SCORE AND RANK AGGREGATION METHODS FOR EXPLICIT SEARCH RESULT DIVERSIFICATION

Ahmet Murat Ozdemiray and Ismail Sengor Altingovde

TECHNICAL REPORT

METU-CENG-2013-01

September 2013

Department of Computer Engineering
Middle East Technical University
Dumlupinar Bulvari, 06531, Ankara
TURKEY

Score and Rank Aggregation Methods For Explicit Search Result Diversification

Ahmet Murat Ozdemiray[1], Ismail Sengor Altingovde[2] (Corr. author)

Computer Engineering Department, Middle East Technical University, Ankara, Turkey

[1]murat.ozdemiray@tubitak.gov.tr, [2]altingovde@ceng.metu.edu.tr

Abstract

Search result diversification is one of the key techniques to cope with the ambiguous and/or underspecified information needs of the web users. In the last few years, strategies that are based on the explicit knowledge of query aspects emerged as highly effective ways of diversifying the search results. Our contributions in this work are two-fold. First, we extensively evaluate the performance of a state-of-the-art explicit diversification strategy and pin-point its weaknesses. We propose basic yet novel optimizations to remedy these weaknesses and boost the performance of this algorithm. As a second contribution, inspired from the success of the current diversification strategies that exploit the relevance of the candidate documents to individual query aspects, we cast the diversification problem to the problem of ranking aggregation. To this end, we propose to materialize the re-rankings of the candidate documents for each query aspect and then merge these rankings by adapting the score(-based) and rank(-based) aggregation methods. Our extensive experimental evaluations show that certain ranking aggregation methods are superior to the existing explicit diversification strategies in terms of the diversification effectiveness. Furthermore, these ranking aggregation methods have lower computational complexity than the state-of-the-art diversification strategies.

Keywords: Search result diversification, explicit aspect modeling, ranking aggregation, score normalization

Introduction

Search result diversification is a popular problem that receives attention from both academia and industry. At the heart of the problem lies the fact that a large fraction of web queries are vaguely specified and/or ambiguous, making it very hard (if not impossible) for a search system to figure out the underlying search intent of the users. For such queries, it seems to be a good compromise to provide a result set that can cover possible different interpretations of the query and, thus, try to minimize the risks of disappointing the users (e.g., (Zhai & Lafferty, 2006)).

A number of result diversification strategies in the literature assume that potential query aspects can be explicitly identified (say, by categorizing the queries according to a taxonomy (Agrawal, Gollapudi, Halverson, & Ieong, 2009) or mining query logs (Santos, Macdonald, & Ounis, 2010; Capannini, Nardini, Perego, & Silvestri, 2011)), and aim to diversify the initial retrieval results (candidate documents) of a query based on these already known aspects. In this study, we also assume the availability of explicit query aspects and propose new strategies for result diversification in this setup.

Our contributions in this paper are two-fold. First, we extensively evaluate the performance of a state-of-the-art explicit diversification strategy, namely, xQuAD (Santos et al., 2010), and pin-point some of its weaknesses. In particular, we identify two issues, so-called “aspect elimination problem" and “aspect fading problem", that might arise due to the ways the relevance and novelty probabilities are computed and/or estimated in xQuAD. In essence, both of these problems are related to having some query aspects that end up with a negligible or no impact during the early stages of the diversification process; i.e., after selecting a few documents into the final result set.

To remedy the former problem, we explore a variety of relevance score normalization methods and also propose a normalization strategy based on the upper-bound score estimated

for a given query and retrieval model. To address the second problem, we propose to employ more stable functions while computing the novelty component of xQuAD.

Our second contribution is motivated by the following observation. We realize that computing the relevance of the candidate documents to each query aspect plays a central role in the success of the current explicit diversification strategies, such as xQuAD. Encouraged by this finding, we propose to materialize the re-rankings of the candidate documents for each query aspect and then merge them by adapting the score(-based) and rank(-based) aggregation methods that are widely applied in the meta-search scenario. In other words, we cast the diversification problem to the problem of merging/aggregating the re-rankings per query aspect. We hypothesize that if each of these re-rankings can place the most relevant documents for their respective aspects in their top-k results, then the aggregation of these rankings would be both relevant and diverse in terms of the coverage of these aspects, in a natural way.

To the best of our knowledge, we are the first to propose to model and solve the result diversification problem using the score and rank aggregation methods. For the purposes of score aggregation, we adapt two traditional methods, namely, CombSUM and CombMNZ (Fox & Shaw, 1994; Lee, 1997), and investigate their performance employing various score normalization techniques. For the rank aggregation, we adapt the classical methods like simple voting and Borda voting (Borda, 1781) as well as the Markov chain based approaches (Dwork, Kumar, Naor, & Sivakumar, 2001). Obviously, unlike the score aggregation techniques and the other explicit diversification strategies in the literature, these rank aggregation techniques do not require any score normalization stage.

A major difference of the diversification problem from the result merging problem is that instead of combining the results obtained for the same query from *different resources* involving potentially *different retrieval models*, we need to combine the re-rankings that are

generated with the *same retrieval model* over the *same candidate set* for different yet related query aspects. This is an important difference that can be exploited to improve the performance of the score aggregation methods, and we show that the normalization strategy proposed for xQuAD also serves well for this purpose. Another crucial difference in the diversification scenario is the requirement of balancing the relevance, which implies favoring the initial ranking, and diversity, which implies favoring the re-rankings for each query aspect. Therefore, we extend the score and rank aggregation methods by weighting the initial ranking and aspect rankings within the classical probability mixture framework of the diversification approaches.

We evaluate the performance of the xQuAD variants and ranking aggregation methods using the standard TREC datasets and explicit aspects discovered from different sources, and report the results for a number of well-known metrics. We compare the proposed methods to three state-of-the-art explicit diversification strategies, namely IA-Select (Agrawal et al., 2009), xQuAD (as originally proposed in (Santos et al., 2010)), and PM2 strategy in (Dang & Croft, 2012). Our experiments show that the xQuAD variants with the new score normalization and novelty components outperform the original algorithm as well as the other baselines. We further find that, for various parameter configurations and evaluation metrics, certain ranking aggregation methods as adapted here are also superior to all of the baseline strategies. This is a remarkable finding as these ranking aggregation methods can be computed more efficiently than the baseline diversification strategies and our xQuAD variants.

The rest of the paper is organized as follows. In the next section, we provide an overview of the related studies in the literature. We identify two potential weaknesses of a state-of-the-art explicit diversification framework, xQuAD, and introduce our solutions in the section xQuAD Framework: Potential Weaknesses and Extensions. In the section Ranking

Aggregation Methods for Diversification, we tackle the result diversification problem from a rank aggregation perspective and adapt a number of score and rank aggregation methods from the literature. In the next two sections, we describe our experimental setup and present the evaluation results, respectively. The last section provides the conclusion and points to future research directions.

Related Work

Search Result Diversification

Generating diverse/novel results is a hot topic with the potential of application in various contexts, ranging from web search engines (e.g., (Santos, Castells, Altingovde, & Can, 2013))) to recommenders (e.g., (Vargas & Castells, 2011)) and topic tracking systems (e.g., (Aksoy, Can, & Kocberber, 2012)). In this paper, we focus on the search result diversification problem that aims to provide both relevant and diverse results for the ambiguous or underspecified web queries. In the literature, the approaches that address this problem are broadly categorized as either *implicit* or *explicit* (Santos et al., 2010).

Implicit Search Result Diversification. The strategies in this category assume no prior knowledge of the query aspects; so they either exploit the inter-similarity of the documents in the candidate set or attempt to discover the underlying query aspects in an unsupervised manner (Santos, 2013). A pioneering example of the former approach is the Maximum Marginal Relevance (MMR) strategy that constructs the final ranking in a greedy manner (Carbonell & Goldstein, 1998). In each iteration, a document's score is computed by the difference of its relevance to the original query and similarity to the documents that are selected into the final ranking up to this point; and the document with the highest score is selected. Various strategies in the literature adapt this greedy algorithm, yet differ in the way they compute the inter-document similarities. For instance, Zhai et al. (Zhai, Cohen, & Lafferty, 2003) utilize unigram language models for representing the individual documents as

well as the set of documents that are already selected into the final ranking at any point during the greedy iterations. In contrast, Zuccon and Azzopardi (2010) make use of the quantum probability ranking principle while modeling the interference among the ranked documents. Two independent works in the literature propose to adapt the modern portfolio theory to the result diversification problem (Rafiei, Bharat, & Shukla, 2010; Wang & Zhu, 2009). In this case, the inter-document similarities are modeled based on the variance of the relevance among the ranked documents.

Gollapudi and Sharma (2009) identify the connection between the result diversification problem and facility dispersion optimization problems, and adapt some approximate solutions (namely, Max-Sum and Max-Min algorithms) from the operations research field to the diversification context. Minack et al. employ these algorithms and improve their efficiency for diversifying continuous data streams (Minack, Siberski, & Nejdl, 2011). A comparative analysis of various implicit diversification algorithms using five different datasets (other than standard TREC collections) is provided by Vieira et al. (2011). More recently, Zuccon et al. introduce an alternative perspective and model the diversification problem within the desirable facility placement (DES) framework (Zuccon, Azzopardi, Zhang, & Wang, 2012).

