo, T. & Orr, D. (2009). Predicting the readability of short web summaries. In *Proceedings of the Second ACM International Conference on Web Search and Data Mining*, 202–211, ACM. 37

Kaplan, A.M. & Haenlein, M. (2010). Users of the world, unite! the challenges and opportunities of social media. *Business Horizons*, **53**, 59–68. 13

Kearns, M. (1988). Thoughts on hypothesis boosting. Unpublished manuscript. 45, 105

Kelly, D. & Teevan, J. (2003). Implicit feedback for inferring user preference: A bibliography. *SIGIR Forum*, **37**, 18–28. 47

Kemeny, J.G. & Snell, J.L. (1960). *Finite Markov Chains*. Springer. 67, 158

Khuller, S., Moss, A. & Naor, J.S. (1999). The budgeted maximum coverage problem. *Information Processing Letters*, **70**, 39–45. 62

Kirkpatrick, S., Gelatt, C.D. & Vecchi, M.P. (1983). Optimization by simulated annealing. *Science*, **220**, 671–680. 107, 176

Kleinberg, J.M. (1998). Authoritative sources in a hyperlinked environment. In *Proceedings of the 9th Annual ACM-SIAM Symposium on Discrete Algorithms*, 668–677, Society for Industrial and Applied Mathematics, San Francisco, CA, USA. 40

Kleinberg, J.M., Kumar, R., Raghavan, P., Rajagopalan, S. & Tomkins, A. (1999). The Web as a graph: Measurements, models, and methods. *Computing and Combinatorics*, 1–17. 10, 40

Kohavi, R. & John, G.H. (1997). Wrappers for feature subset selection. *Artificial Intelligence*, **97**, 273–324. 200

Kohlschütter, C., Fankhauser, P. & Nejdl, W. (2010). Boilerplate detection using shallow text features. In *Proceedings of the 3rd ACM International Conference on Web Search and Data Mining*, 441–450, ACM, New York, NY, USA. 14

Kraaij, W., Westerveld, T. & Hiemstra, D. (2002). The importance of prior probabilities for entry page search. In *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 27–34, Tampere, Finland. 36, 37

Kumaran, G. & Allan, J. (2008). Effective and efficient user interaction for long queries. In *Proceedings of the 31st International ACM SIGIR Conference on Research and Development in Information Retrieval*, 11–18, ACM, Singapore, Singapore. 17

Kumaran, G. & Carvalho, V.R. (2009). Reducing long queries using query quality predictors. In *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 564–571, ACM, Boston, MA, USA. 17

Lafferty, J. & Zhai, C. (2001). Document language models, query models, and risk minimization for information retrieval. In *Proceedings of the 24th Annual*

International ACM SIGIR Conference on Research and Development in Information Retrieval, 111–119, ACM, New Orleans, LA, USA. 29

Lagergren, E. & Over, P. (1998). Comparing interactive information retrieval systems across sites: The TREC-6 Interactive track matrix experiment. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 164–172, ACM, Melbourne, Australia. 75

Lanczos, C. (1964). A precision approximation of the Gamma function. *Journal of the Society for Industrial and Applied Mathematics: Series B, Numerical Analysis*, **1**, 86–96. 35

Laplace, P.S. (1814). *Essai philosophique sur les probabilités*. Courcier, Paris, France. 33

Lavrenko, V. & Croft, W.B. (2001). Relevance based language models. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 120–127, ACM, New Orleans, LA, USA. 17, 29, 50, 69

le Cessie, S. & van Houwelingen, J. (1992). Ridge estimators in logistic regression. *Applied Statistics*, **41**, 191–201. 162

Li, H. (2010). Query understanding in web search: By large scale log data mining and statistical learning. In *Proceedings of the 2nd Workshop on NLP Challenges in the Information Explosion Era*, 1, Beijing, China. 16

Li, H. (2011). *Learning to Rank for Information Retrieval and Natural Language Processing*. Morgan & Claypool. 44

Li, H. & Xu, J. (2012). Machine learning for query-document matching in search. In *Proceedings of the 5th ACM International Conference on Web Search and Data Mining*, 767–768, ACM, Seattle, WA, USA. 19

Li, J., Huffman, S. & Tokuda, A. (2009). Good abandonment in mobile and pc internet search. In *Proceedings of the 32nd Annual International ACM SI-*

GIR Conference on Research and Development in Information Retrieval, 43–50, ACM, Boston, MA, USA. 41, 47

Li, M., Zhu, M., Zhang, Y. & Zhou, M. (2006). Exploring distributional similarity based models for query spelling correction. In *Proceedings of the 21st International Conference on Computational Linguistics*, 1025–1032, ACL, Sydney, Australia. 16

Li, Y., Luk, W.P.R., Ho, K.S.E. & Chung, F.L.K. (2007). Improving weak ad-hoc queries using Wikipedia external corpus. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 797–798, ACM, Amsterdam, The Netherlands. 17

Limsopatham, N., McCreadie, R., Albakour, M.D., Macdonald, C., Santos, R.L.T. & Ounis, I. (2012). University of Glasgow at TREC 2012: experiments with Terrier in Medical Records, Microblog, and Web tracks. In *Proceedings of the 21st Text REtrieval Conference*, Gaithersburg, MD, USA. 6, 221

