Exploiting Query Reformulations for Web Search Result Diversification

Rodrygo L. T. Santos
rodrygo@dcs.gla.ac.uk

Craig Macdonald
craigm@dcs.gla.ac.uk

Iadh Ounis
ounis@dcs.gla.ac.uk

Department of Computing Science
University of Glasgow
G12 8QQ Glasgow, UK

ABSTRACT

When a Web user's underlying information need is not clearly specified from the initial query, an effective approach is to diversify the results retrieved for this query. In this paper, we introduce a novel probabilistic framework for Web search result diversification, which explicitly accounts for the various aspects associated to an underspecified query. In particular, we diversify a document ranking by estimating how well a given document satisfies each uncovered aspect and the extent to which different aspects are satisfied by the ranking as a whole. We thoroughly evaluate our framework in the context of the diversity task of the TREC 2009 Web track. Moreover, we exploit query reformulations provided by three major Web search engines (WSEs) as a means to uncover different query aspects. The results attest the effectiveness of our framework when compared to state-of-the-art diversification approaches in the literature. Additionally, by simulating an upper-bound query reformulation mechanism from official TREC data, we draw useful insights regarding the effectiveness of the query reformulations generated by the different WSEs in promoting diversity.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*retrieval models*

General Terms

Algorithms, Experimentation, Performance

Keywords

Web Search, Relevance, Diversity

1. INTRODUCTION

As inherently underspecified representations of more complex information needs, queries submitted to a Web search engine are often ambiguous [29]. Such ambiguity is manifested in different ways. For instance, a query may not express a clearly defined sense (e.g., 'java'), or it can represent a genuine need for a broader coverage of a clearly defined sense (e.g., 'java programming language'). In the first case, the query is open to different *interpretations* (e.g., programming language, coffee, island), whereas in the second case, the user might be interested in different *aspects* underlying the query (e.g., development kit download, courses, books, language specifications, tutorials) [11].

Whatever the level of ambiguity associated to a query, a search engine has to tackle it. The simplest approach could be to completely ignore any sort of ambiguity and treat the query as representing a single, well defined information need. This could result in satisfying the user's need only by chance. A different approach could be to infer the most plausible meaning underlying the query (e.g., the most popular), and to focus the retrieval process on results satisfying that particular meaning. This, in turn, could represent a risky choice, as a wrong guess could leave the user unsatisfied. Another alternative could be to explicitly ask the user for feedback on the correct meaning underlying the query. This has been one of the approaches taken by many Web search engines, in which a user is presented with different reformulations of the original query. However, one cannot expect that the users will always be willing to tell the search engine what they mean exactly [16] and, even if they would, their underlying need might still be for multiple aspects related to the specified meaning. In such a scenario, where 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 results retrieved for this query, in the hope that at least one of them will satisfy the user [1].

Diversifying search results usually involves a departure from the independent document relevance assumption underlying the well-known probability ranking principle in information retrieval (IR) [12, 24]. Indeed, it is arguable whether users will still find a given document relevant to their information need once other documents already satisfying this need have been observed. Therefore, a search engine should consider not only the relevance of each individual document, but also how relevant the document is in light of the other retrieved documents [13]. By doing so, the retrieved documents should provide the maximum coverage and minimum redundancy with respect to the possible aspects underlying a query [10]. Ideally, the covered aspects should also reflect their relative importance, as perceived by the user population [1]. In its general form, this is an NP-hard problem [1, 6]. Most previous approaches to this problem are based on a greedy approximation algorithm, inspired by the notion of maximal marginal relevance [5]. In common, they seek to promote diversity by directly 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

WWW 2010, April 26–30, 2010, Raleigh, North Carolina, USA.
ACM 978-1-60558-799-8/10/04.

documents already selected. Therefore, these approaches *implicitly* assume that similar documents will cover similar interpretations or aspects underlying the query, and should hence be demoted, in order to achieve a diversified ranking.

Alternatively, the broad topic underlying an ambiguous or underspecified query can be usually decomposed into its constituent sub-topics. Hence, we can *explicitly* account for different aspects of the query, in order to produce a diverse ranking of results. In this paper, we introduce a novel framework for search result diversification that exploits this intuition. In particular, our framework uncovers different aspects underlying the original query in the form of *sub-queries*, and estimates the relevance of the retrieved documents to each identified sub-query. As a consequence, we can take into account both the variety of aspects covered by a single document, as well as its novelty in face of the aspects already covered by other retrieved documents. Moreover, the relative importance of each identified sub-query can be directly incorporated within our framework, so as to bias the diversification process towards more plausible aspects of the initial query. In a thorough experimentation, we evaluate the effectiveness of our framework using a large Web test collection, in the context of the diversity task of the TREC 2009 Web track [9]. In addition, to demonstrate its applicability in a real setting, we investigate two different strategies for sub-query generation, based on query reformulations provided by three major Web search engines. The results attest the effectiveness of our framework when compared to current state-of-the-art diversification approaches.

The remainder of this paper is organised as follows. Section 2 overviews related work on search result diversification. Section 3 further details our main contributions. Section 4 introduces our proposed framework for the diversification problem, its probabilistic derivation and the estimation of its components. Sections 5 and 6 describe the experimental setup and the evaluation of our proposed framework, respectively, in the context of the TREC 2009 Web track. Lastly, Section 7 presents our concluding remarks.

2. RELATED WORK

Search result diversification can be characterised as a bi-criterion optimisation problem, in which one seeks to maximise the overall relevance of a document ranking to multiple query aspects, while minimising its redundancy with respect to these aspects [14]. In its general form, this problem is an instance of the maximum coverage problem [18], which makes it NP-hard. Indeed, if we consider a query q as comprising a set of aspects A, and each document d from an initial ranking R as comprising a subset of the aspects in A, then the problem is to find a subset $S \subseteq R$, with $|S| \leq \tau$, such that $|\cup_{d_i \in S} d_i|$ is maximised for a given τ [1].

In practice, most previous works on search result diversification are based on a greedy approximation to this problem. Given a ranking R for an ambiguous query, a re-ranking S is produced by iteratively selecting a 'local-best' document from $R \setminus S$. This document should provide the maximum coverage of the aspects underlying the initial query, and the minimum redundancy with respect to the aspects covered by the documents already in S (i.e., the documents selected in previous iterations). The existing approaches differ mostly in how they account for the different query aspects. In particular, these approaches can be categorised as either *implicit* or *explicit* [27]. Implicit approaches assume that similar documents will cover similar aspects, and should hence be demoted in the final ranking, so as to reduce its overall redundancy. In turn, explicit approaches directly model the query aspects, actively seeking to maximise the coverage of their selected documents with respect to these aspects.