Different from the above approaches, some other implicit diversification strategies (so-called coverage based methods in (Santos, 2013)) attempt to model the underlying query aspect from the initial retrieval results. For instance, Carterette and Chandar (2009) identify the aspects (facets) using relevance modeling and topic models, and then constructs the final ranking in a round-robin fashion, i.e., by choosing the best document for each facet. He et al. (He, Meij, & de Rijke, 2011) also use topic models to partition the candidate documents into clusters; but they only consider the most relevant clusters to the query for the subsequent diversification stages where well-known strategies such as the MMR and round-robin are applied.

Explicit Search Result Diversification. In the explicit diversification methods, query aspects are modeled explicitly, i.e., by exploiting the query labels, which are assigned either manually or automatically, or from the reformulations of the query. IA-Select approach adopts the former option and assumes that both queries and documents are associated with some categories from a taxonomy (Agrawal et al., 2009). The diversification is achieved by favoring documents from different categories and penalizing the documents that fall into already covered categories. Alternatively, Radlinski and Dumais (2006) use a given query and its reformulations to obtain a candidate result set; which is then re-ranked and personalized for a given user. Capannini et al. (2011) employ query logs to decide when/how query results should be diversified, and propose a new algorithm based on the popularity of query reformulations in the log.

xQuAD is one of the most effective diversification strategies that also exploit query reformulations obtained from TREC subtopics and search engines to model the query aspects (Santos et al., 2010). In a follow-up work, Santos et al. (Santos, Macdonald, & Ounis, 2011) employ both xQuAD and IA-Select to achieve result diversification for the queries with navigational, informational, or transactional intents. Vallet and Castells (2012) incorporate a personalization component into both of the latter algorithms by explicitly introducing the user as a random variable. In another study, Vargas et al. again employ these two algorithms, xQuAD and IA-Select, and propose to model their relevance models explicitly, i.e., using the relevance judgments or, more practically, click statistics (Vargas, Castells, & Vallet, 2012). Finally, Zheng et al. propose a coverage based diversification framework where they experiment with several coverage functions (Zheng, Wang, Fang, & Cheng, 2012). While these latter works also improve or build on xQuAD, none of them focus on its components in a way similar to ours. Different from the previous studies, we propose optimizations for the relevance score normalization and novelty estimation components of xQuAD.

Ranking Aggregation

A common use of ranking aggregation (a.k.a. ranking fusion, result merging/fusion) methods in the real life is the election systems that allow voters to rank the candidates in the order of preference[1]. In computer science, these methods are widely investigated for and applied to research problems, such as meta-search (Aslam & Montague, 2001; Dwork et al., 2001; Renda & Straccia, 2003), spam detection (Dwork et al., 2001), word association (Dwork et al., 2001) and result generation from search engine caches (Cambazoglu, Altingovde, Ozcan, & Ulusoy, 2012). However, to the best of our knowledge, no previous study proposes to adapt such methods for the result diversification task (We discuss the details of these methods in the section Ranking Aggregation Methods for Diversification.).

Note that, while the proportionality framework of Dang and Croft (2012) also has its roots in the voting systems; their approach is different than ours. More specifically, their diversification strategies are based on the votes given to the *aspects* whereas here we focus on the votes given to the *documents* by each aspect. We employ the best-performing strategy (PM2) reported in (Dang & Croft, 2012) among our baseline diversification strategies in the experimental evaluations.

The problem of score normalization is often tackled in the context of score-based ranking aggregation. For instance, Fernandez et al. propose a probabilistic normalization strategy for score-based aggregation (Fernandez, Vallet, & Castells, 2006). Arampatzis and Kamps (2009) propose a normalization approach based on the assumption that the retrieval scores are composed of a signal and a noise component. In a rather different context, Ravana and Moffat (2009) investigate the score aggregation techniques for summarizing the performance of a retrieval system over a set of queries. To the best of our knowledge, none of the previous studies explore the impact of score normalization on the explicit result diversification.

[1] http://en.wikipedia.org/wiki/Voting_system

xQuAD Framework: Potential Weaknesses and Extensions

Preliminaries

Assume a query q is processed over a collection C and retrieves a ranked list of documents τ_q, where $|\tau_q|=N$.

Result Diversification Problem: Construct a ranked list τ_q^* of k documents $(k < N)$ such that τ_q^* maximizes both the relevance and diversity among all possible rankings $\tau_i (|\tau_i| = k)$ of τ_q.

A particular case of this general problem is the explicit result diversification problem, where there is a set of explicitly identified query aspects (a.k.a., sub-topics, interpretations, sub-queries) denoted as $T = \{q_1,...,q_m\}$ associated with the original query q. Then, the objective function is finding a top-k ranking τ_q^* that maximizes the overall relevance to multiple query aspects and at the same time, minimizes its redundancy with respect to these aspects (Gollapudi & Sharma, 2009).

It can be shown that the general form of this problem is an instance of the maximum coverage problem and thus, it is NP-hard (e.g., see (Santos et al., 2010)). A large number of diversification strategies based on the approximation algorithms, heuristics and/or meta-heuristics are proposed in the literature (as briefly reviewed in the previous section). In what follows, we describe one of the most effective strategies, xQuAD, that is investigated and extended in more depth in the following sections.

xQuAD Framework

xQuAD is a probabilistic framework (Santos et al., 2010) that constructs the ranking τ_q^* in a greedy manner, by choosing the document $d_i \in \tau_q$ that maximizes the following probability mixture model at each iteration:

$$(1-\lambda)P(d \mid q) + \lambda \sum_{q_i \in T} P(q_i \mid q) P(d \mid q_i) P(\bar{\tau}_q^* \mid q_i), \quad (1)$$

where $P(d \mid q)$ denotes the relevance (i.e., likelihood of observing d for the query q) whereas the summation captures the diversity. In particular, $P(q_i \mid q)$ denotes the likelihood of the aspect (sub-query) q_i for the query q (referred to as sub-query importance in (Santos et al., 2010)), $P(d \mid q_i)$ is the likelihood of observing d for the aspect q_i and finally $P(\bar{\tau}_q^* \mid q_i)$ denotes the probability of q_i not being satisfied by the documents that are already in τ_q^*. The latter probability, which indeed captures the novelty, can be represented as the product of the probabilities of each document in τ_q^* for not satisfying q_i:

$$P(\bar{\tau}_q^* \mid q_i) = \prod_{d_j \in \tau_q^*} (1 - P(d_j \mid q_i)). \quad (2)$$

Potential Weaknesses of xQuAD

xQuAD is one of the most successful strategies for the explicit result diversification and placed among the top-performers in the diversity tasks of both TREC 2009 and 2010 (Clarke, Craswell, & Soboroff, 2009; Clarke, Craswell, Soboroff, & Cormack, 2010). However, we still identify two problems that can significantly diminish the performance of xQuAD, as follows.

Aspect elimination problem. In the above model, a key component is the relevance computation of a document d to the query q and its aspects (sub-queries) q_i, denoted as $P(d \mid q)$ and $P(d \mid q_i)$, respectively. In previous works, these probabilities are usually based on the popular weighting models like BM25, language models, etc. (e.g., (Santos et al., 2010)). Typically, the scores produced by these methods are normalized to [0,1] range at the query-level, so that they can be employed in the xQuAD's mixture model. While no details are provided on the exact procedure employed in previous works, a practical and tempting

approach is using the MinMax score normalization, where the score range of a query is mapped to the range [0,1]; i.e., the top-ranked document in a list having the score 1. MinMax normalization can be formally expressed as (Renda & Straccia, 2003):

$$P(d \mid q) = \frac{s(d,q) - \min_{d_i \in \tau_q} s(d_i,q)}{\max_{d_i \in \tau_q} s(d_i,q) - \min_{d_i \in \tau_q} s(d_i,q)}, \quad (3)$$

where τ_q is the ranked retrieval result for q, $s(d,q)$ is the score generated by the retrieval model and $P(d \mid q)$ is the normalized relevance probability.

However, we realize that MinMax and other normalization techniques that set the $P(d \mid q)$ (or, $P(d \mid q_i)$) value to 1 for the highest scoring documents for q (or, q_i) cause a deficiency in the model. Once the top-scoring document d^* for an aspect q_i is selected for τ_q^*, for all following iterations, the impact of covering this aspect will be nullified. That is, as $P(d^* \mid q_i)$=1 using, say, MinMax normalization, the probability $\prod_{d_j \in \tau_q^*}(1 - P(d_j \mid q_i))$ will be 0 once d^* is selected for τ_q^*. Therefore, the algorithm will not care covering aspect q_i from this point on. Even worse, for a query with just a few aspects, if the documents with the highest scores for each aspect are selected at the early stages of the algorithm, then diversification part of the xQuAD will be totally neglected, and all remaining documents will be selected solely based on $P(d \mid q)$.