Liu, T.Y. (2009). Learning to rank for information retrieval. *Foundations and Trends in Information Retrieval*, **3**, 225–331. 42, 44, 105, 124, 129

Luhn, H.P. (1957). A statistical approach to mechanized encoding and searching of literary information. *IBM Journal of Research and Development*, **1**, 309–317. 15, 19

Lv, Y. & Zhai, C. (2009). Positional language models for information retrieval. In *Proceedings of the 32nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 299–306, ACM, Boston, MA, USA. 27

Ma, H., Lyu, M.R. & King, I. (2010). Diversifying query suggestion results. In *Proceedings of the 24th AAAI Conference on Artificial Intelligence*, AAAI Press, Atlanta, GA, USA. 216

Macdonald, C. & Ounis, I. (2010). Global statistics in proximity weighting models. In *Proceedings of the SIGIR 2010 Web N-gram Workshop*, 30–36, ACM, Geneva, Switzerland. 35

Macdonald, C., He, B., Plachouras, V. & Ounis, I. (2005). University of Glasgow at TREC 2005: experiments in Terabyte and Enterprise tracks with Terrier. In *Proceedings of the 14th Text REtrieval Conference*, Gaithersburg, MD, USA. 190

Macdonald, C., Plachouras, V., He, B., Lioma, C. & Ounis, I. (2006). University of Glasgow at WebCLEF 2005: experiments in per-field normalisation and language specific stemming. In *Proceedings of the 6th Cross-Language Evaluation Forum*, 898–907, Springer-Verlag, Vienna, Austria. 158

Macdonald, C., McCreadie, R., Santos, R.L.T. & Ounis, I. (2012a). From puppy to maturity: Experiences in developing Terrier. In *Proceedings of the SIGIR 2012 Workshop on Open Source Information Retrieval*, Portland, OR, USA. 102, 134, 175, 199

Macdonald, C., Santos, R.L.T. & Ounis, I. (2012b). On the usefulness of query features for learning to rank. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management*, 2559–2562, ACM, Maui, HI, USA. 216

Macdonald, C., Tonellotto, N. & Ounis, I. (2012c). Learning to predict response times for online query scheduling. In *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval*, ACM. 17

Macdonald, C., Santos, R.L.T. & Ounis, I. (2013). The whens and hows of learning to rank for web search. *Information Retrieval*. 43, 105, 130

Mackay, D.J.C. & Peto, L. (1994). A hierarchical Dirichlet language model. *Natural Language Engineering*, **1**, 1–19. 30

Manning, C.D. & Schütze, H. (1999). *Foundations of statistical natural language processing*. MIT Press. 25

Manning, C.D., Raghavan, P. & Schütze, H. (2008). *Introduction to Information Retrieval*. Cambridge University Press. 11, 15, 30, 109

Markov, A.A. (1913). An example of statistical investigation in the text of Eugene Onegin illustrating coupling of tests in chains. *Proceedings of the Academy of Sciences of St. Petersburg*, **7**, 153–162. 25

Markov, A.A. (1954). *Theory of Algorithms*. Academy of Sciences of the USSR. 26

Markowitz, H. (1952). Portfolio selection. *The Journal of Finance*, **7**, 77–91. 66

Maron, M.E. & Kuhns, J.L. (1960). On relevance, probabilistic indexing and information retrieval. *Journal of the ACM*, **7**, 216–244. 21, 54

McCreadie, R., Macdonald, C., Ounis, I., Peng, J. & Santos, R.L.T. (2009). University of Glasgow at TREC 2009: Experiments with Terrier—Blog, Entity, Million Query, Relevance Feedback, and Web tracks. In *Proceedings of the 18th Text REtrieval Conference*, Gaithersburg, MD, USA. 6, 34

McCreadie, R., Macdonald, C., Santos, R.L.T. & Ounis, I. (2011). University of Glasgow at TREC 2011: Experiments with Terrier in Crowdsourcing, Microblog, and Web tracks. In *Proceedings of the 20th Text REtrieval Conference*, Gaithersburg, MD, USA. 6, 34

Mei, Q., Zhou, D. & Church, K. (2008). Query suggestion using hitting time. In *Proceedings of the 17th ACM Conference on Information and Knowledge Management*, 469–478, ACM, Napa Valley, CA, USA. 122

Metzler, D. & Croft, W.B. (2005). A Markov random field model for term dependencies. In *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 472–479. 28, 35, 50

Metzler, D.A. (2007). Automatic feature selection in the markov random field model for information retrieval. In *Proceedings of the 16th Conference on Information and Knowledge Management*, 253–262, ACM, Lisbon, Portugal. 45, 158

Miller, D.R.H., Leek, T. & Schwartz, R.M. (1999). A hidden Markov model information retrieval system. In *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 214–221, ACM, Berkeley, CA, USA. 26

Mizzaro, S. (1997). Relevance: The whole history. *Journal of the American Society for Information Science*, **48**, 810–832. 47

Moffat, A. & Zobel, J. (2008). Rank-biased precision for measurement of retrieval effectiveness. *ACM Transactions on Information Systems*, **27**, 1–27. 52, 81