Among the implicit diversification approaches in the literature, the maximal marginal relevance (MMR) method of Carbonell and Goldstein [5] is the canonical example. At each iteration, the MMR method selects a document that has the highest combination of a similarity score with respect to a query and a dissimilarity score with respect to the documents selected at earlier ranks. Subsequent approaches inspired by MMR differ mainly in how the similarity between documents is computed. For instance, whereas Carbonell and Goldstein suggested using any content-based similarity function (e.g., cosine), Zhai and Lafferty [33] proposed to model relevance and redundancy within the language modelling framework. In particular, they devised several methods based on the Kullback-Leibler divergence measure and a simple mixture model. Chen and Karger [8] proposed a probabilistic approach to the related problem of retrieving one relevant document for a given query. By assuming that the previously selected documents are not relevant, they achieved an unplanned effect of diversification. More recently, Wang and Zhu [30] employed the correlation between documents as a measure of their similarity. Their work also showed that, by minimising this correlation, the overall variance of a document ranking is reduced, as well as the associated risk of overestimating its relevance.

Instead of accounting for the aspects covered by each document only implicitly, a promising direction is to explicitly model these aspects within the diversification approach. For instance, Agrawal et al. [1] investigated the diversification problem by employing a taxonomy for both queries and documents. In their work, two documents retrieved for a query are considered similar if they are confidently classified into one or more common categories covered by the query. Hence, documents classified into many categories are favoured, while those classified into already well-represented categories are penalised. Similarly, Carterette and Chandar [7] proposed a probabilistic model to maximise the coverage of a document ranking with respect to the aspects of a query, represented as topics or relevance models estimated from the top ranked documents. A different approach was investigated by Radlinski and Dumais [23]. They proposed to filter the ranking produced for a given query, so as to have a more even distribution of documents satisfying each aspect of this query. In particular, they uncovered query aspects from the query log of a commercial search engine.

Our approach also accounts for the aspects of a query in an explicit way. However, differently from the aforementioned approaches, we do not simply filter out documents satisfying an already well satisfied aspect. Moreover, we do not require that these aspects be estimated with respect to a predefined taxonomy, or using the top retrieved documents for the initial query. Instead, by representing the several query aspects as a set of sub-queries, we recognise the multiplicity behind an ambiguous query, without making any limiting assumption regarding the generation of these sub-queries. Furthermore, by estimating the relevance of each retrieved document to every identified sub-query, as well as the importance of each sub-query itself, we address the diversification problem in a principled and effective manner.

3. CONTRIBUTIONS OF THIS PAPER

The major contributions of this paper are:

- A novel probabilistic framework for search result diversification, which explicitly models the information need of an ambiguous query as a set of sub-queries;
- An analysis of the effectiveness of the sub-queries derived from two types of query reformulation provided by three major Web search engines;
- A thorough evaluation of the several components of our proposed framework, which naturally model different dimensions of the diversification problem.

4. EXPLICIT QUERY ASPECT DIVERSIFICATION

The diversification problem can be naturally stated as a tradeoff between finding relevant and novel information:

> Given an initial ranking R for a query q, find a re-ranking S that has the *maximum coverage* and the *minimum redundancy* with respect to the different aspects underlying q.

As discussed in Section 2, in its general form, this bi-criterion optimisation problem can be reduced from the maximum coverage problem [18], which makes it NP-hard [1]. Fortunately, there is a well-known approximation to this problem [5], which works well in practice [6], and is at the heart of most of the approaches to search result diversification presented in Section 2. However, most of these approaches seek to achieve the objective of diversification at the high expense of directly comparing documents to one another. On the other hand, the approaches that somehow explicitly account for the aspects associated to the initial query have their own limitations. For instance, they rely on filtering heuristics [23], or on the estimation of query aspects based on a predefined taxonomy [1] or on the documents retrieved for the initial query [7]. Instead, in this work, we propose to model the aspects associated to a query in a principled yet practical manner. In particular, we consider an ambiguous query as a compound rather than a single representation of an underlying information need. We then model this query as a set of sub-queries, which can be estimated in a variety of ways. In this work, we experiment with sub-queries derived from query reformulations provided by major Web search engines, as discussed in Section 4.2.2.

4.1 The xQuAD Framework

In this section, we introduce xQuAD (eXplicit Query Aspect Diversification), a novel probabilistic framework for search result diversification, which explicitly models an ambiguous query as a set of sub-queries. Sub-queries associated to an initial query can be effectively uncovered using mechanisms available to most modern Web search engines, e.g., query reformulations based on previous user interactions [3]. Moreover, probability theory provides xQuAD with an appropriate groundwork for handling the uncertainty incurred by the underspecification of information needs as queries. In particular, we derive our framework in light of the aforementioned approximation of the general diversification problem, as described in Section 2. The working scheme of xQuAD is described in Algorithm 1.

xQuAD(q, R, τ, λ)
1 $S \leftarrow \emptyset$
2 **while** $|S| < \tau$ **do**
3 $\quad d^* \leftarrow \arg\max_{d \in R \setminus S} (1-\lambda)\,\mathrm{P}(d|q) + \lambda\,\mathrm{P}(d, \bar{S}|q)$
4 $\quad R \leftarrow R \setminus \{d^*\}$
5 $\quad S \leftarrow S \cup \{d^*\}$
6 **end while**
7 **return** S

Algorithm 1: The xQuAD framework.

Given an ambiguous query q and an initial ranking R produced for this query, we build a new ranking S by iteratively selecting the τ highest scored documents from R, according to the following probability mixture model:

$$(1-\lambda)\,\mathrm{P}(d|q) + \lambda\,\mathrm{P}(d, \bar{S}|q), \tag{1}$$

where $\mathrm{P}(d|q)$ is the likelihood of document d being observed given the initial query q, and $\mathrm{P}(d, \bar{S}|q)$ is the likelihood of observing this document but not the documents already in S, which were selected in previous iterations of the algorithm. In particular, these two probabilities can be regarded as modelling *relevance* and *diversity*, respectively, with a mixing parameter λ controlling the tradeoff between the two.

In order to derive $\mathrm{P}(d, \bar{S}|q)$, we explicitly consider the possibly several aspects underlying the initial query q as a set of sub-queries, generated by some mechanism Q, such that $Q = \{q_1, \cdots, q_k\}$. By enforcing $\sum_{q_i \in Q} \mathrm{P}(q_i|q) = 1$, we can marginalise $\mathrm{P}(d, \bar{S}|q)$ across multiple sub-queries:

$$\mathrm{P}(d, \bar{S}|q) = \sum_{q_i \in Q} \mathrm{P}(q_i|q)\,\mathrm{P}(d, \bar{S}|q_i), \tag{2}$$

where $\mathrm{P}(q_i|q)$ can be seen as a measure of the relative *importance* of the sub-query q_i with respect to the other sub-queries associated to q. For instance, this probability could reflect the fraction of the user population that is interested in the aspect represented by the sub-query q_i more so than in other aspects of q. Next, assuming that the observation of the document d is independent of the documents already in S for a given sub-query q_i (since the documents in S are fixed at a given iteration), we break down $\mathrm{P}(d, \bar{S}|q_i)$ as:

$$\mathrm{P}(d, \bar{S}|q_i) = \mathrm{P}(d|q_i)\,\mathrm{P}(\bar{S}|q_i), \tag{3}$$

where $\mathrm{P}(d|q_i)$ is a measure of the *coverage* of document d with respect to the sub-query q_i. In turn, $\mathrm{P}(\bar{S}|q_i)$ provides a measure of *novelty*, as the probability of q_i not being satisfied by any of the documents already selected in S.