The problem is more emphasized for the queries with a small number of aspects and when the diversified set size is relatively large; i.e., $k \geq 20$. In Figure 1, we show the number of *eliminated* aspects after choosing the documents for each rank position i $(1 \leq i \leq 20)$ using xQuAD on TREC 2009 diversity task setup for the λ that yields the highest α-nDCG@20 score (see the section Experimental Setup for the details). The figure shows that even after

selecting the first two documents into τ_q^*, 23% of the query aspects (i.e., 56 out of 241 aspects specified for the 50 topics in TREC 2009) are neglected, which is clearly not helpful for the diversification purposes.

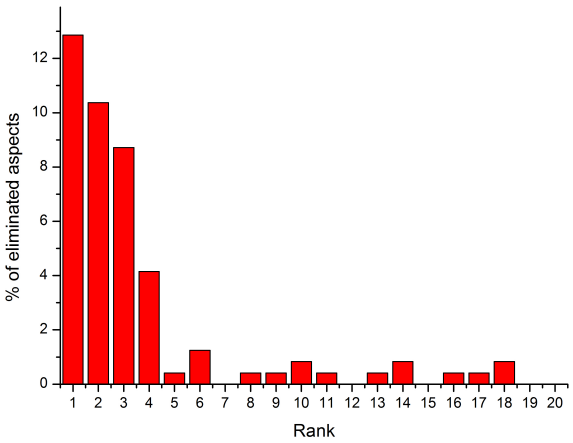

FIG 1. Percentage of the eliminated aspects after choosing the documents for each rank in τ_q^*.

Finally, the aspect elimination problem can be further harmful for the informational queries, for which the users usually need more than one document (per aspect) to satisfy their information needs. Within this latter context, Welch et al. (2011) report the existence of the aspect elimination problem for another diversification strategy, namely, IA-Select (Agrawal et al., 2009). Note that, since the IA-Select strategy in its original setup employs the scores obtained from a classifier, the problem in their case is not directly related to the normalization techniques. Nevertheless, in this paper, we include IA-Select among our baseline strategies (replacing the classifier scores with $P(d \mid q_i)$ scores as in (Santos et al., 2010)), and evaluate the impact of the relevance score normalization techniques (described in the next section) also for IA-Select.

Aspect fading problem. Even when the top-scoring document of an aspect is not selected for τ_q^*, the impact of the aspect q_i fades away after choosing, say, a couple of documents with high $P(d \mid q_i)$ values; as the novelty component is based on the product of $(1-P(d \mid q_i))$

scores. For instance, if only two documents with 0.9 coverage probability of the aspect q_1 are in τ_q^*, for all the remaining documents, their $P(d \mid q_1)$ scores will be multiplied with 0.01, rendering this aspect practically useless[2]. Furthermore, for queries with small number of aspects and after selection of a few documents into τ_q^*, this may yield very small numerical values for the document novelty scores of the remaining documents; and again, the remaining documents would be selected essentially based on the relevance scores $P(q \mid d)$. In the following sub-sections we discuss solution methods for each of these problems.

Relevance Score Normalization for xQuAD

While the problem of retrieval score normalization is investigated on its own in previous works and especially in the context of rank aggregation in meta-search (Renda & Straccia, 2003; Fernandez et al., 2006), to the best of our knowledge, its impact on the result diversification is not yet addressed[3]. To remedy the aspect elimination problem discussed in the previous section, a straightforward solution can be using a normalization that does not map the top-ranked document relevance to 1 for a given list τ. To this end, a practical approach is using Sum normalization, defined as follows (Fernandez et al., 2006):

$$P(d \mid q) = \frac{s(d,q)}{\sum_{d_i \in \tau_q} s(d_i,q)} \tag{4}$$

Our problem at hand is different than the traditional rank aggregation problem for meta-search engines in that the diversification is usually applied by the party that actually generates

[2] As the astute reader would notice, the λ parameter can help to remedy the situation if the numerical differences are small; but it is still useless when the relevance and diversity scores vary in the order of magnitudes.

[3] Note that, Vargas et al. (2012) recently proposed using the number of clicks instead of the retrieval scores for estimating the relevance probabilities. This is a viable though orthogonal approach to what we propose here.

the initial retrieval scores for τ_q; i.e., the system does not only know the scores but also knows how they are computed. Exploiting this information, we propose an alternative normalization based on the highest possible score that can be generated for a given query and retrieval model. In this paper, we employ two weighting models for initial retrieval, namely, a variant of Okapi-BM25 (Robertson, Walker, Jones, Hancock-Beaulieu, & Gatford, 1994) and the query-likelihood language model with Dirichlet smoothing (Zhai & Lafferty, 2004) as implemented in the Zettair text retrieval system (Zettair, 2006).

For each retrieval model, we define a virtual best score that would be generated by a virtual document that is supposed to include each query term in the document with the frequency of the document length, i.e., as if the document is only composed of the query terms[4]. We set this virtual document's length to the average document length in the collection. While this is an unrealistically high upper-bound, our experiments reveal that it serves quite well for the purposes of this paper. Therefore, we normalize the scores in τ_q by dividing each score by the virtual best score obtained for q using the same retrieval function that generated τ_q. Note that, the same procedure is also applied while normalizing the scores for $P(d \mid q_i)$. We call this normalization Virtual:

$$P(d \mid q) = \frac{s(d,q)}{s(d^V,q)} \tag{5}$$

where $s(d^V,q)$ is the upper-bound score computed for the virtual best document d^V.

Document Novelty Estimation for xQuAD

As discussed above, the aspect fading problem arises as xQuAD computes the novelty of a document d for an aspect q_i by multiplying the dissatisfaction probability of q_i by the

[4] This is similar to computing an upper-bound for the relevance scores in dynamic pruning strategies, e.g., see (Macdonald, Ounis, & Tonellotto, 2011).

documents in the current set τ_q^*, as follows:

$$P(\bar{\tau}_q^* \mid q_i) = \prod_{d_j \in \tau_q^*} (1 - P(d_j \mid q_i)).$$

To avoid the negligible document novelty estimations (in comparison to the relevance scores), instead of taking the product of probabilities in $P(\bar{\tau}_q^* \mid q_i)$, we propose to use either arithmetic mean or geometric mean of the aspect dissatisfaction probabilities (as shown in Equations 6 and 7, respectively). This is a simple yet effective optimization to make the relevance and diversity sides of the mixture model comparable to each other in terms of their numerical values. Furthermore, by this optimization, λ can be determined more accurately among various queries, as it would serve only as a trade-off parameter as intended, but not for the purposes of remedying the gap between the numerical scores.

$$P(\bar{\tau}_q^* \mid q_i) = \frac{\sum_{d_j \in \tau_q^*} (1 - P(d_j \mid q_i))}{|\tau_q^*|} \tag{6}$$

$$P(\bar{\tau}_q^* \mid q_i) = \sqrt[|\tau_q^*|]{\prod_{d_j \in \tau_q^*} (1 - P(d_j \mid q_i))} \tag{7}$$

The xQuAD versions that employ the arithmetic and geometric means of the probabilities in the novelty estimation component are referred to as art_xQuAD and geo_xQuAD in the rest of this paper.

Ranking Aggregation Methods for Diversification

A key component of the xQuAD framework discussed in the previous section is $P(d \mid q_i)$, i.e., the likelihood of observing d for the aspect q_i (see Equations 1 and 2). In practice, this component computes the relevance of candidate documents to each query aspect using a retrieval model. Indeed, such a computation is not only involved in xQuAD, but also included in two other competing strategies, namely, IA-Select (Agrawal et al., 2009) and PM2 (Dang

& Croft, 2012). Encouraged by the success of all these explicit diversification strategies demonstrated in the earlier works, we propose an alternative perspective to exploit this key component.

In this paper, we materialize the *re-rankings* of the candidate documents for each query aspect and then tackle the result diversification problem from a *ranking aggregation* perspective. In the classical ranking aggregation context, the goal is producing a merged list τ from the given full or partial rankings $\{\tau_1,...,\tau_m\}$ so that the final list τ has the minimal distance from each individual list τ_i. In our case, for a given query q with the set of aspects $T = \{q_1,...,q_m\}$ and initial retrieval result $\tau_q(|\tau_q| = N)$, let's assume that τ_{q_i} denotes the re-ranking of the documents in τ_q with respect to the relevance probabilities $P(d \mid q_i)$ for the aspect q_i, and $\tau_{q_i}^k$ denotes the top-k documents in τ_{q_i}. We hypothesize that if each ranking τ_{q_i} places the most relevant documents higher for the corresponding aspect q_i, then the aggregation of these top-k rankings would be both relevant and diverse; i.e., cover as many diverse aspects as possible.