Mowshowitz, A. & Kawaguchi, A. (2002). Assessing bias in search engines. *Information Processing and Management*, **38**, 141–156. 85

Murdock, V. & Lalmas, M. (2008). Workshop on aggregated search. *SIGIR Forum*, **42**, 80–83. 218

Najork, M. & Wiener, J.L. (2001). Breadth-first crawling yields high-quality pages. In *Proceedings of the 10th International Conference on World Wide Web*, 114–118, ACM, Hong Kong, Hong Kong. 12

Nemhauser, G.L., Wolsey, L.A. & Fisher, M.L. (1978). An analysis of approximations for maximizing submodular set functions—I. *Mathematical Programming*, **14**, 265–294. 62, 86

Newman, M.E.J. (2003). The structure and function of complex networks. *SIAM Review*, **45**, 167–256. 40

Ntoulas, A., Cho, J. & Olston, C. (2004). What's new on the Web? The evolution of the Web from a search engine perspective. In *Proceedings of the 13th International Conference on World Wide Web*, 1–12, ACM, New York, NY, USA. 13

Ntoulas, A., Najork, M., Manasse, M. & Fetterly, D. (2006). Detecting spam web pages through content analysis. In *Proceedings of the 15th International Conference on World Wide Web*, 83–92, ACM. 39

Omohundro, S.M. (1989). Five balltree construction algorithms. Tech. Rep. TR-89-063, International Computer Science Institute. 194

Ounis, I., Amati, G., Plachouras, V., He, B., Macdonald, C. & Lioma, C. (2006). Terrier: A high performance and scalable information retrieval platform. In *Proceedings of ACM SIGIR 2006 Workshop on Open Source Information Retrieval*. 102

Over, P. (1997). Trec-6 interactive report. In *Proceedings of the 6th Text REtrieval Conference*, 73–81, Gaithersburg, MD, USA. 75

Over, P. (1998). Trec-7 interactive track report. In *Proceedings of the 7th Text REtrieval Conference*, 33–39, Gaithersburg, MD, USA. 75

Page, L., Brin, S., Motwani, R. & Winograd, T. (1999). The PageRank citation ranking: Bringing order to the Web. Tech. Rep. 1999-66, Stanford InfoLab. 40

Pant, G., Srinivasan, P. & Menczer, F. (2004). Crawling the Web. In M. Levene & A. Poulovassilis, eds., *Web dynamics: Adapting to change in content, size, topology and use*, Springer. 11

Paramita, M.L., Tang, J. & Sanderson, M. (2009). Generic and spatial approaches to image search results diversification. In *Proceedings of the 31st European Conference on IR Research on Advances in Information Retrieval*, 603–610, Springer, Toulouse, France. 217

Pearson, K. (1901). On lines and planes of closest fit to systems of points in space. *Philosophical Magazine*, **2**, 559–572. 162, 200

Peng, D. & Dabek, F. (2010). Large-scale incremental processing using distributed transactions and notifications. In *Proceedings of the 9th USENIX Conference on Operating Systems Design and Implementation*, 1–15, USENIX Association, Vancouver, BC, Canada. 14

Peng, F., Ahmed, N., Li, X. & Lu, Y. (2007a). Context sensitive stemming for web search. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 639–646, ACM, Amsterdam, The Netherlands. 15, 16

Peng, J. (2010). *Learning to select for information retrieval*. Ph.D. thesis, University of Glasgow, Glasgow, UK. 190

Peng, J. & Ounis, I. (2009). Selective application of query-independent features in web information retrieval. In *Proceedings of the 31st European Conference on IR Research on Advances in Information Retrieval*, 375–387, Springer, Toulouse, France. 190

Peng, J., Macdonald, C., He, B., Plachouras, V. & Ounis, I. (2007b). Incorporating term dependency in the DFR framework. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 843–844, ACM, Amsterdam, The Netherlands. 35, 50

Peng, J., Macdonald, C. & Ounis, I. (2010). Learning to select a ranking function. In *Proceedings of the 31st European Conference on IR Research on Advances in Information Retrieval*, 114–126, Springer, Milton Keynes, UK. 149, 150, 151, 190

Plachouras, V. (2006). *Selective web information retrieval*. Ph.D. thesis, University of Glasgow, Glasgow, UK. 190

Plachouras, V. & Ounis, I. (2004). Usefulness of hyperlink structure for query-biased topic distillation. In *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 448–455, ACM, Sheffield, UK. 33, 190

Plachouras, V., Ounis, I. & Amati, G. (2005). The static absorbing model for the Web. *Journal of Web Engineering*, **4**, 165–186. 40, 158

Plakhov, A. (2011). Entity-oriented search result diversification. In *Proceedings of the 1st International Workshop on Entity-Oriented Search*, Beijing, China. 215

Platt, J.C. (1998). Sequential minimal optimization: a fast algorithm for training support vector machines. Tech. Rep. MSR-TR-98-14, Microsoft Research. 162

Poisson, S. (1837). *Recherches sur la probabilité des jugements en matière criminelle et en matière civile: Précédées des règles générales du calcul des probabilités*. Bachelier. 23, 33