The independence assumption in Equation (3) 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 the sub-queries satisfied by a document. In other words, instead of comparing a document d to all documents already selected in S, as implicit diversification approaches would do, we estimate the utility of any document satisfying the sub-query q_i, given how well it is already satisfied by the documents in S. Although we achieve the same objective of promoting novelty, we do so in a much more efficient way. In particular, our approach does not require looking up the terms contained in all documents from the initial ranking R, so as to enable their direct comparison. Instead, we just update the novelty estimation of a given sub-query, based on the estimated relevance of each

document in S to this sub-query. In contrast to implicit approaches, this only incurs a few additional inverted file lookups for the documents matching the sub-query terms.

In order to derive $\mathrm{P}(\bar{S}|q_i)$, we further assume that the relevance of a document d_j in S to a given sub-query q_i is independent of the relevance of other documents in S to the same sub-query. Since our goal is to estimate the likelihood of the ranking S (as an entire *set* of documents) not satisfying the sub-query q_i, this constitutes a plausible assumption. Under this assumption, we have:

$$\begin{aligned}\mathrm{P}(\bar{S}|q_i) &= \mathrm{P}(\overline{d_1, \cdots, d_{n-1}}|q_i) \\ &= \prod_{d_j \in S} (1 - \mathrm{P}(d_j|q_i)). \end{aligned} \quad (4)$$

Finally, by replacing Equations (2), (3) and (4) into Equation (1), we have the final score computed by xQuAD:

$$(1-\lambda)\,\mathrm{P}(d|q) + \lambda \sum_{q_i \in Q} \Big[\mathrm{P}(q_i|q)\,\mathrm{P}(d|q_i) \prod_{d_j \in S} (1 - \mathrm{P}(d_j|q_i))\Big]. \quad (5)$$

4.2 Components Estimation

Within the xQuAD framework, several dimensions of the diversification problem are naturally modelled as individual probabilities. In practice, we estimate each of these probabilities as a separate component of the framework. Along with the sub-query generation mechanism, these components can be summarised as follows:

1. document relevance, $\mathrm{P}(d|q)$,
2. document diversity, $\mathrm{P}(d, \bar{S}|q)$:
   (a) sub-query generation, Q,
   (b) sub-query importance, $\mathrm{P}(q_i|q)$,
   (c) document coverage, $\sum_{q_i \in Q} \mathrm{P}(d|q_i)$,
   (d) document novelty, $\sum_{q_i \in Q|\,\mathrm{P}(d|q_i)>0} \mathrm{P}(\bar{S}|q_i)$.

In the remainder of this section, we propose suitable alternatives for estimating these components. In Section 6, the impact of each of these components on the diversification performance of xQuAD is thoroughly investigated.

4.2.1 Document Relevance, Coverage, and Novelty

The document relevance, coverage, and novelty components of the xQuAD framework are based on estimations of relevance. In particular, the document relevance component estimates the relevance of a document to the initial query, while the coverage and novelty components are based on relevance estimations with respect to sub-queries. In practice, any probabilistic retrieval approach can be used to produce these estimations, e.g., language modelling [17]. Moreover, different approaches can be deployed to produce the relevance estimation for each individual component.

In Section 6, we experiment with three effective document weighting models, from different families of probabilistic models, in order to estimate these components. In practice, we produce a document ranking for both the initial query, as well as each of the generated sub-queries. For the sake of clarity, a ranking produced for the initial query is denoted a *baseline ranking*, while those generated for sub-queries are denoted *sub-rankings*.

4.2.2 Sub-Query Generation

Sub-queries play a fundamental role within our proposed diversification framework. Indeed, the introduction of this component allows us to depart from the usually inefficient approach of directly comparing the retrieved documents to one another. More importantly, we diverge from the implicit assumption that similar documents will cover similar aspects underlying a query. Instead, by explicitly modelling these aspects in the form of sub-queries, we claim that a more effective search result diversification can be attained.

Several techniques can be used for generating keyword-based representations of query aspects in the form of sub-queries. For instance, using the target document collection itself, one could apply traditional query expansion techniques [26] in order to generate 'expanded sub-queries' from the top retrieved documents in a baseline ranking, or from different document clusters identified in this ranking [27, 32]. Alternatively, using external resources, such as a query log, one could mine sub-queries related to the initial query, by analysing patterns of query reformulations, or their distance to the initial query in a bipartite click-through graph [3, 31]. For instance, one could observe that documents clicked for the query 'java' are also likely to be clicked for the query 'sun microsystems'. Another observation could be that users frequently reformulate the query 'java' into 'java development kit', and less frequently so into 'java indonesia tourism'.

In this work, we investigate the effectiveness of using query reformulations provided by three major Web search engines (WSEs) as the sub-query generation mechanism Q of our diversification framework. To preserve anonymity, we refer to these search engines as A, B, and C. As further detailed in Section 5, for each of these WSEs and each of the 50 TREC 2009 Web track queries used in our investigation, we derive two sets of sub-queries, extracted in late July 2009:

- *related sub-queries*, as displayed alongside the results for the initial query, in the WSEs' interface,
- *suggested sub-queries*, as displayed in a dropdown list, as the initial query is typed in the WSEs' search box.

Table 1 shows the main statistics of the generated sub-queries: the average number of sub-queries per initial query, $\langle|Q|\rangle$, the average length of the generated sub-queries, $\langle|q_i|\rangle$, and the average number of results associated to each sub-query according to the Google WSE,[1] $\langle n_w(q_i)\rangle$.

WSE	Sub-queries	$\langle\|Q\|\rangle$	$\langle\|q_i\|\rangle$	$\langle n_w(q_i)\rangle$
A	related	7.37	2.88	10,439,498
	suggested	6.73	2.95	8,551,542
B	related	9.74	2.64	10,258,938
	suggested	9.30	3.31	7,142,431
C	related	15.92	1.84	18,130,123
	suggested	8.86	3.29	9,544,095

Table 1: Statistics of the generated sub-queries.

From Table 1, we can observe that the three WSEs provide roughly the same number of sub-queries on average, evenly distributed between related and suggested sub-queries. The exception is WSE C, which provides almost double the number of related sub-queries when compared to the other WSEs.

[1]For the sake of uniformity, we choose Google as a single source to derive result set statistics for the sub-queries generated from the three considered WSEs.

The average sub-query length is also very similar across the different WSEs, with the short sub-queries resembling typical Web queries [19]. However, these sub-queries are overall longer than the 50 queries considered in our experiments. Indeed, the TREC 2009 Web track queries have an average length of 2.1 terms. This observation suggests that the generated sub-queries are likely to correspond to alternative specialisations of the initial queries [4]. Finally, the estimated result set size of the average sub-query also shows a high similarity across WSEs, with the related sub-queries yielding a consistently bigger size when compared to the suggested ones.