In the context of ranking aggregation described above, it is tempting to optimize the Kendall tau distance, which counts the number of pairwise disagreements between two lists, as a typical measure of distance between two rankings. However, Dwork et al. (2001) show that computing the aggregation that optimizes the Kendall distance, so-called *Kemeny optimal aggregation*, is NP-hard even for four different rankings. Fortunately, there are various sub-optimal methods that are shown to serve well in real life applications, such as building meta-search engines and combating spam results (see the section Related Work for other examples). Such ranking aggregation methods in the literature are categorized based on the type of information used during the fusion process. Score-based aggregation methods exploit the relevance scores associated with each document in each ranking, whereas rank-based

aggregation methods only rely on the document's position in the list. In the rest of this section, we adapt a number of representative methods from each category for the purposes of result diversification.

An important difference of our problem from the rank aggregation in meta-search is that in our setup, there exists an initial ranking τ_q, and all τ_{q_i} lists are basically re-rankings of the former. In the ranking aggregation methods employed in this paper, we exploit both τ_q and τ_{q_i} rankings to generate the final diversified ranking τ_q^*. To emphasize this mixture of the initial and aspect rankings, the abbreviations of the method names are prefixed with mix in the following discussions.

Score-based Aggregation Methods

One of the well-known approaches for ranking aggregation in the context of meta-search is combining the normalized relevance scores with various functions, such as min, max, median and sum (Fox & Shaw, 1994; Lee, 1997). Among these variants, CombSUM and CombMNZ are the most effective ones that are widely employed in the subsequent works (e.g., (Renda & Straccia, 2003; Aslam & Montague, 2001)).

CombSUM (mix_CombSUM). This method computes the overall score of d for the query q by simply adding up the document's scores in each ranking τ_{q_i}. For the purposes of diversification, we also incorporate the initial ranking τ_q using a mixture model as typical in all diversification frameworks and come up with the following formula:

$$S(q,d) = (1-\lambda)P(d \mid q) + \lambda \sum_{q_i \in T} P(q_i \mid q)P(d \mid q_i), \quad (8)$$

where $P(q_i \mid q)$ denotes the aspect likelihood that is typically included in most of the explicit diversification strategies. A similar notion of associating priorities to the rankings has also

been employed for the score aggregation methods in the meta-search context (Renda & Straccia, 2003). The final ranking τ_q^* includes the top-k documents (computed using a min-heap) in descending order of $S(q,d)$ values (ties are broken randomly).

Notice that the formula is indeed quite similar to that of xQuAD (and IA-Select method defined in (Agrawal et al., 2009)) with one crucial difference: the latter strategy constructs the final ranking in a greedy manner and takes into account the novelty with respect to the documents that are already selected in τ_q^* while computing the score $S(q,d)$. In contrast, mix_CombSUM applies a linear weighted summation of the scores for every aspect as well as the initial results, which can be cheaper in terms of the computational efficiency. In particular, mix_CombSUM needs to make a single pass over the candidate documents, which implies a complexity of $O(N \log k)$ using a min-heap of size k to create the final ranking (e.g., see (Witten, Moffat, & Bell, 1999)). In contrast, since xQuAD compares every candidate document to those already selected into the τ_q^* for each iteration, its overall complexity is $O(Nk)$ (Capannini et al. (2011)). Therefore, mix_CombSUM is more efficient than xQuAD, as well as the other diversification baselines that make similar kind of comparisons, such as IA-Select and PM2 (see the section Experimental Setup for the details of the latter strategies).

CombMNZ (mix_CombMNZ). This method is similar to the previous one, but the score of d is weighted by the sum of the votes for d given by each $\tau_{q_i}^k$, as follows:

$$S(q,d) = (1-\lambda)P(d \mid q) + \lambda \sum_{q_i \in T} v(d, \tau_{q_i}^k) \sum_{q_i \in T} P(q_i \mid q) P(d \mid q_i). \qquad (9)$$

In Equation 9, $v(d, \tau_{q_i}^k)$ denotes the number of rankings $\tau_{q_i}^k$ where d appears, and it is computed as

$$v(d,\tau_{q_i}^k)=\begin{cases}1, & \text{if } d\in \tau_{q_i}^k,\\ 0, & \text{otherwise.}\end{cases} \quad (10)$$

Similar to the mix_CombSUM method, the final ranking τ_q^* includes the top-k documents (computed using a min-heap) in descending order of $S(q,d)$ values (ties are broken randomly).

Note that, the relevance probabilities $P(d \mid q)$ and $P(d \mid q_i)$ in Equations 8 and 9 should be normalized, as in the case of xQuAD. As we mention before, the diversification scenario allows us to employ the Virtual technique that makes use of the actual retrieval model and the collection statistics, in addition to the traditional MinMax and Sum normalization schemes. In our experimental evaluations, we consider all three normalization techniques along with the mix_CombSUM and mix_CombMNZ techniques.

Rank-based Aggregation Methods

In rank(-based) aggregation methods, the relevance scores are not taken into account and the final ranking is obtained by only using the order of documents in each aspect ranking.

Simple voting (mix_SV). In this method (e.g., see (Cambazoglu et al., 2012)), we assume that each document $d \in \tau_q$ receives a vote from each ranking[5] $\tau_{q_i}^k$ weighted with the aspect likelihood $P(q_i \mid q)$; i.e, the vote count per document is computed as:

$$C(q,d) = (1-\lambda)v(d,\tau_q^k) + \lambda \sum_{q_i \in T} P(q_i \mid q)v(d,\tau_{q_i}^k) \quad (11)$$

where vote $v(d,\tau_{q_i}^k)$ is computed as in Equation 10 and τ_q^k denotes the top-k documents of

[5] We consider only $\tau_{q_i}^k$ lists for this method, as using τ_{q_i}'s would result in the same vote count for all the documents.

the initial ranking τ_q.

The final ranking τ_q^* includes the top-k documents (computed using a min-heap) in descending order with respect to the vote counts $C(q,d)$ (ties are broken using $P(d \mid q)$ values).

Borda voting (mix_BV). This is based on Borda's classical method (Borda, 1781) that also takes the position of the documents in the ranked lists into account while computing the vote counts, as follows:

$$C(q,d) = (1-\lambda)\tau_q(d) + \lambda \sum_{q_i \in T} P(q_i \mid q)\tau_{q_i}(d) \tag{12}$$

where $\tau_q(d)$ is the rank position of d in some list τ_q. The final ranking τ_q^* is constructed in ascending order with respect to the vote count (again, ties are broken using $P(d \mid q)$ values).

As we discuss for the mix_CombSUM method, the worst-case complexity of mix_BV is also $O(N \log k)$.

Markov chain based methods. Dwork et al. (2001) have proposed using Markov chains for aggregating ranked partial lists and described four different variants. In what follows we discuss this approach using our own problem setup and notation, please refer to (Dwork et al., 2001) for basics and adaptation to the general ranking aggregation problem.

For this case, we define the document space U as the union of the documents in τ_q^k as well as the all top-*k* re-rankings per aspect ($\tau_{q_i}^k$), as follows:

$$U = \bigcup_{q_i \in T} \tau_{q_i}^k \bigcup \tau_q^k \tag{13}$$

Note that, the document space is limited to top-k documents from each ranked list, as otherwise the number of states and size of the transition matrix would be too large for on-the-fly-computation of the diversified results. In this approach, each document $d \in U$ is

considered as a state in the Markov chain. A non-negative stochastic matrix M (of size $|U| \times |U|$) defines the probability of the systems' transitions from one state to another. In our case, these probabilities are based on the positions of the documents in various ranked lists. Once the system starts on some state probability distribution (typically, the uniform distribution), it eventually reaches to a unique fixed point where the state distribution does not change. This is called the stationary distribution and for our purposes, the stationary probabilities of the states at this point are used to sort the documents (states) and obtain the final τ_q^*.

Dwork et al. (2001) define four different Markov chains by describing four different ways of constructing the transition matrix, as follows:

- *MC1*: If the current state (document) is d_i, the next state is chosen uniformly from the multiset of all documents d_j such that both d_i and d_j appear in some ranking τ and d_j is ranked higher; i.e., $\tau(d_j) \leq \tau(d_i)$.
- *MC2*: If the current state (document) is d_i, then first pick a ranking τ uniformly from all rankings that include d_i, and then choose a document d_j uniformly that is ranked higher than d_i in τ, i.e., $\tau(d_j) \leq \tau(d_i)$.
- *MC3*: If the current state (document) is d_i, then first pick a ranking τ uniformly from all rankings that include d_i, and then choose a document d_j uniformly from τ. If $\tau(d_j) < \tau(d_i)$ then go to d_j else stay in the state d_i.
- *MC4*: If the current state (document) is d_i, then first pick a document d_j uniformly from U. If $\tau(d_j) < \tau(d_i)$ for the majority of the lists τ that ranked both d_i and d_j, then go to d_j, else stay in the state d_i.

Some nice theoretical intuitions for constructing these particular Markov chains are provided in (Dwork et al., 2001), and a set of example transition matrices for the ranking aggregation in meta-search scenario is given in (Renda & Straccia, 2003). Following the practice in the latter work, we computed the stationary distribution using the simple power-iteration method. That is, we start the iteration by a state vector where each state has $1/|U|$ probability and repetitively multiply it with the transition matrix M till the state probabilities are stabilized, i.e., converge to the stationary distribution.