Ponte, J.M. & Croft, W.B. (1998). A language modeling approach to information retrieval. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 275–281, ACM, Melbourne, Australia. 26, 27

Popper, K.R. (1934). *The Logic of Scientific Discovery*. Hutchinson, London, UK. 34

Porter, M.F. (1980). An algorithm for suffix stripping. *Program*, **14**, 130–137. 15, 16, 102

Qin, T., Liu, T.Y., Xu, J. & Li, H. (2010). LETOR: A benchmark collection for research on learning to rank for information retrieval. *Information Retrieval*, **13**, 346–374. 105, 136

Radlinski, F. & Dumais, S. (2006). Improving personalized web search using result diversification. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 691–692, ACM, Seattle, WA, USA. 64, 69, 85, 96, 107, 109, 110

Radlinski, F., Kleinberg, R. & Joachims, T. (2008a). Learning diverse rankings with multi-armed bandits. In *Proceedings of the 25th International Conference on Machine Learning*, 784–791, ACM, Helsinki, Finland. 63, 64, 70, 72, 220

Radlinski, F., Kurup, M. & Joachims, T. (2008b). How does clickthrough data reflect retrieval quality? In *Proceedings of the 17th ACM International Conference on Information and Knowledge Management*, 43–52, ACM. 48

Radlinski, F., Szummer, M. & Craswell, N. (2010a). Inferring query intent from reformulations and clicks. In *Proceedings of the 19th International Conference on World Wide Web*, 1171–1172, Raleigh, NC, USA. 75, 76, 216

Radlinski, F., Szummer, M. & Craswell, N. (2010b). Metrics for assessing sets of subtopics. In *Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 853–854, ACM, Geneva, Switzerland. 75, 76

Rafiei, D., Bharat, K. & Shukla, A. (2010). Diversifying web search results. In *Proceedings of the 19th International Conference on World Wide Web*, 781–790, Raleigh, NC, USA. 63, 64, 66, 85, 169, 217

Raghavan, S. & Garcia-Molina, H. (2000). Crawling the hidden Web. Tech. Rep. 2000-36, Stanford InfoLab. 12

Raman, K., Shivaswamy, P. & Joachims, T. (2012). Online learning to diversify from implicit feedback. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ACM, Beijing, China. 64, 72, 220

Richardson, M., Dominowska, E. & Ragno, R. (2007). Predicting clicks: estimating the click-through rate for new ads. In *Proceedings of the 16th International Conference on World Wide Web*, 521–530, ACM, Banff, AB, Canada. 41

Risvik, K.M., Mikolajewski, T. & Boros, P. (2003). Query segmentation for web search. In *Proceedings of the 12th International Conference on World Wide Web*, ACM, Budapest, Hungary. 15, 17

Robertson, S. (2008). On the optimisation of evaluation metrics. In *Proceedings of SIGIR 2007 Workshop on Learning to Rank for Information Retrieval*, ACM, Singapore, Singapore. 51, 136

Robertson, S. & Zaragoza, H. (2009). The probabilistic relevance framework: BM25 and beyond. *Foundations and Trends in Information Retrieval*, **3**, 333–389. 22, 23, 24, 25, 31

Robertson, S., Zaragoza, H. & Taylor, M. (2004). Simple bm25 extension to multiple weighted fields. In *Proceedings of the 13th ACM International Conference on Information and Knowledge Management*, 42–49, ACM, Washington, DC, USA. 21

Robertson, S.E. (1977). The probability ranking principle in IR. *Journal of Documentation*, **33**, 294–304. 2, 21, 54, 57, 67

Robertson, S.E. & Spärck Jones, K. (1976). Relevance weighting of search terms. *Journal of the American Society for Information Science*, **27**, 129–146. 21, 22

Robertson, S.E. & Walker, S. (1994). Some simple effective approximations to the 2-Poisson model for probabilistic weighted retrieval. In *Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 232–241, Springer, Dublin, Ireland. 23, 24

Robertson, S.E., van Rijsbergen, C.J. & Porter, M.F. (1981). Probabilistic models of indexing and searching. In *Proceedings of the 3rd Annual ACM Conference on Research and Development in Information Retrieval*, 35–56, Butterworth & Co. 21, 23

Robertson, S.E., Walker, S., Jones, S., Hancock-Beaulieu, M. & Gatford, M. (1993). Okapi at TREC-2. In *Proceedings of the 2nd Text REtrieval Conference*, Gaithersburg, MD, USA. 23, 24, 25

Robertson, S.E., Walker, S., Jones, S., Hancock-Beaulieu, M. & Gatford, M. (1994). Okapi at TREC-3. In *Proceedings of the 3rd Text REtrieval Conference*, Gaithersburg, MD, USA. 21, 25

Rocchio, J.J. (1971). Relevance feedback in information retrieval. In G. Salton, ed., *The SMART Retrieval System: Experiments in Automatic Document Processing*, 313–323, Prentice Hall. 17, 29, 50

Rose, D.E. & Levinson, D. (2004). Understanding user goals in web search. In *Proceedings of the 13th International Conference on World Wide Web*, 13–19, ACM, New York, NY, USA. 47, 78, 149, 150