Although the precise mechanisms used by these WSEs for producing query reformulations are not publicly known, these mechanisms can be arguably regarded as delivering state-of-the-art query log mining, even if only on the basis that the WSEs have a wealth of click data available. Moreover, by evaluating them as black-box implementations of the sub-query generation component of xQuAD, we can draw useful insights regarding their effectiveness in providing a diverse coverage of the aspects of an initial query.

4.2.3 Sub-Query Importance

In order to favour sub-queries more likely to represent aspects of interest to the user population, we propose three different ways of estimating the sub-query importance component, $\mathrm{P}(q_i|q)$, within our framework. The first of these can be seen as a baseline estimation mechanism, which considers all sub-queries as being equally important:

$$\mathrm{P}_u(q_i|q) = \frac{1}{|Q|}, \tag{6}$$

where, as before, Q represents the set of sub-queries generated with respect to the initial query q.

However, the relative importance of the sub-queries generated from an initial query should ideally reflect the interest of information consumers (i.e., the user population) in the particular aspect represented by each sub-query [10]. This could be estimated, for instance, based on the relative frequency of each sub-query in a query log. When no such data is available to estimate the interest of information consumers in a particular aspect, an alternative is to estimate the interest this aspect sparks from information providers. In particular, we propose to estimate the relative importance of each sub-query based on how well it is covered by a given collection. Our next importance estimator builds upon this idea and relies once more on information available from all major Web search engines. It is given as:

$$\mathrm{P}_w(q_i|q) = \frac{n_w(q_i)}{\sum_{q_j \in Q} n_w(q_j)}, \tag{7}$$

where $n_w(q_i)$ is the estimated number of results retrieved for the sub-query q_i according to the search engine w. As discussed in Section 4.2.2, we rely on estimates from Google.

As an alternative to relying on an external resource, we propose an analogous estimator, which is solely based on a local corpus. In particular, we estimate the relative importance of each generated sub-query, by considering the ranking produced for this sub-query as a sample of the documents covering this sub-query in the local corpus. This estimation mechanism is inspired by the Central Rank-based Collection Selection (CRCS [28]) algorithm in distributed information retrieval. To rank distributed collections of documents for a given query, CRCS builds a centralised ranking of documents sampled from the different collections. The rank of each candidate collection is then computed based on its estimated size and the rank of its documents in the centralised ranking. Inspired by CRCS, we devise our third and final sub-query importance estimation mechanism as:

$$i_c(q_i|q) = \frac{n_c(q_i)}{\max_{q_i \in Q} n_c(q_i)} \frac{1}{\hat{n}_c(q_i)} \sum_{d|\,\mathrm{P}(d|q_i)>0} \tau - j(d,q), \tag{8}$$

where $n_c(q_i)$ is the total number of results retrieved for q_i in the local corpus, $\hat{n}_c(q_i)$ corresponds to the number of results associated to the sub-query q_i that are among the top τ ranked results for the initial query q, with $j(d,q)$ giving the ranking position of the document d with respect to q. Finally, the estimator $i_c(q_i|q)$ is further normalised to yield a probability distribution:

$$\mathrm{P}_c(q_i|q) = \frac{i_c(q_i|q)}{\sum_{q_j \in Q} i_c(q_j|q)}. \tag{9}$$

5. EXPERIMENTAL SETUP

In this section, we describe the experimental setup that supports the evaluation of our proposed framework, which is reported in Section 6. In particular, our experimentation aims to answer three main research questions:

1. Does explicitly modelling the aspects of a query help in diversifying the results for this query?
2. Are query suggestions provided by Web search engines an effective resource for explicit diversification?
3. What is the impact of the components of xQuAD on the performance of the whole framework?

In the following, we detail the document collection, the topics, and the metrics used in our evaluation. Additionally, we describe the baselines to which our approach is compared, including the training procedure to set their parameters, when necessary. The Terrier IR platform[2] [21] is used for both indexing and retrieval, with Porter's stemmer and standard English stopwords removal.

5.1 Collection and Topics

Our experiments are conducted in the context of the diversity task of the TREC 2009 Web track [9]. The goal of this task is to produce a ranking of documents for a given query that maximises the coverage of the possible aspects underlying this query, while reducing its overall redundancy with respect to the covered aspects. The test collection used in this task is the new TREC ClueWeb09 dataset.[3] In our experiments, we consider a subset of this collection, as used in the TREC 2009 Web track, comprising a total of 50 million English Web documents.

A total of 50 topics were available for this task. Each topic includes from 3 to 8 sub-topics, as identified by TREC assessors, with relevance judgements provided at the sub-topic level. Figure 1 illustrates an example topic with different fields, including its identified sub-topics. In our experiments, the 'query' field of a topic is used as the initial query. Besides the sub-queries generated based on the