The computational complexity of computing the stationary distribution is $O(|U|)$, as shown by Dwork et al. (2001). Given that the input top-*k* rankings per aspect ($\tau_{q_i}^k$) (see Equation 13) can be constructed in $O(N \log k)$ time (using a min-heap of size *k*), the overall complexity becomes $O(|U|+|T|N\log k) \approx O(|T|N\log k)$, where $|T|$ denotes the number of aspects. This would be still better than the diversification baselines (i.e., xQuAD, IA-Select and PM2), which have the complexity of $O(Nk)$, when there exists a few aspects per query and $k \geq 20$ as typical in practice.

Note that, as we use both the initial ranking τ_q and aspect rankings τ_{q_i} while constructing the document space U, we again prefix the names of these methods with mix, hereafter.

Experimental Setup

Collection, Queries and Aspects

We use the standard framework of “Diversity Task" as described in the TREC Web Track. In particular, we employ ClueWeb09 collection Part-B that includes around 50 million English web documents. The collection is initially parsed and indexed using the publicly available Zettair IR system (Zettair, 2006). During the indexing, Zettair is executed with the “no stemming" option, yielding a vocabulary of 163,629,158 terms.

We report our results for TREC 2009 and 2010 topic sets that include 50 and 48 query topics, respectively[6]. For each topic in these sets, a number of sub-topics (up to 8) are described and the relevance judgments are provided at the sub-topic level. In the following experiments, we generate the query aspects in two ways. First, following the common practice in the previous works (e.g., (Dang & Croft, 2012; Santos et al., 2010)), we use the "query" field of each topic as the initial query and generate its aspects (sub-queries) using the official sub-topic descriptions provided in the TREC topic sets. This case represents the idealistic scenario with the perfect knowledge of the query aspects. Secondly, we simulate a more realistic scenario and use top-10 query suggestions (auto-completions) collected from Google search engine as the aspects of each query, as first proposed in (Santos et al., 2010).

Initial Retrieval Model

For the initial retrieval runs, we used our homemade IR system with two popular retrieval models, namely, a variant of the well-known Okapi BM25 metric (Robertson et al., 1994) and the query-likelihood language model with Dirichlet smoothing (Zhai & Lafferty, 2004). For BM25 we set k_1 to 1.2 and b to 0.50, and for the language model (LM) we set $\mu = 2000$.

We first retrieve top-N candidate documents (τ_q) using one of these weighting models, and then run the diversification strategies to obtain the final top-k results, i.e, τ_q^*. Unless stated otherwise, for all the experiments we set N=100 and k=20. During retrieval, standard stopwords are removed.

Previous studies that experimented with the ClueWeb09 collection report that applying spam filtering can considerably improve the initial retrieval performance. Therefore, we also employ the spam filtering technique in (Cormack, Smucker, & Clarke, 2011). In particular,

[6] Note that, we prefer to report evaluations separately on each topic set (but not their union) for the sake of comparability with the previous works.

we utilize the publicly available Waterloo Spam Rankings[7] that assigns a spam percentile score to each document in the ClueWeb09 collection. During the initial retrieval, we set the relevance scores of the documents with a spam score of 60 or higher to $-\infty$ (as in (Dang & Croft, 2012)), so that these documents are eliminated from the top-N candidate documents.

Baseline Diversification Strategies and Evaluation Metrics

We have three strategies that serve as the diversification baselines. All of these strategies are greedy in nature and differ in the scoring function that is used to select the best document at each iteration, until all k documents are selected into τ_q^*. We briefly summarize these strategies as follows:

Intent Aware (IA)-select. This strategy aims to choose the document with the highest probability of satisfying the user given that all previously selected ones fail to do so (Agrawal et al., 2009). The scoring function of IA-Select is as follows:

$$\sum_{q_i \in T} P(q_i \mid q) V(d \mid q, q_i) \prod_{d_j \in \tau_q^*} (1 - V(d_j \mid q, q_i)). \quad (14)$$

where $V(d \mid q, q_i)$ is the likelihood of d satisfying q for the underlying aspect q_i. As there is no strict enforcement on the implementation of this latter component in (Agrawal et al., 2009), it is replaced by $P(d \mid q_i)$ in our experiments (as in (Santos et al., 2010)).

org_xQuAD. This is the original xQuAD algorithm (Santos et al., 2010) as elaborated in the previous sections. Its scoring function, which is basically the combination of Equations 1 and 2, is as follows:

$$(1-\lambda) P(d \mid q) + \lambda \sum_{q_i \in T} \left[P(q_i \mid q) P(d \mid q_i) \prod_{d_j \in \tau_q^*} (1 - P(d_j \mid q_i)) \right]. \quad (15)$$

[7] http://plg.uwaterloo.ca/ gvcormac/clueweb09spam/

PM2. In (Dang & Croft, 2012), two strategies, namely PM1 and PM2, are proposed within a proportionality-based diversification framework. The authors report that PM2 outperforms both its simpler predecessor PM1 and the original xQuAD for several evaluation metrics. Therefore, we include PM2 strategy as our third diversification baseline.

The intuition for this strategy is that, in a similar manner to allocation of seats to party representatives in some election systems, the ranks in τ_q^* should be allocated to documents that satisfy the query aspects in proportion to the popularity of these aspects in τ_q. At a given iteration p, first the *winner* aspect q_{i^*} is determined by the popularity of the aspect in τ_q and number of positions in τ_q^* that are allocated to this aspect up to iteration p (i.e., referred to as *quotient score*). Next, for this winner aspect q_{i^*}, PM2 selects the document d that maximizes the following score function:

$$\lambda \times qt[i^*] \times P(d \mid q_{i^*}) + (1-\lambda) \sum_{i \neq i^*} qt[i] P(d \mid q_i) \tag{16}$$

where $qt[i]$ is the quotient score and λ is the trade-off parameter between the relevance to the winner aspect and other aspects. Since the selected document in PM2 is expected to satisfy not only the winner aspect but also some other aspects, the number of positions allocated to each aspect is also updated accordingly (see (Dang & Croft, 2012) for details).

In all of these diversification baselines, we compute the relevance of the candidate documents to query aspects, i.e., $P(d \mid q_i)$, using the same model employed for the initial retrieval. While doing so, standard stopwords are removed from the aspect descriptions. Following the practice in (Santos et al., 2010), aspect probabilities $P(q_i \mid q)$ are computed uniformly as $1/|T|$, where T is the set of aspects $\{q_1,...,q_m\}$ for a given query q.

For the strategies xQuAD and PM2, we test all values of the trade-off parameter λ in [0,1] range with a step size of 0.01, and the best λ values obtained on one of the topic sets

(say, TREC 2009) is employed to obtain the reported results on the other topic set (say, TREC 2010).

Evaluation metrics. To evaluate the diversification performance, we compute most common measures, namely, α-nDCG, ERR-IA and Precision-IA, at the cut-off value of 20, using ndeval software[8]. For α-nDCG, α is typically set to 0.5, i.e., relevance and diversity are equally weighted.

Reproducibility of the results.

For search result diversification, a standard evaluation framework, namely "Diversity Task" in TREC Web Track, is available, which allows the use of a common dataset, queries and relevance judgments. Still, we identified some issues that complicate, or occasionally, make it impossible to make direct comparison of the results in different studies. First, even when the same document collection is employed (usually ClueWeb09 in the last years), the software used for indexing (e.g., Zettair, Terrier (e.g., (Santos et al., 2010)), Lemur/Indri (e.g., (Dang & Croft, 2012)), etc.) and choice of the parameters (list of stopwords, stemming options, handling various HTML tags during the parsing, spam filtering, etc.) can considerably alter the final results. Secondly, the retrieval models and their parameters can differ. A third issue that complicates comparing the results in our case is the list of query aspects. Even when the original TREC sub-topics are used for generating the aspects, there might be subtle differences in parsing the sub-topic descriptions. Obviously, if Web search engine suggestions are used to this end, the aspects employed by the works conducted at different times would differ significantly, making the results even less comparable.

In the light of above discussion, we provide the following data items to allow other researchers compare and contrast their findings with ours[9]. First, we provide the initial

[8] http://trec.nist.gov/data/web10.html

[9] www.ceng.metu.edu.tr/ altingovde/diversification/

retrieval results, i.e., top-100 document identifiers, obtained over the ClueWeb09 Part-B collection. This would allow researchers to start with the same basis, i.e., candidate document set, to apply their own diversification strategies. Secondly, we provide the list of query aspects generated for each topic using TREC sub-topics and search engine suggestions.

Evaluation Results

In this section, we seek answers to the following research questions:

- What is the impact of the score normalization techniques on the performance of the baseline diversification strategies, especially xQuAD and IA-Select that can suffer from the aspect elimination problem?
- Can the xQuAD variants with the new relevance normalization and novelty estimation components outperform the original xQuAD strategy and other baselines?
- Can the score and rank aggregation methods outperform the diversification baselines?

In the following experiments, we essentially report our results using the BM25 model for the initial retrieval stage and official TREC sub-topics for representing the query aspects. In the section Impact of the Components and Parameters, we provide additional experiments where we explore the impact of the alternative retrieval models and aspect representations.