Rowe, B.R., Wood, D.W., Link, A.N. & Simoni, D.A. (2010). Economic impact assessment of NIST's Text REtrieval Conference (TREC) program. Tech. Rep. 0211875, RTI International. 48

Sakai, T. (2012). Evaluation with informational and navigational intents. In *Proceedings of the 21st International Conference on World Wide Web*, 499–508, ACM, Lyon, France. 81

Sakai, T. & Song, R. (2012). Diversified search evaluation: Lessons from the NTCIR-9 Intent task. *Information Retrieval*. 77

Sakai, T., Craswell, N., Song, R., Robertson, S., Dou, Z. & Lin, C.Y. (2010). Simple evaluation metrics for diversified search results. In *Proceedings of the 3rd International Workshop on Evaluating Information Access*, 42–50, NII, Tokyo, Japan. 78, 79

Salomon, D. (2007). *Data Compression: The Complete Reference*. Springer. 39

Salton, G. & Buckley, C. (1988). Term-weighting approaches in automatic text retrieval. *Information Processing and Management*, **24**, 513–523. 19, 20

Salton, G., Wong, A. & Yang, C.S. (1975). A vector space model for automatic indexing. *Communications of the ACM*, **18**, 613–620. 20

Samuelson, P.A. & Nordhaus, W.D. (2001). *Microeconomics*. McGraw-Hill. 60

Sanderson, M. (2008). Ambiguous queries: Test collections need more sense. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 499–506, ACM, Singapore, Singapore. 54, 55, 74, 155, 196

Sanderson, M. (2010). Test collection based evaluation of information retrieval systems. *Foundations and Trends in Information Retrieval*, **4**, 247–375. 47, 50

Sanderson, M. & Zobel, J. (2005). Information retrieval system evaluation: effort, sensitivity, and reliability. In *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 162–169, ACM, Salvador, Brazil. 104

Sanderson, M., Paramita, M.L., Clough, P. & Kanoulas, E. (2010). Do user preferences and evaluation measures line up? In *Proceedings of the 33rd Interna-*

tional ACM SIGIR Conference on Research and Development in Information Retrieval, 555–562, ACM, Geneva, Switzerland. 82

Sanner, S., Guo, S., Graepel, T., Kharazmi, S. & Karimi, S. (2011). Diverse retrieval via greedy optimization of expected 1-call@k in a latent subtopic relevance model. In *Proceedings of the 20th ACM International Conference on Information and Knowledge Management*, 1977–1980, ACM, Glasgow, UK. 88

Santos, R.L.T. & Ounis, I. (2011). Diversifying for multiple information needs. In *Proceedings of the 1st International Workshop on Diversity in Document Retrieval*, 37–41, Dublin, Ireland. 5

Santos, R.L.T., Macdonald, C. & Ounis, I. (2010a). Exploiting query reformulations for web search result diversification. In *Proceedings of the 19th International Conference on World Wide Web*, 881–890, ACM, Raleigh, NC, USA. 5, 63, 64, 217

Santos, R.L.T., Macdonald, C. & Ounis, I. (2010b). Selectively diversifying web search results. In *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*, 1179–1188, ACM, Toronto, Canada. 5

Santos, R.L.T., Macdonald, C. & Ounis, I. (2010c). Voting for related entities. In *Proceedings of the 9th International Conference on Adaptivity, Personalization and Fusion of Heterogeneous Information (Recherche d'Information et ses Applications)*, CID, Paris, France. 155

Santos, R.L.T., McCreadie, R., Macdonald, C. & Ounis, I. (2010d). University of Glasgow at TREC 2010: Experiments with Terrier in Blog and Web tracks. In *Proceedings of the 19th Text REtrieval Conference*, Gaithersburg, MD, USA. 6, 34

Santos, R.L.T., Peng, J., Macdonald, C. & Ounis, I. (2010e). Explicit search result diversification through sub-queries. In *Proceedings of the 31st European Conference on IR Research on Advances in Information Retrieval*, 87–99, Springer, Milton Keynes, UK. 5, 63, 64

Santos, R.L.T., Macdonald, C. & Ounis, I. (2011a). Aggregated search result diversification. In *Proceedings of the 3rd International Conference on the Theory of Information Retrieval*, 250–261, Springer, Bertinoro, Italy. 6, 217, 218

Santos, R.L.T., Macdonald, C. & Ounis, I. (2011b). Effectiveness beyond the first crawl tier. In *Proceedings of the 20th ACM International Conference on Information and Knowledge Management*, 1937–1940, ACM, Glasgow, UK. 102

Santos, R.L.T., Macdonald, C. & Ounis, I. (2011c). How diverse are web search results? In *Proceedings of the 34th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1187–1188, ACM, Beijing, China. 135

Santos, R.L.T., Macdonald, C. & Ounis, I. (2011d). Intent-aware search result diversification. In *Proceedings of the 34th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 595–604, ACM, Beijing, China. 5, 45

Santos, R.L.T., Macdonald, C. & Ounis, I. (2011e). On the suitability of diversity metrics for learning-to-rank for diversity. In *Proceedings of the 34th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1185–1186, ACM, Beijing, China. 6