[2] http://www.terrier.org

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

```
<topic number="1" type="faceted">
  <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 1: TREC 2009 Web track, topic 1, along with its corresponding sub-topics.

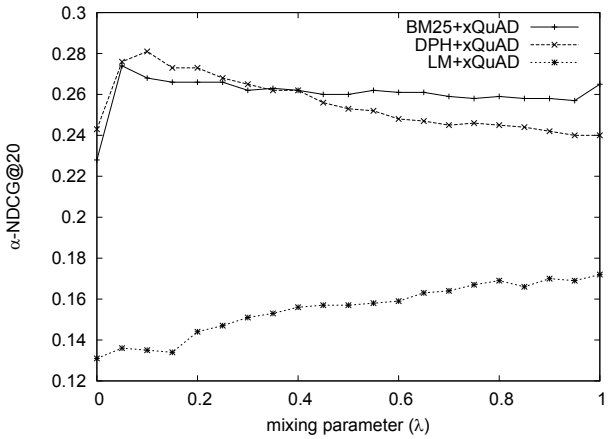

Figure 2: Tradeoff between relevance and diversity.

WSEs' reformulations for each of the considered 50 queries, as discussed in Section 4.2.2, we build an alternative set of sub-queries from the provided official sub-topics. This provides an upper-bound sub-query generation mechanism for evaluating the sub-queries derived from the considered WSEs. Moreover, by employing this upper-bound as a uniform, standard setting, we can focus on evaluating the diversification strategy deployed by our framework compared to that of state-of-the-art diversification approaches.

5.2 Evaluation Metrics

The evaluation results in the diversity task of the TREC 2009 Web track are reported according to two official metrics: α-NDCG and IA-P. The α-normalised discounted cumulative gain (α-NDCG [10]) metric balances relevance and diversity through the tuning parameter α. The larger the value of α, the more diversity is rewarded. In contrast, when $\alpha = 0$, only relevance is rewarded, and this metric is equivalent to the traditional NDCG [20].

Besides α-NDCG, our evaluation is also based on a generalisation of standard IR metrics that rewards the diversity of a ranking. In particular, we use the intent-aware precision (IA-P [1]) metric, which extends the traditional notion of precision in order to account for the possible aspects underlying a query and their relative importance.

In our evaluation, both α-NDCG and IA-P are reported at three different rank cutoffs: 5, 10, and 100. These cutoffs focus on the evaluation at early ranks, which are particularly important in a Web search context [19]. Both α-NDCG and IA-P are computed following the standard practice in the TREC 2009 Web track [9]. In particular, α-NDCG is computed with $\alpha = 0.5$, in order to give equal weights to both relevance and diversity, and IA-P is computed with all query aspects considered equally important.

5.3 Retrieval Baselines

We evaluate the effectiveness of xQuAD at diversifying the search results produced by three effective probabilistic document weighting models: BM25 [25], the DPH Divergence From Randomness (DFR) model [2], and Hiemstra's language modelling (LM) [17]. In particular, we employ these weighting models with their often suggested default settings: BM25's $b = 0.75$ [25], DPH's $c = 1.0$ [2], and LM's $\lambda_{LM} = 0.15$ [17]. As discussed in Section 4.2.1, these weighting models are used to produce both a baseline ranking as well as the sub-rankings for different sub-queries.

Besides the baseline ranking produced for an initial query, we compare xQuAD to both implicit and explicit diversification approaches, deployed on top of this baseline ranking. To differentiate between these approaches and the baseline ranking, the former are referred to as *diversification baselines*. These include the approaches of Carbonell and Goldstein [5], Radlinski and Dumais [23], and Agrawal et al. [1], described in Section 2. In particular, the last two make use of external resources or judgements, such as query logs or a classification taxonomy, which are not available for the test collection at hand. Therefore, we simulate their best-case scenario, by considering the official sub-topics provided by the collection as input to their diversification models. To simulate the approach of Radlinski and Dumais [23], the official Web track sub-topics are directly used as a proxy for query log reformulations. As for the approach of Agrawal et al. [1], the official sub-topics are used as a representation of taxonomy classes. The confidence of the classification of a document to a given class, in turn, is surrogated by the estimated relevance of the document to the sub-topic that represents the class. All techniques are applied to re-rank the top $\tau = 1000$ documents retrieved by the baseline ranking for each query.

5.4 Training Procedure

As discussed in Section 4, the xQuAD framework is based on a mixture of a relevance component and a diversity component, parameterised by λ. In order to illustrate the tradeoff between these two components, Figure 2 shows the performance of xQuAD when varying the parameter λ, with the official TREC sub-topics used as sub-queries. From Figure 2, we observe that our framework consistently improves over the baseline ranking ($\lambda = 0$) for a range of λ values, with a peak around 0.15 for both BM25 and DPH. For LM, which exhibits a lower performance, a more aggressive diversification approach (i.e., a larger λ) seems more appropriate.

In our experiments, in order to train λ, we perform a 5-fold cross validation over the 50 topics, optimising for α-

NDCG@10, the primary evaluation metric at the diversity task of the TREC 2009 Web track. The same approach is used to train a similar interpolation parameter employed by MMR, in order to trade off its relevance and novelty estimations [5]. Note that the approaches of Radlinski and Dumais [23] and Agrawal et al. [1] do not require training under their simulated best-case scenario.

6. EXPERIMENTAL EVALUATION

In this section, we thoroughly evaluate the xQuAD framework and the impact of its components on a diversification task. In order to answer the main research questions stated in Section 5, we proceed as follows. In Section 6.1, we assess the diversification effectiveness of xQuAD, by comparing it to state-of-the-art diversification approaches in their simulated best-case scenario, as described in Section 5. In Section 6.2.1, we investigate the impact of the sub-query generation component in our framework, by comparing the performance of xQuAD using related and suggested sub-queries provided by major WSEs. In Section 6.2.2, we evaluate the three sub-query importance estimation mechanisms introduced in Section 4.2.3. Finally, in Section 6.2.3, by attempting to improve the baseline ranking and also the sub-rankings, we evaluate the impact of the relevance, coverage, and novelty components on the performance of xQuAD.

6.1 Framework Performance

In this experiment, we aim to answer the first of our stated research questions, namely, whether accounting for the aspects underlying a query in an explicit way helps diversifying the results for this query. To investigate this, we evaluate the performance of xQuAD at diversifying the baseline rankings produced by three different weighting models: BM25, DPH, and LM. Additionally, we compare its performance to that of three diversification baselines. The classical MMR method of Carbonell and Goldstein [5] is used as a representative of implicit diversification approaches. As explicit diversification baselines, we consider simulated versions of the approaches of Radlinski and Dumais [23] and Agrawal et al. [1], which we refer to as Q-Filter and IA-Select, respectively. As discussed in Section 5, in our simulation, we experiment with both xQuAD and these explicit diversification baselines using the official TREC 2009 Web track sub-topics. By doing so, we can isolate the impact of the query generation component and focus on comparing the diversification strategies provided by these approaches.