Performance of the Score Normalization Techniques

We begin with comparing the performance of the baseline diversification strategies on TREC 2009 and 2010 topic sets and using the aspects obtained from the official sub-topics (Table 1). For each diversification strategy, we normalize the relevance scores using the MinMax and Sum methods from the literature (Fernandez et al., 2006), as well as the virtual best score (denoted as Virtual) as we describe in this paper. We also report the trade-off parameter λ employed in each case.

TABLE 1. Diversification performance w.r.t. the relevance normalization techniques using the query aspects obtained from the official sub-topics. The highest scores are shown in boldface.

	Relevance norm.	TREC 2009				TREC 2010			
		λ	ERR-IA	α-nDCG	P-IA	λ	ERR-IA	α-nDCG	P-IA
BM25	-	-	0.1878	0.2757	0.0760	-	0.1947	0.2788	0.1254
org_xQuAD	MinMax	1.00	0.2242	0.3240	0.0769	0.99	0.2372	0.3281	0.1256
	Sum	0.15	0.2181	0.3154	**0.0902**	0.77	0.2506	**0.3570**	**0.1589**
	Virtual	1.00	**0.2318**	**0.3263**	0.0802	0.95	**0.2634**	0.3509	0.1315
IA-Select	MinMax	-	0.2242	0.3240	0.0769	-	0.2445	0.3386	0.1252
	Sum	-	0.2141	0.3162	**0.0929**	-	0.2529	0.3568	**0.1592**
	Virtual	-	**0.2318**	**0.3263**	0.0802	-	**0.2681**	**0.3660**	0.1334
PM2	MinMax	0.40	0.2233	0.3271	0.0899	0.57	0.2477	0.3576	0.1515
	Sum	0.57	0.2233	0.3266	0.0898	0.62	0.2571	0.3651	0.1555
	Virtual	0.52	**0.2328**	**0.3330**	**0.0932**	0.46	**0.2675**	**0.3713**	**0.1601**

Our observations from Table 1 can be summarized as follows. First, as shown in the literature, all the baseline strategies outperform the non-diversified BM25 baseline. Secondly, we find that all the diversification strategies are sensitive to the score normalization component; a finding that justifies our interest in the normalization techniques in this paper. Third, for the org_xQuAD and IA-Select strategies, the normalization schemes Virtual and/or Sum yield a better performance than the MinMax (especially on TREC 2010), which demonstrates that they can help in remedying the aspect elimination problem for these two diversification strategies. In particular, the org_xQuAD strategy with Virtual yields the highest ERR-IA scores (on both topic sets) and α-nDCG score (on TREC 2010 set). Similarly, IA-Select achieves its best performance with the normalization techniques Sum (for the Precision-IA metric) and Virtual (for ERR-IA and α-nDCG metrics). Finally, PM2 also benefits from Virtual as for all the reported evaluation metrics, its best performance is observed with the latter normalization technique.

Performance of xQuAD variants

In Table 2, we compare the diversification performance of the original xQuAD to the variants that use arithmetic and geometric means for the novelty estimation components, namely, art_xQuAD and geo_xQuAD, respectively. For the ease of comparison, we repeat the results for org_xQuAD from Table 1. As before, each strategy is combined with three different normalization techniques.

Our findings in Table 2 reveal that the novelty estimation methods proposed in this paper considerably improve the org_xQuAD. The highest scores for all of the evaluation metrics (as shown in boldface in Table 2) are produced by the art_xQuAD and geo_XQuAD strategies that usually employ Virtual method for the relevance score normalization. For instance, using the TREC2010 topics, the art_xQuAD variant with Virtual normalization scheme provides a relative improvement of around 7% for both ERR-IA and α-nDCG metrics over the best-performing configuration of the original xQuAD strategy.

TABLE 2. Diversification performance of the xQuAD variants using the query aspects obtained from the official sub-topics. The highest scores across all methods are shown in boldface.

	Relevance norm.	TREC 2009				TREC 2010			
		λ	ERR-IA	α-nDCG	P-IA	λ	ERR-IA	α-nDCG	P-IA
BM25			0.1878	0.2757	0.0760	-	0.1947	0.2788	0.1254
org_xQuAD	MinMax	1	0.2242	0.3240	0.0769	0.99	0.2372	0.3281	0.1256
	Sum	0.15	0.2181	0.3154	0.0902	0.77	0.2506	0.3570	0.1589
	Virtual	1	0.2318	0.3263	0.0802	0.95	0.2634	0.3509	0.1315
geo_xQuAD	MinMax	0.92	0.2305	0.3301	0.0857	0.97	0.2494	0.3461	0.1418
	Sum	0.15	0.2174	0.3134	0.0892	0.75	0.2495	0.3515	0.1571
	Virtual	0.56	0.2333	0.3292	0.0905	0.86	**0.2842**	**0.3876**	0.1606
art_xQuAD	MinMax	0.91	0.2326	**0.3374**	0.0912	0.92	0.2629	0.3732	0.1578
	Sum	0.15	0.2174	0.3134	0.0892	0.75	0.2495	0.3515	0.1571
	Virtual	0.57	**0.2338**	0.3301	**0.0918**	0.86	0.2835	0.3868	**0.1609**

We further investigate the impact of the trade-off parameter λ on the performance of xQuAD using the union of topics from TREC 2009 and 2010. In Figure 2, we report the α-nDCG@20 scores for org_xQuAD using all three normalization methods, and for our geo_XQuAD and art_XQuAD only with the best-performing normalization, Virtual (to simplify the plot). The trade-off parameter λ is varied in the range [0, 1] with a step size of 0.01. Our findings reveal that both Sum and Virtual normalization techniques outperform MinMax for the entire range of values for the org_xQuAD strategy. Furthermore, while Sum reaches the peak effectiveness score when λ is around 0.15, the other two techniques perform better as we increase the λ; and the overall best performance for org_xQuAD is obtained with Virtual for $\lambda = 1$. Vargas et al. (2012) and Zheng et al. (Zheng & Fang, 2011) independently report a similar finding; i.e., the best λ value being 1 for xQuAD, and the latter work attributes this due to the use of real sub-topics from TREC as the query aspects. Nevertheless, our art_xQuAD and geo_xQuAD strategies with Virtual normalization yield the best effectiveness results and outperform org_xQuAD coupled with any of these normalization techniques.

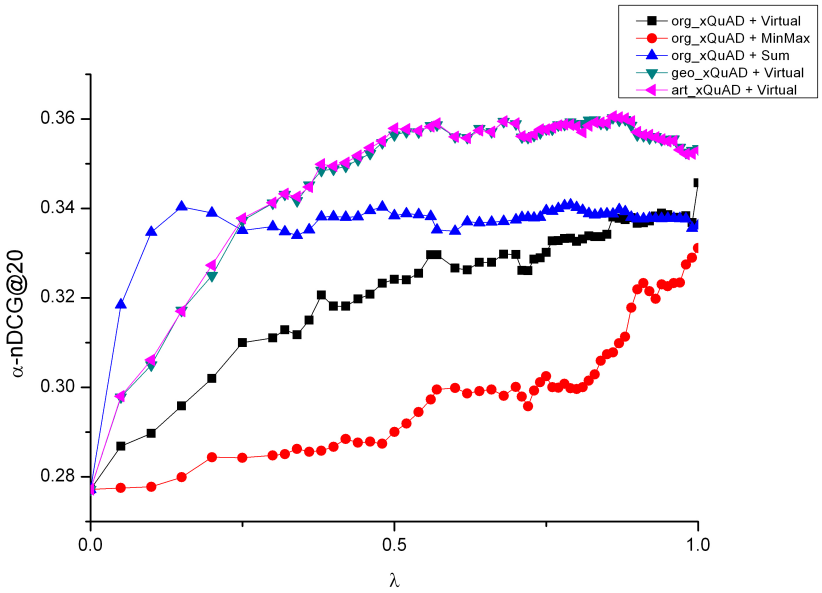

FIG 2. Diversification performance of the xQuAD variants vs. trade-off parameter λ.

Performance of the Score and Rank Aggregation Methods

In this section, we first evaluate the performance of the score aggregation methods mix_CombSUM and mix_CombMNZ. Table 3 shows their performance when coupled with each of the three relevance normalization schemes. Our findings reveal that both methods significantly outperform the non-diversified BM25 baseline. For both TREC 2009 and 2010 topic sets, mix_CombSUM coupled with the Virtual normalization technique outperforms all other configurations for the majority of the metrics (see the boldfaced cells in Table 3). This is a further evidence for the robustness and usability of the Virtual technique in the context of result diversification.

TABLE 3. Diversification performance of the score aggregation methods using the query aspects obtained from the official sub-topics. The highest scores across all methods are shown in boldface.