Santos, R.L.T., Macdonald, C. & Ounis, I. (2011f). University of Glasgow at the NTCIR-9 Intent task. In *Proceedings of the 9th NTCIR Workshop Meeting on Evaluation of Information Access Technologies*, Tokyo, Japan. 6, 77

Santos, R.L.T., McCreadie, R. & Plachouras, V. (2011g). Large-scale information retrieval experimentation with Terrier. In *Proceedings of the 20th ACM International Conference on Information and Knowledge Management*, 2601–2602, ACM, Glasgow, UK. 102

Santos, R.L.T., Macdonald, C., McCreadie, R., Ounis, I. & Soboroff, I. (2012a). Information retrieval on the blogosphere. *Foundations and Trends in Information Retrieval*, **6**, 1–125. 217

Santos, R.L.T., Macdonald, C. & Ounis, I. (2012b). On the role of novelty for search result diversification. *Information Retrieval*, **15**, 478–502. 5, 63, 64

Santos, R.L.T., Macdonald, C. & Ounis, I. (2013). Learning to rank query suggestions for adhoc and diversity search. *Information Retrieval*. 5

Saracevic, T. (1995). Evaluation of evaluation in information retrieval. In *Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 138–146, ACM, Seattle, WA, USA. 48

Schapire, R.E. (1990). The strength of weak learnability. *Machine Learning*, **5**, 197–227. 45, 105

Searcóid, M. (2006). *Metric Spaces*. Springer Undergraduate Mathematics Series, Springer. 73

Shahabi, C. & Chen, Y.S. (2003). Web information personalization: challenges and approaches. In *Proceedings of the 3rd International Workshop on Databases in Networked Information Systems*, 5–15, Aizu, Japan. 219

Shannon, C.E. (1948). A mathematical theory of communication. *Bell System Technical Journal*, **27**, 379–423, 623–656. 25

Sheldon, D., Shokouhi, M., Szummer, M. & Craswell, N. (2011). Lambdamerge: Merging the results of query reformulations. In *Proceedings of the 4th ACM International Conference on Web Search and Data Mining*, 795–804, ACM, Hong Kong, China. 146

Shen, D., Sun, J.T., Yang, Q. & Chen, Z. (2006). Building bridges for web query classification. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 131–138, ACM, Seattle, WA, USA. 17

Silverstein, C., Marais, H., Henzinger, M. & Moricz, M. (1999). Analysis of a very large web search engine query log. *SIGIR Forum*, **33**, 6–12. 17, 18

Silvestri, F. (2010). Mining query logs: Turning search usage data into knowledge. *Foundations and Trends in Information Retrieval*, **4**, 1–174. 121, 122, 124, 131, 135, 197

Singhal, A., Buckley, C. & Mitra, M. (1996). Pivoted document length normalization. In *Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 21–29, ACM, Zurich, Switzerland. 19

Slivkins, A., Radlinski, F. & Gollapudi, S. (2010). Learning optimally diverse rankings over large document collections. In *Proceedings of the 27th Annual International Conference on Machine Learning*, 983–990, Omnipress, Haifa, Israel. 64, 72, 220

Smucker, M.D., Allan, J. & Carterette, B. (2007). A comparison of statistical significance tests for information retrieval evaluation. In *Proceedings of the 16th ACM Conference on Information and Knowledge Management*, 623–632, ACM, Lisbon, Portugal. 104

Song, F. & Croft, W.B. (1999). A general language model for information retrieval. In *Proceedings of the 8th International Conference on Information and Knowledge Management*, 316–321, ACM. 27

Song, R., Luo, Z., Nie, J.Y., Yu, Y. & Hon, H.W. (2009). Identification of ambiguous queries in web search. *Information Processing and Management*, **45**, 216–229. 1, 16, 54, 55, 132, 157, 198, 208

Song, R., Zhang, M., Sakai, T., Kato, M.P., Liu, Y., Sugimoto, M., Wang, Q. & Orii, N. (2011a). Overview of the NTCIR-9 Intent task. In *Proceedings of the 9th NTCIR Workshop Meeting on Evaluation of Information Access Technologies*, Tokyo, Japan. 6, 77

Song, Y., Zhou, D. & wei He, L. (2011b). Post-ranking query suggestion by diversifying search results. In *Proceedings of the 34th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 815–824, ACM, Beijing, China. 123, 216

Spärck Jones, K. (1972). A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation*, **28**, 11–21. 19

Spärck Jones, K. & van Rijsbergen, C. (1975). Report on the need for and provision of an ideal information retrieval test collection. Tech. rep., Computer Laboratory, University of Cambridge. 49

Spärck-Jones, K., Robertson, S.E. & Sanderson, M. (2007). Ambiguous requests: Implications for retrieval tests, systems and theories. *SIGIR Forum*, **41**, 8–17. 1, 54, 58, 74, 84, 87, 132

Srikanth, M. & Srihari, R. (2002). Biterm language models for document retrieval. In *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 425–426, ACM, Tampere, Finland. 27

Stamou, S. & Efthimiadis, E.N. (2010). Interpreting user inactivity on search results. In *Proceedings of the 32nd European Conference on IR Research on Advances in Information Retrieval*, 100–113, Springer, Milton Keynes, UK. 41, 47