Table 2 shows the results of this evaluation in terms of α-NDCG and IA-P. The best result per baseline ranking is highlighted in bold. To provide a fair comparison to the diversification baselines, which do not take into account the relative importance of different query aspects, xQuAD is applied with the P_u importance estimator, as given by Equation (6). Accordingly, it is denoted xQuAD$_u$.

From Table 2, we first observe that xQuAD markedly outperforms the diversification baselines in most settings. In particular, it is the only approach that consistently improves the initial baseline ranking provided by BM25, DPH, and LM, with gains of up to 30% (BM25, α-NDCG@5). The only exception is IA-P for the DPH baseline ranking, in which case the initial ranking itself performs the best, followed closely by the other approaches. However, none of these differences is statistically significant, according to the Wilcoxon signed-rank test with $p < 0.05$. A further investigation shows that the considered weighting models underperform for some queries, particularly at earlier ranks, as emphasised by the reported metrics. We hypothesise that improving the relevance estimations produced by these weighting models could result in further improvements. An initial analysis in this direction is conducted in Section 6.2.3. Overall, the obtained results attest the effectiveness of the explicit diversification strategy implemented by xQuAD when compared to other diversification approaches.

	α-NDCG			IA-P		
	@5	@10	@100	@5	@10	@100
BM25	0.159	0.186	0.288	0.075	0.071	**0.059**
+MMR	0.120	0.150	0.224	0.056	0.058	0.039
+Q-Filter	0.159	0.186	0.286	0.075	0.071	0.057
+IA-Select	0.110	0.119	0.180	0.043	0.037	0.023
+xQuAD$_u$	**0.208**	**0.227**	**0.324**	**0.080**	**0.075**	0.056
DPH	0.198	0.212	0.304	**0.109**	**0.106**	**0.062**
+MMR	0.195	0.211	0.303	0.105	0.103	**0.062**
+Q-Filter	0.198	0.212	0.303	**0.109**	**0.106**	0.060
+IA-Select	0.148	0.157	0.203	0.077	0.071	0.023
+xQuAD$_u$	**0.208**	**0.243**	**0.334**	0.097	0.096	0.061
LM	0.082	0.096	0.180	0.041	0.040	0.032
+MMR	0.083	0.096	0.183	0.041	0.039	0.032
+Q-Filter	0.078	0.095	0.179	0.040	0.040	0.031
+IA-Select	0.081	0.086	0.127	0.037	0.027	0.014
+xQuAD$_u$	**0.085**	**0.104**	**0.198**	**0.045**	**0.042**	**0.034**

Table 2: Diversification performance using the official TREC 2009 Web track diversity sub-topics.

6.2 Components Performance

In the previous section, we have demonstrated the effectiveness of the xQuAD framework for search result diversification across different baseline rankings, when compared to state-of-the-art diversification approaches. In the remainder of this section, we investigate the performance of alternative implementations for its components, and how they impact the performance of the framework as a whole.

6.2.1 Sub-Query Generation

Recalling our main research questions, in this experiment, we investigate the effectiveness of using query reformulations provided by three major WSEs for the diversification task. In particular, we resort to the related and suggested sub-queries provided by these WSEs in order to implement the sub-query generation component of xQuAD. Table 3 shows the performance of xQuAD using the obtained sub-queries to re-rank the results provided by the same baseline rankings used in the previous experiment, namely, BM25, DPH, and LM. The best performance with respect to each of these baselines is highlighted in bold. Once again, xQuAD is applied with the P_u sub-query importance estimator.

From Table 3, we first observe that, in general, a lower performance is obtained when compared to the results in Table 2, which simulate an upper-bound query generation mechanism. Nonetheless, improvements over the initial ranking are still obtained in most settings. When we compare the performance of xQuAD using query reformulations provided by the three WSEs, no clear trend can be observed. This suggests that the considered WSEs' query reformulation mechanisms perform comparably, at least in terms of their effectiveness in covering diverse aspects of an initial query. A distinction can be made, however, between the two types of sub-queries derived from the WSEs. In partic-

		related sub-queries						suggested sub-queries					
		α-NDCG			IA-P			α-NDCG			IA-P		
	WSE	@5	@10	@100	@5	@10	@100	@5	@10	@100	@5	@10	@100
BM25		0.159	**0.186**	**0.288**	0.075	0.071	**0.059**	0.159	**0.186**	0.288	0.075	**0.071**	**0.059**
+xQuAD$_u$	A	0.154	0.184	0.282	0.070	0.072	0.057	**0.171**	**0.186**	**0.291**	0.082	**0.071**	0.053
+xQuAD$_u$	B	0.154	0.182	0.279	0.073	**0.076**	0.054	0.129	0.158	0.261	0.065	0.067	0.052
+xQuAD$_u$	C	**0.161**	0.182	0.285	**0.076**	**0.076**	0.057	0.163	0.184	0.287	**0.084**	0.069	0.053
DPH		0.198	**0.212**	0.304	**0.109**	**0.106**	**0.062**	0.198	0.212	0.304	**0.109**	**0.106**	**0.062**
+xQuAD$_u$	A	0.164	0.189	0.288	0.086	0.083	0.056	**0.215**	0.222	0.313	0.108	0.088	0.055
+xQuAD$_u$	B	0.186	0.205	0.295	0.090	0.082	0.057	0.162	0.189	0.281	0.088	0.085	0.055
+xQuAD$_u$	C	**0.206**	0.209	**0.307**	0.108	0.090	**0.062**	0.201	**0.236**	**0.320**	0.093	0.092	0.059
LM		0.082	0.096	0.180	**0.041**	0.040	0.032	0.082	0.096	0.180	0.041	0.040	0.032
+xQuAD$_u$	A	**0.088**	0.103	**0.192**	0.038	0.038	0.032	**0.101**	0.123	0.204	0.043	0.046	0.032
+xQuAD$_u$	B	0.081	**0.105**	0.188	0.040	**0.045**	**0.033**	0.093	0.118	0.197	0.041	0.043	0.033
+xQuAD$_u$	C	0.082	0.100	0.183	0.037	0.039	0.032	**0.101**	**0.127**	**0.205**	**0.046**	**0.047**	**0.034**

Table 3: Diversification performance using related and suggested sub-queries from different WSEs.

	α-NDCG			IA-P		
	@5	@10	@100	@5	@10	@100
BM25	0.159	0.186	0.288	0.075	0.071	**0.059**
+xQuAD$_u$	**0.208**	**0.227**	**0.324**	**0.080**	**0.075**	0.056
+xQuAD$_c$	0.176	0.206	0.296	0.066	0.066	0.048
+xQuAD$_w$	0.184	0.201	0.297	0.077	0.067	0.053
DPH	0.198	0.212	0.304	**0.109**	**0.106**	**0.062**
+xQuAD$_u$	**0.208**	**0.243**	**0.334**	0.097	0.096	0.061
+xQuAD$_c$	0.169	0.204	0.299	0.073	0.073	0.053
+xQuAD$_w$	0.203	0.226	0.316	0.101	0.088	0.060
LM	0.082	0.096	0.180	0.041	0.040	0.032
+xQuAD$_u$	0.085	0.104	0.198	**0.045**	0.042	0.034
+xQuAD$_c$	**0.110**	**0.146**	**0.234**	0.044	**0.047**	**0.041**
+xQuAD$_w$	0.078	0.095	0.187	0.039	0.039	0.033

Table 4: Diversification performance using different sub-query importance estimators.

ular, the suggested sub-queries outperform the related ones in most settings. Looking back at Table 1, the observation that the suggested sub-queries tend to produce considerably smaller result sets might be an indication of their suitability in discriminating between different aspects of a query.

6.2.2 Sub-Query Importance

Besides generating a quality set of sub-queries, we hypothesise that the relative importance given to each sub-query might influence the overall diversification performance of our framework. To investigate this, we experiment with xQuAD using three different sub-query importance estimators, as introduced in Section 4.2.3. In particular, Table 4 shows the performance of xQuAD using two collection-based importance estimators when compared to the previously introduced uniform importance estimator, P_u. As discussed in Section 4.2.3, the P_w estimator (Equation (7)) is based on the result set size produced for a given sub-query, as estimated by the Google WSE, whereas P_c (Equation (9)) relies on estimates derived from the target collection itself, inspired by a resource selection approach. In Table 4, the use of these estimators is denoted by their respective subscripts being used by xQuAD (e.g., xQuAD$_u$ stands for the use of xQuAD with the P_u importance estimator). As in the previous experiments, the diversification performance is evaluated across three baseline rankings and measured according to α-NDCG and IA-P at different cutoffs. The best value for each baseline ranking and each evaluation metric is highlighted in bold.