	Relevance norm.	TREC 2009				TREC 2010			
		λ	ERR-IA	α-nDCG	P-IA	λ	ERR-IA	α-nDCG	P-IA
BM25		-	0.1878	0.2757	0.0760	-	0.1947	0.2788	0.1254
mix_CombMNZ	MinMax	0.45	0.2188	0.3191	0.0915	0.85	0.2510	0.3536	0.1563
	Sum	0.3	0.2135	0.3142	0.0950	0.05	0.2448	0.3502	0.1541
	Virtual	0.75	0.2209	0.3216	0.0958	0.25	0.2450	0.3487	0.1557
mix_CombSUM	MinMax	0.7	0.2230	0.3255	0.0923	0.95	0.2599	0.3599	**0.1638**
	Sum	0.15	0.2174	0.3134	0.0892	0.75	0.2495	0.3511	0.1569
	Virtual	0.55	**0.2370**	**0.3320**	**0.0975**	0.9	**0.2719**	**0.3712**	0.1609

Next, we report our results for the rank aggregation methods, namely, Simple Voting (mix_SV), Borda Voting (mix_BV) and Markov chain based models (mix_MC1, mix_MC2, mix_MC3, and mix_MC4). Table 4 reveals that, mix_MC2 outperforms both the other Markov chain based strategies and the relatively simplistic methods mix_SV and mix_BV (especially for the ERR-IA and α-nDCG metrics). In contrary, the latter methods perform well for the P-IA metric. A further comparison of Tables 3 and 4 shows that the score aggregation methods are usually superior to Simple Voting and Borda Voting. However, the rank aggregation

TABLE 4. Diversification performance of the rank aggregation methods using the query aspects obtained from the official sub-topics. The highest scores across all methods are shown in boldface.

	TREC 2009				TREC 2010			
	λ	ERR-IA	α-nDCG	P-IA	λ	ERR-IA	α-nDCG	P-IA
BM25		0.1878	0.2757	0.0760	-	0.1947	0.2788	0.1254
mix_SV	0.9	0.2077	0.3094	**0.0954**	0.9	0.2327	0.3381	0.1524
mix_BV	0.85	0.2140	0.3135	0.0910	1	0.2437	0.3475	**0.1743**
mix_MC1	-	0.2129	0.3182	0.0915	-	0.2234	0.3328	0.1460
mix_MC2	-	**0.2249**	**0.3307**	0.0888	-	**0.2559**	**0.3645**	0.1404
mix_MC3	-	0.2183	0.3204	0.0914	-	0.2275	0.3367	0.1462
mix_MC4	-	0.2177	0.3157	0.0878	-	0.2489	0.3505	0.1390

methods based on the Markov chains perform comparable to the score based methods. These findings confirm the previous results reported in the context of meta-search (Renda & Straccia, 2003).

Finally, in Table 5 we make an overall comparison of the best-performing configurations (determined based on the α-nDCG@20 scores) of each of the three diversification baselines to those representing each class of the strategies proposed in this paper, namely, xQuAD variants, and the score and rank aggregation methods. From Table 5, we first observe that Virtual turns out to be the most effective normalization technique for the majority of the diversification strategies. More crucially, the score aggregation method mix_CombSUM and xQuAD variants are always the best performers for different evaluation metrics on both TREC 2009 and 2010 topic sets (see the boldfaced cells in Table 5). Given that we have three strong diversification strategies that are presented in their best configurations, our improvements are remarkable. For instance, on TREC 2010, our geo_XQuAD variant provides a relative improvement of 4% and 6% for the ERR-IA and α-nDCG metrics, respectively, over the best diversification baseline (PM2 with the Virtual normalization).

TABLE 5. Comparison of the best cases for the baseline and proposed methods using the query aspects obtained from the official sub-topics. The highest scores across all methods are in boldface.

		Rel. norm.	λ	ERR-IA	α-nDCG	P-IA
TREC 2009						
Baseline	BM25	-	-	0.1878	0.2757	0.0760
	IA-Select	Virtual	-	0.2318^{B}	0.3263^{B}	$0.0802^{P, C}$
	org_xQuAD	Virtual	1.00	0.2318^{B}	0.3263^{B}	$0.0802^{P, C}$
	PM2	Virtual	0.52	0.2328^{B}	0.3330^{B}	$0.0932^{X, I}$
Proposed	mix_CombSUM	Virtual	0.55	$\mathbf{0.2370}^{B*}$	0.3320^{B*}	$\mathbf{0.0975}^{B, X, I}$
	mix_MC2	-	-	0.2249^{B*}	0.3307^{B*}	0.0888^{B}
	art_xQuAD	MinMax	0.91	0.2326^{B*}	$\mathbf{0.3374}^{B*}$	0.0912^{B*}
TREC 2010						
Baseline	BM25	-	-	0.1947	0.2788	0.1254
	IA-Select	Virtual	-	$0.2681^{B*, Xg}$	$0.3660^{B*, Xg*}$	$0.1334^{X*,P*,C*,Xg*}$
	org_xQuAD	Sum	0.77	0.2506^{B}	$0.3570^{B*, Xg}$	$0.1589^{B*, I*}$
	PM2	Virtual	0.46	$0.2675^{B, Xg*}$	$0.3713^{B*, Xg*}$	$0.1601^{B*, I*, M}$
Proposed	mix_CombSUM	Virtual	0.9	$0.2719^{B*, Xg}$	$0.3712^{B*, Xg}$	$\mathbf{0.1609}^{B*, I*, M}$
	mix_MC2	-	-	0.2559^{B*}	0.3645^{B*}	$0.1404^{B, P, C, Xg}$
	geo_xQuAD	Virtual	0.86	$\mathbf{0.2842}^{B*, P*, I, C}$	$\mathbf{0.3876}^{B*,X,P*,I*,C}$	$0.1606^{B*, I*, M}$

Note. The superscripts of a result denote a statistically significant difference from the BM25 (B), IA-Select (I), org_xQuAD (X), PM2 (P), mix_MC2 (M), mix_CombSUM (C) or geo_xQuAD (X_g) at 0.05 level. The superscripts with (*) denote a statistically significant difference at 0.01 level.

We also conducted an analysis of the statistical significance of our findings using Wilcoxon signed-rank test at the 95% and 99% confidence levels. We found that while all the diversification strategies significantly outperform the non-diversified baseline for most of the cases, the results are mixed among the diversification strategies. However, recent works in the literature also present similar findings. For instance, Dang and Croft report that none of the improvements of PM2 over the original xQuAD strategy are indeed statistically significant on TREC 2009 topics; and their results are also mixed on TREC 2010 (see Table 2 in (Dang & Croft, 2012)). We also observed a larger number of statistically significant cases on TREC

2010 topic set, which is possibly due to the much larger differences among the actual effectiveness scores of the strategies (e.g., see Table 5).

Impact of the Components and Parameters

Impact of the aspect representation. In this experiment, for each query in our topic files, we obtain the top-10 query suggestions (auto-completions) from Google search engine to represent the aspects, as in (Santos et al., 2010). Some of these suggestions include terms that are not in the collection vocabulary, and after filtering the suggestions with such terms, we ended up with 9 aspects per query, on the average.

In Table 6, we present the best-performing configurations for the sake of brevity[10]. We first notice that the effectiveness scores are considerably lower than those presented in Table 5. This is expected and confirms the previous findings (e.g., see (Santos et al., 2010)), as the suggestions cannot perfectly represent the query aspects as the actual sub-topics from TREC. As a further difference, for the baseline strategies, there are cases where MinMax outperform the others. This is because in this setup, we have a far larger number of aspects per query as mentioned above, and this probably makes the aspect elimination problem less of a concern.

Nevertheless, the trends in Table 6 are still similar to our previous results, as the xQuAD variants and/or rank and score aggregation methods are superior to all the traditional baselines. In particular, geo_xQuAD (mix_CombSUM) achieves the highest P-IA and α-nDCG scores on TREC 2009 (2010) sets, respectively. Remarkably, mix_CombSUM provides a relative improvement of 15.3% over the best-performing baseline strategy, PM2 with the Sum normalization, in terms of the P-IA metric on TREC 2010 topics. In this latter case, the differences between the mix_CombSUM and all other strategies (except art_xQuAD) are found to be statistically significant at 95% confidence level.

[10] The detailed results are provided in the Appendix A1.

TABLE 6. Comparison of the best cases for the baseline and proposed methods using the query aspects obtained from the suggestions. The highest scores across all methods are shown in boldface.