Szpektor, I., Gionis, A. & Maarek, Y. (2011). Improving recommendation for long-tail queries via templates. In *Proceedings of the 20th international conference on World wide web*, 47–56, ACM, Hyderabad, India. 122

Tan, B. & Peng, F. (2008). Unsupervised query segmentation using generative language models and wikipedia. In *Proceedings of the 17th International Conference on World Wide Web*, 347–356, ACM. 15

Teevan, J., Dumais, S.T. & Horvitz, E. (2007). Characterizing the value of personalizing search. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 757–758, ACM, Amsterdam, The Netherlands. 48, 51

Thelwall, M. & Stuart, D. (2006). Web crawling ethics revisited: Cost, privacy, and denial of service. *Journal of the American Society for Information Science and Technology*, **57**, 1771–1779. 11

Thomas, P. & Hawking, D. (2006). Evaluation by comparing result sets in context. In *Proceedings of the 15th ACM International Conference on Information and Knowledge Management*, 94–101, ACM. 48

Tombros, A. & Sanderson, M. (1998). Advantages of query biased summaries in information retrieval. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2–10, ACM, Melbourne, Australia. 37

Tsochantaridis, I., Joachims, T., Hofmann, T. & Altun, Y. (2005). Large margin methods for structured and interdependent output variables. *Journal of Machine Learning Research*, **6**, 1453–1484. 72

Turpin, A. & Scholer, F. (2006). User performance versus precision measures for simple search tasks. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 11–18, ACM, Seattle, WA, USA. 181

Turtle, H. & Flood, J. (1995). Query evaluation: Strategies and optimizations. *Information Processing and Management*, **31**, 831–850. 17

Turtle, H.R. & Croft, W.B. (1996). Uncertainty in information retrieval systems. In *Uncertainty Management in Information Systems*, 189–224, Kluwer Academic Publishers, Norwell, MA, USA. 57

Vallet, D. & Castells, P. (2012). Personalized diversification of search results. In *Proceedings of the 35th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 841–850, ACM, Portland, OR, USA. 219

van Leuken, R.H., Garcia, L., Olivares, X. & van Zwol, R. (2009). Visual diversification of image search results. In *Proceedings of the 18th International Conference on World Wide Web*, 341–350, ACM, Madrid, Spain. 217

Vee, E., Srivastava, U., Shanmugasundaram, J., Bhat, P. & Yahia, S.A. (2008). Efficient computation of diverse query results. In *Proceedings of the 24th Inter-*

national Conference on Data Engineering, 228–236, IEEE Computer Society, Cancún, Mexico. 217

Vieira, K., da Silva, A.S., Pinto, N., de Moura, E.S., Cavalcanti, J.a.M.B. & Freire, J. (2006). A fast and robust method for web page template detection and removal. In *Proceedings of the 15th ACM International Conference on Information and Knowledge Management*, 258–267, ACM, Arlington, VA, USA. 14

Vohra, R.V. & Hall, N.G. (1993). A probabilistic analysis of the maximal covering location problem. *Discrete Applied Mathematics*, **43**, 175–183. 60

von Neumann, J. & Morgenstern, O. (1944). *Theory of Games and Economic Behavior*. Princeton University Press. 57

Voorhees, E.M. (2007). TREC: Continuing information retrieval's tradition of experimentation. *Communications of the ACM*, **50**, 51–54. 48, 49

Voorhees, E.M. & Harman, D. (1997). Overview of the 6th Text REtrieval Conference. In *Proceedings of the 6th Text REtrieval Conference*, Gaithersburg, MD, USA. 75

Voorhees, E.M. & Harman, D. (1998). Overview of the 7th Text REtrieval Conference. In *Proceedings of the 7th Text REtrieval Conference*, Gaithersburg, MD, USA. 75

Voorhees, E.M. & Harman, D. (1999). Overview of the 8th Text REtrieval Conference. In *Proceedings of the 8th Text REtrieval Conference*, Gaithersburg, MD, USA. 75

Voorhees, E.M. & Harman, D.K. (2005). *TREC: Experiment and Evaluation in Information Retrieval*. Digital Libraries and Electronic Publishing, MIT Press. 48, 49

Wang, J. & Zhu, J. (2009). Portfolio theory of information retrieval. In *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 115–122, ACM, Boston, MA, USA. 63, 64, 66, 85, 169, 170, 171, 172, 217

Wang, X. & Zhai, C. (2008). Mining term association patterns from search logs for effective query reformulation. In *Proceedings of the 17th ACM Conference on Information and Knowledge Management*, 479–488, ACM, Napa Valley, CA, USA. 121

Wang, Y. & Agichtein, E. (2010). Query ambiguity revisited: Clickthrough measures for distinguishing informational and ambiguous queries. In *Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics—Human Language Technologies*, 361–364. 197

Welch, M.J., Cho, J. & Olston, C. (2011). Search result diversity for informational queries. In *Proceedings of the 20th International Conference on World Wide Web*, 237–246, ACM, Hyderabad, India. 196

Witten, I.H. & Frank, E. (2005). *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann, San Francisco, CA, USA, 2nd edn. 153, 162, 170, 193, 194, 200