From Table 4, we observe that the P_u estimator consistently outperforms the others. Among the collection-based estimators (i.e., P_w and P_c), P_w generally performs better. As discussed in Section 4.2.3, this estimator is based on a much bigger resource than the ClueWeb09 dataset, namely, the Google index. An exception is the LM baseline, which markedly benefits from our resource selection-inspired estimator, P_c, in terms of both α-NDCG and IA-P. Overall, the best performance attained by the simpler uniform estimator is not totally unexpected, since neither α-NDCG nor IA-P reward approaches that take non-uniform aspect importance distributions into account.[4] Nevertheless, we hypothesise that a good importance estimator might be related to features other than the relative popularity of a sub-query. For instance, the overlap between the sub-rankings produced for different sub-queries may have an impact on the performance of xQuAD, as its coverage and novelty components estimate the relevance of a document to multiple sub-queries. This investigation, however, is left for future work.

6.2.3 Relevance, Coverage, and Novelty

Sections 6.2.1 and 6.2.2 investigated the impact of different sub-query generation and importance estimators, respectively, on the performance of xQuAD. In this section, we investigate the influence of three other important components of our framework. As described in Section 4.2.1, the relevance component is based on relevance estimations with respect to the initial query, while the coverage and novelty components depend on relevance estimations with respect to sub-queries. Accordingly, we experiment with xQuAD by improving the document weighting models that have been used so far for estimating these three components.

In this experiment, we apply the pBiL proximity model from the DFR framework [22], in order to favour documents in which the query terms appear in close proximity. In particular, variants of this model were shown to improve ad-hoc retrieval performance on several TREC collections [15]. In our investigation, we apply this model to both the baseline ranking (the relevance component) as well as the sub-rankings generated for the various identi-

[4]Although the IA-P metric can account for the relative importance of different query aspects, it is not trivial to derive a ground-truth importance distribution for evaluation purposes. Hence, as discussed in Section 5.2, IA-P is computed assuming that all aspects are equally important, in accordance with the TREC 2009 Web track setup.

	α-NDCG			IA-P		
	@5	@10	@100	@5	@10	@100
BM25	0.159	0.186	0.288	0.075	0.071	0.059
+xQuAD$_u$	**0.208**	**0.227**	**0.324**	**0.080**	0.075	0.056
+xQuAD$_u$ (b)	0.147	0.181	0.280	0.076	**0.085**	0.061
+xQuAD$_u$ (s)	0.157	0.188	0.283	0.076	0.078	0.061
+xQuAD$_u$ (bs)	0.148	0.168	0.267	0.079	0.080	**0.063**
DPH	0.198	0.212	0.304	**0.109**	**0.106**	0.062
+xQuAD$_u$	0.208	**0.243**	**0.334**	0.097	0.096	0.061
+xQuAD$_u$ (b)	0.165	0.207	0.306	0.088	0.092	0.063
+xQuAD$_u$ (s)	**0.212**	0.230	0.323	0.102	0.094	0.063
+xQuAD$_u$ (bs)	0.185	0.204	0.304	0.096	0.088	**0.064**
LM	0.082	0.096	0.180	0.041	0.040	0.032
+xQuAD$_u$	0.085	0.104	0.198	0.045	0.042	0.034
+xQuAD$_u$ (b)	0.125	0.156	0.247	0.072	**0.076**	0.050
+xQuAD$_u$ (s)	0.117	0.133	0.215	0.057	0.051	0.038
+xQuAD$_u$ (bs)	**0.132**	**0.162**	**0.248**	**0.078**	**0.076**	**0.053**

Table 5: Diversification performance by enhancing the baseline ranking (b), the ranking for each sub-query (s), or both (bs). xQuAD is applied with the uniform importance estimator.

	α-NDCG			IA-P		
	@5	@10	@100	@5	@10	@100
BM25	0.159	0.186	0.288	0.075	0.071	0.059
+xQuAD$_c$	0.176	0.206	0.296	0.066	0.066	0.048
+xQuAD$_c$ (b)	**0.183**	**0.208**	**0.300**	0.074	0.068	0.052
+xQuAD$_c$ (s)	0.163	0.191	0.297	**0.076**	**0.079**	0.059
+xQuAD$_c$ (bs)	0.144	0.172	0.278	0.065	0.070	**0.061**
DPH	0.198	0.212	0.304	**0.109**	**0.106**	**0.062**
+xQuAD$_c$	0.169	0.204	0.299	0.073	0.073	0.053
+xQuAD$_c$ (b)	0.160	0.200	0.297	0.069	0.075	0.054
+xQuAD$_c$ (s)	**0.205**	**0.235**	**0.327**	0.096	0.097	0.059
+xQuAD$_c$ (bs)	0.173	0.206	0.305	0.068	0.072	0.059
LM	0.082	0.096	0.180	0.041	0.040	0.032
+xQuAD$_c$	0.110	0.146	0.234	0.044	0.047	0.041
+xQuAD$_c$ (b)	0.119	0.151	0.242	0.053	0.055	0.044
+xQuAD$_c$ (s)	**0.144**	**0.190**	0.265	**0.062**	**0.077**	0.049
+xQuAD$_c$ (bs)	0.138	0.180	**0.274**	0.059	0.073	**0.057**

Table 6: Diversification performance by enhancing the baseline ranking (b), the ranking for each sub-query (s), or both (bs). xQuAD is applied with the resource selection-inspired importance estimator.

fied sub-queries (the coverage and novelty components). Table 5 presents the results of this investigation. In the table, (b) and (s) stand for the application of the pBiL proximity model in order to enhance the baseline ranking and the sub-rankings, respectively, whereas (bs) account for the application of this technique to both components. For uniformity, once again, xQuAD is applied with the P_u importance estimator. The performance of xQuAD without any enhancement is included as an additional baseline.

From Table 5, we can observe that the diversification performance of xQuAD can be further improved by improving the baseline ranking (b), the sub-rankings (s), or both (bs). Note, however, that these improvements are only consistent when the initially weaker LM baseline is considered. In the case of LM, improvements can be substantial, for both the α-NDCG and IA-P metrics and across all cutoffs, particularly when all three components are improved (the (bs) variant). Nonetheless, for BM25 and DPH, improving the performance of individual components can also harm the diversification performance of xQuAD as a whole. This is likely due to the aforementioned tradeoff between relevance and diversity (as illustrated in Figure 2), and could probably be overcome by having the xQuAD's mixing parameter λ appropriately trained for the enhanced components.

As a final observation, we note that improvements on these components may also impact the performance of other components. To illustrate this effect, Table 6 presents similar results to those presented in Table 5, however using our resource selection-inspired importance estimator (P_c) instead of the uniform one (P_u). In particular, as this estimator takes into account how well the documents retrieved for a particular sub-query are ranked with respect to the initial query, it can substantially benefit from improvements to both the baseline ranking and the sub-rankings.

As shown in Table 6, improvements can be attained on top of all baselines, and are particularly marked for the LM baseline. Indeed, the performance of xQuAD on top of this relatively weaker baseline is raised to a comparable level as that of our framework using BM25 or DPH. This observation attests the potential benefit of improving the relevance, coverage, and novelty components of xQuAD. Furthermore, it highlights the benefit of handling these components within a unified diversification framework, which allows for different estimation choices to be made in a principled manner.

7. CONCLUSIONS AND FUTURE WORK

In this paper, we have introduced a novel probabilistic framework for search result diversification. In particular, the xQuAD (eXplicit Query Aspect Diversification) framework explicitly models the aspects underlying an initial query, in the form of sub-queries. Instead of comparing documents to one another—which usually demands expensive computations—our approach achieves an effective diversification performance by directly estimating the relevance of the retrieved documents to multiple sub-queries. Besides being efficient in practice, the principled formulation of xQuAD naturally models several dimensions of interest in a diversification task, as components within the framework. These include the relevance of a document to an initial query and its multiple aspects, identified as sub-queries, as well as the relative importance of each sub-query and how novel a document satisfying each sub-query is.

We have thoroughly investigated the effectiveness of the xQuAD framework at diversifying Web search results, within the standard experimentation paradigm provided by the diversity task of the TREC 2009 Web track. In particular, by simulating an upper-bound sub-query generation mechanism, we have shown that our framework outperforms existing implicit and explicit diversification approaches across several settings. By investigating the impact of each of the components of our framework, we have shown that effective sub-queries can be generated based on query reformulations provided by major Web search engines. Reformulations provided in the form of suggested queries showed encouraging promise in generating a diverse representation of the aspects underlying an initial query. Moreover, we have experimented with different mechanisms to estimate the relative importance of each uncovered sub-query, based on statistics derived from both the local collection and from the index of the Google WSE. Our results have shown that these estimators can bring further performance improvements, particularly when associated with a better estimation of the relevance and coverage components of xQuAD.

In the future, we plan to further improve xQuAD by investigating alternative mechanisms for estimating each of its components. For instance, different query reformulation approaches could be investigated in order to generate effective sub-queries, as well as to better estimate their relative importance with respect to other sub-queries. In addition, the relevance, coverage, and novelty components could be enhanced by the deployment of more sophisticated document retrieval techniques. A promising direction for investigation is the analysis of the type of each individual sub-query (e.g., navigational sub-queries are likely to benefit more from the use of link analysis than informational ones).

8. ACKNOWLEDGEMENTS

We are thankful to Mark Girolami for useful discussions on the probabilistic interpretation of the xQuAD framework.

9. REFERENCES

[1] R. Agrawal, S. Gollapudi, A. Halverson, and S. Ieong. Diversifying search results. In *Proc. of WSDM*, pages 5–14, 2009.

[2] G. Amati, E. Ambrosi, M. Bianchi, C. Gaibisso, and G. Gambosi. FUB, IASI-CNR and University of Tor Vergata at TREC 2007 Blog track. In *Proc. of TREC*, 2007.

[3] R. A. Baeza-Yates, C. A. Hurtado, and M. Mendoza. Query recommendation using query logs in search engines. In *Proc. of EDBT Workshops*, pages 588–596, 2004.

[4] P. Boldi, F. Bonchi, C. Castillo, and S. Vigna. From "Dango" to "Japanese cakes": query reformulation models and patterns. In *Proc. of WI-IAT*, pages 183–190, 2009.

[5] J. Carbonell and J. Goldstein. The use of MMR, diversity-based reranking for reordering documents and producing summaries. In *Proc. of SIGIR*, pages 335–336, 1998.

[6] B. Carterette. An analysis of NP-completeness in novelty and diversity ranking. In *Proc. of ICTIR*, pages 200–211, 2009.

[7] B. Carterette and P. Chandar. Probabilistic models of ranking novel documents for faceted topic retrieval. In *Proc. of CIKM*, pages 1287–1296, 2009.

[8] H. Chen and D. R. Karger. Less is more: probabilistic models for retrieving fewer relevant documents. In *Proc. of SIGIR*, pages 429–436, 2006.

[9] C. L. A. Clarke, N. Craswell, and I. Soboroff. Preliminary report on the TREC 2009 Web track. In *Proc. of TREC*, 2009.

[10] C. L. A. Clarke, M. Kolla, G. V. Cormack, O. Vechtomova, A. Ashkan, S. Büttcher, and I. MacKinnon. Novelty and diversity in information retrieval evaluation. In *Proc. of SIGIR*, pages 659–666, 2008.

[11] C. L. A. Clarke, M. Kolla, and O. Vechtomova. An effectiveness measure for ambiguous and underspecified queries. In *Proc. of ICTIR*, pages 188–199, 2009.

[12] W. S. Cooper. The inadequacy of probability of usefulness as a ranking criterion for retrieval system output. Technical report, Univ. of California, 1971.

[13] W. Goffman. On relevance as a measure. *IP&M*, 2(3):201–203, 1964.

[14] S. Gollapudi and A. Sharma. An axiomatic approach for result diversification. In *Proc. of WWW*, pages 381–390, 2009.

[15] B. He, C. Macdonald, I. Ounis, J. Peng, and R. L. T. Santos. University of Glasgow at TREC 2008: experiments in Blog, Enterprise, and Relevance Feedback tracks with Terrier. In *Proc. of TREC*, 2008.

[16] M. A. Hearst. *Search User Interfaces*. Cambridge University Press, 2009.

[17] D. Hiemstra. *Using Language Models for Information Retrieval*. PhD thesis, Univ. of Twente, 2001.

[18] D. S. Hochbaum, editor. *Approximation algorithms for NP-hard problems*. PWS Publishing Co., 1997.

[19] B. J. Jansen, A. Spink, J. Bateman, and T. Saracevic. Real life information retrieval: a study of user queries on the Web. *SIGIR Forum*, 32(1):5–17, 1998.

[20] K. Järvelin and J. Kekäläinen. Cumulated gain-based evaluation of IR techniques. *ACM TOIS*, 20(4):422–446, 2002.

[21] I. Ounis, G. Amati, V. Plachouras, B. He, C. Macdonald, and C. Lioma. Terrier: a high performance and scalable information retrieval platform. In *Proc. of SIGIR, OSIR Workshop*, 2006.

[22] J. Peng, C. Macdonald, B. He, V. Plachouras, and I. Ounis. Incorporating term dependency in the DFR framework. In *Proc. of SIGIR*, pages 843–844, 2007.

[23] F. Radlinski and S. Dumais. Improving personalized web search using result diversification. In *Proc. of SIGIR*, pages 691–692, 2006.

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

[25] S. E. Robertson, S. Walker, S. Jones, M. Hancock-Beaulieu, and M. Gatford. Okapi at TREC-3. In *Proc. of TREC*, 1994.

[26] J. J. Rocchio. Relevance feedback in information retrieval. In *The SMART Retrieval System*, pages 313–323. 1971.

[27] R. L. T. Santos, J. Peng, C. Macdonald, and I. Ounis. Explicit search result diversification through sub-queries. In *Proc. of ECIR*, 2010.

[28] M. Shokouhi. Central-rank-based collection selection in uncooperative distributed information retrieval. In *Proc. of ECIR*, pages 160–172, 2007.

[29] K. Spärck-Jones, S. E. Robertson, and M. Sanderson. Ambiguous requests: implications for retrieval tests, systems and theories. *SIGIR Forum*, 41(2):8–17, 2007.

[30] J. Wang and J. Zhu. Portfolio theory of information retrieval. In *Proc. of SIGIR*, pages 115–122, 2009.

[31] J. Yi and F. Maghoul. Query clustering using click-through graph. In *Proc. of WWW*, pages 1055–1056, 2009.

[32] H.-J. Zeng, Q.-C. He, Z. Chen, W.-Y. Ma, and J. Ma. Learning to cluster Web search results. In *Proc. of SIGIR*, pages 210–217, 2004.

[33] C. Zhai, W. W. Cohen, and J. Lafferty. Beyond independent relevance: methods and evaluation metrics for subtopic retrieval. In *Proc. of SIGIR*, pages 10–17, 2003.