Witten, I.H., Moffat, A. & Bell, T.C. (1999). *Managing Gigabytes: Compressing and Indexing Documents and Images*. Morgan Kaufmann, San Francisco, CA, USA, 2nd edn. 14, 44

Woodbury, M.A. (1950). Inverting modified matrices. Tech. Rep. MR38136, Statistical Research Group, Princeton University, Princeton, NJ, USA. 67

Wu, Q., Burges, C.J.C., Svore, K.M. & Gao, J. (2008). Ranking, boosting, and model adaptation. Tech. Rep. MSR-TR-2008-109, Microsoft Research. 45, 105

Xu, Y., Jones, G.J. & Wang, B. (2009). Query dependent pseudo-relevance feedback based on wikipedia. In *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 59–66, ACM. 17

Yom-Tov, E., Fine, S., Carmel, D. & Darlow, A. (2005). Learning to estimate query difficulty: Including applications to missing content detection and distributed information retrieval. In *Proceedings of the 28th Annual International*

ACM SIGIR Conference on Research and Development in Information Retrieval, 512–519, ACM, Salvador, Brazil. 190

Yue, Y. & Joachims, T. (2008). Predicting diverse subsets using structural svms. In *Proceedings of the 25th International Conference on Machine Learning*, 1224–1231, ACM, Helsinki, Finland. 64, 71, 72, 220

Zaragoza, H., Craswell, N., Taylor, M.J., Saria, S. & Robertson, S.E. (2004). Microsoft Cambridge at TREC 13: Web and Hard tracks. In *Proceedings of the 13th Text REtrieval Conference*, Gaithersburg, MD, USA. 15, 21, 125, 158

Zeng, H.J., He, Q.C., Chen, Z., Ma, W.Y. & Ma, J. (2004). Learning to cluster web search results. In *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval*, 210–217, ACM, Sheffield, United Kingdom. 215

Zhai, C. (2008). Statistical language models for information retrieval: A critical review. *Foundations and Trends in Information Retrieval*, **2**, 137–213. 26, 29, 30

Zhai, C. (2011). Axiomatic analysis and optimization of information retrieval models. In *Proceedings of the 3rd International Conference on Advances in Information Retrieval Theory*, 1, Springer-Verlag, Bertinoro, Italy. 42

Zhai, C. & Lafferty, J. (2001). Model-based feedback in the language modeling approach to information retrieval. In *Proceedings of the 10th International Conference on Information and Knowledge Management*, 403–410, ACM, Atlanta, GA, USA. 17, 28, 29, 50

Zhai, C. & Lafferty, J. (2004). A study of smoothing methods for language models applied to information retrieval. *ACM Transactions on Information Systems*, **22**, 179–214. 29, 30

Zhai, C. & Lafferty, J. (2006). A risk minimization framework for information retrieval. *Information Processing and Management*, **42**, 31–55. 65

Zhai, C., Cohen, W.W. & Lafferty, J. (2003). Beyond independent relevance: Methods and evaluation metrics for subtopic retrieval. In *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval*, 10–17, ACM, Toronto, Canada. 63, 64, 65, 78, 85, 170, 184, 217

Zhang, H.P., Yu, H.K., Xiong, D.Y. & Liu, Q. (2003). HHMM-based Chinese lexical analyzer ICTCLAS. In *Proceedings of the 2nd SIGHAN Workshop on Chinese Language Processing – Volume 17*, 184–187, ACL, Sapporo, Japan. 15

Zhang, Z. & Nasraoui, O. (2006). Mining search engine query logs for query recommendation. In *Proceedings of the 15th international conference on World Wide Web*, 1039–1040, ACM, Edinburgh, UK. 121

Zhao, J. & Yun, Y. (2009). A proximity language model for information retrieval. In *Proceedings of the 32nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 291–298, ACM, Boston, MA, USA. 27

Zheng, W., Fang, H. & Yao, C. (2012). Exploiting concept hierarchy for result diversification. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management*, 1844–1848, ACM, Maui, HI, USA. 215

Zhou, Y. & Croft, W.B. (2005). Document quality models for web ad hoc retrieval. In *Proceedings of the 14th ACM International Conference on Information and Knowledge Management*, 331–332, ACM, Bremen, Germany. 38

Zhou, Y. & Croft, W.B. (2007). Query performance prediction in web search environments. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 543–550, ACM, Amsterdam, The Netherlands. 157, 198

Zhu, X., Goldberg, A.B., Gael, J.V. & Andrzejewski, D. (2007). Improving diversity in ranking using absorbing randomwalks. In *Proceedings of the Annual*

Conference of the North American Chapter of the Association for Computational Linguistics—Human Language Technologies, 97–104, ACL, Rochester, NY, USA. 64, 67

Zobel, J. & Moffat, A. (2006). Inverted files for text search engines. *ACM Computing Surveys*, **38**. 15

Zuccon, G. & Azzopardi, L. (2010). Using the quantum probability ranking principle to rank interdependent documents. In *Proceedings of the 32nd European Conference on IR Research on Advances in Information Retrieval*, 357–369, Springer, Milton Keynes, UK. 64, 67