		Rel. norm.	λ	ERR-IA	α-nDCG	P-IA
TREC 2009						
Baseline	BM25	-	-	0.1878	0.2757	0.0760
	IA-Select	MinMax	-	0.1778	0.2814	0.0783
	org_xQuAD	MinMax	0.83	0.1884^{C}	0.2801^{Xg}	0.0757^{Xg}
	PM2	MinMax	0.25	0.1937	0.2891	0.0840
Proposed	mix_CombSUM	Virtual	0.25	$0.2004^{B, X}$	0.2913	0.0847
	mix_MC4		-	**0.2014**	0.2937^{B}	0.0801
	geo_xQuAD	MinMax	0.86	0.1938	$\mathbf{0.2948}^{X}$	$\mathbf{0.0868}^{B, X}$
TREC 2010						
Baseline	BM25	-	-	0.1947	0.2788	0.1254
	IA-Select	Virtual	-	0.2028	0.2966	$0.1129^{X, C*}$
	org_xQuAD	Sum	0.1	0.2041^{C}	0.2963	$0.1369^{B, I, C}$
	PM2	Sum	0	0.2145	0.3028	0.1297^{C*}
Proposed	mix_CombSUM	Virtual	0.3	0.2161^{X}	**0.3123**	$\mathbf{0.1499}^{B*, X, P*, I, S*}$
	mix_SV	-	0.85	**0.2271**	0.3027	0.1277^{C*}
	art_xQuAD	Virtual	0.38	0.2070	0.3008	0.1360^{B}

Note. The superscripts of a result denote a statistically significant difference from the BM25 (B), IA-Select (I), org_xQuAD (X), PM2 (P), mix_SV (S), mix_CombSUM (C) or geo_xQuAD (X_g) at 0.05 level. The superscripts with (*) denote a statistically significant difference at 0.01 level.

Impact of the initial retrieval model. In order to investigate the impact of the initial retrieval model, we repeated all the experiments using the query-likelihood language model (LM) with Dirichlet smoothing (Zhai & Lafferty, 2004). Table 7 shows the best-performing configurations when the query aspects are based on the TREC sub-topics[11]. As before, the proposed methods perform quite well and for the majority of the evaluation metrics, the score aggregation methods mix_CombSUM and mix_CombMNZ outperform their competitors. Note that, the second best-performer is again an xQuAD variant, either art_xQuAD or geo_xQuAD.

[11] The detailed results are provided in the Appendix A2.

TABLE 7. Comparison of the best cases for the baseline and proposed methods using the LM for the initial retrieval and query aspects obtained from the official sub-topics. The highest scores across all methods are shown in boldface.

		Rel. norm.	λ	ERR-IA	α-nDCG	P-IA
TREC 2009						
Baseline	LM			0.1738	0.2645	0.0930
	IA-Select	MinMax	-	0.2240^{B}	0.3311^{B*}	0.0920
	org_xQuAD	MinMax	1.00	0.2240^{B}	0.3311^{B*}	0.0920
	PM2	MinMax	0.66	0.2160	0.3259^{B}	$0.0923^{C, Xa}$
Proposed	mix_CombMNZ	MinMax	0.45	$\mathbf{0.2343}^{B*}$	$\mathbf{0.3334}^{B*}$	$\mathbf{0.1022}^{P}$
	mix_MC2		-	0.2222^{B*}	0.3282^{B*}	0.0944
	art_xQuAD	Virtual	0.95	0.2240	0.3284^{B}	0.1006^{P}
TREC 2010						
Baseline	LM	-	-	0.1959	0.2842	0.1406
	IA-Select	Virtual	-	0.2631^{B*}	0.3624^{B*}	$0.1291^{X*, C*, Xg*}$
	org_xQuAD	Sum	0.56	0.2634^{B*}	0.3689^{B*}	$0.1562^{P*, I*, M}$
	PM2	MinMax	0.76	0.2679^{B*}	0.3751^{B*}	$0.1314^{X*, C, Xg*}$
Proposed	mix_CombSUM	Virtual	-	$\mathbf{0.2740}^{B*}$	$\mathbf{0.3805}^{B*}$	$0.1519^{P, I*}$
	mix_MC2	-	-	0.2645^{B*}	0.3714^{B*}	0.1394^{X}
	geo_xQuAD	Virtual	0.78	0.2721^{B*}	0.3792^{B*}	$\mathbf{0.1614}^{P*, I*}$

Note. The superscripts of a result denote a statistically significant difference from LM (B), IA-Select (I), org_xQuAD (X), PM2 (P), mix_MC2 (M), mix_CombSUM (C), art_xQuAD (X_a) or geo_xQuAD (X_g) at 0.05 level. The superscripts with (*) denote a statistically significant difference at 0.01 level.

In Table 8, we continue with the best-performing configurations for the experiments that employ the search engine suggestions as the query aspects[12]. As before, the actual scores are lower for all metrics in comparison to Table 7, but the trends are similar in that the score aggregation method mix_CombSUM yields the best diversification performance for the majority of the cases.

[12] The detailed results are provided in the Appendix A3.

TABLE 8. Comparison of the best cases for the baseline and proposed methods using the LM for the initial retrieval and query aspects obtained from the suggestions. The highest scores across all methods are shown in boldface.

		Rel. norm.	λ	ERR-IA	α-nDCG	P-IA
TREC 2009						
Baseline	LM	-		0.1738	0.2645	0.0930
	IA-Select	MinMax	-	0.1913	0.2916	$0.0908^{X, Xg}$
	org_xQuAD	Sum	0.57	0.1928^{B}	0.2929^{B}	$0.1014^{I, M}$
	PM2	MinMax	0.74	0.1891	0.2870	0.0921^{Xg}
Proposed	mix_CombSUM	MinMax	0.85	**0.1992**	**0.2967**	0.0965
	mix_MC2	-	-	0.1955	0.2917	$0.0907^{X, Xg}$
	geo_xQuAD	Virtual	0.76	0.1938	0.2941	$\mathbf{0.1001}^{I, M}$
TREC 2010						
Baseline	LM	-	-	0.1959	0.2842	0.1406
	IA-Select	Virtual	-	0.2106	0.3078^{B}	$0.1195^{X*, C, Xg}$
	org_xQuAD	Sum	0.46	0.2164^{B}	0.3147^{B*}	$0.1486^{I*, M}$
	PM2	MinMax	0.46	0.2039	0.3033	$0.1344^{X*, C, Xg}$
Proposed	mix_CombSUM	Sum	0.50	$\mathbf{0.2149}^{B}$	0.3124^{B}	$\mathbf{0.1480}^{I, M}$
	mix_MC1	-	-	0.2124	$\mathbf{0.3165}^{B}$	$0.1329^{X, C, Xg}$
	geo_xQuAD	Virtual	0.62	0.2146	0.3122^{B}	$0.1464^{I, M}$

Note. The superscripts of a result denote a statistically significant difference from the LM (B), IA-Select (I), org_xQuAD (X), PM2 (P), mix_CombSUM (C), mix_MC2 (M), or geo_xQuAD (X_g) at 0.05 level. The superscripts with (*) denote a statistically significant difference at 0.01 level.

Other score normalization techniques. In addition to those discussed in the previous sections, we also repeat our experiments using another normalization technique, namely, z-score normalization (Renda & Straccia, 2003). This technique subtracts the mean score of τ from each score, and then divides them by the standard deviation of the ranking. Since the resulting score values do not fall into [0, 1] range, they are further normalized using the MinMax method. In our experiments, we find that the z-score normalization does not yield

better results than MinMax when coupled with our diversification strategies, and thus the results are not reported here.

Impact of the probability mixture model in ranking aggregation. For all the score and rank aggregation methods considered in this study, we also experimented with the versions that do not take the initial ranked list τ_q into account during the diversification process. Our results reveal that, for almost all cases and evaluation metrics, the versions with the probability mixture model are superior to their counterparts without the model.

Summary of the Main Findings

Our experimental evaluations reveal that the new xQuAD variants art_xQuAD and geo_xQuAD (coupled with the Virtual normalization technique) considerably improve the performance of the original strategy. We further show that the score and rank aggregation methods adapted for the result diversification problem are quite effective. In particular, we find that mix_CombSUM and mix_MC2 are the best-performing representatives of the score and rank aggregation methods, respectively.

Overall, the proposed xQuAD variants and certain ranking aggregation methods (especially mix_CombSUM) consistently outperform all three diversification baselines for most of the cases and evaluation metrics (as shown in Tables 5, 6, 7, and 8). The success of mix_CombSUM is remarkable as its computational complexity is less than the baseline diversification strategies and xQuAD variants, as we discuss in the section Score-based Aggregation Methods. This finding further justifies the use of the ranking aggregation methods in the context of search result diversification, as we propose in this paper.

Conclusion

In this paper, we improved the state-of-the-art in explicit search result diversification from two major directions. First, we proposed optimizations for the relevance score normalization and novelty estimation components of xQuAD, a top-performing approach for the explicit result diversification. We showed that the new xQuAD variants outperform both the original strategy and other diversification baselines employed in our paper. Second, we adapted various score and rank aggregation strategies that are used in meta-search scenarios in the literature to the diversification problem. Our experiments revealed that some of these strategies, despite their simplicity, also serve well for the diversification purposes and outperform three state-of-the-art baselines from the literature. This is an especially important finding given that these ranking aggregation methods can be computed more efficiently than the baseline diversification strategies and our xQuAD variants.

As a future work, we plan to investigate the diversification performance of our methods in other domains (e.g., for the recommendation systems).

References

Agrawal, R., Gollapudi, S., Halverson, A., & Ieong, S. (2009). Diversifying search results. In *Proceedings of the 2nd Int'l Conf. on Web Search and Web Data Mining (WSDM'09)* (p. 5-14).

Aksoy, C., Can, F., & Kocberber, S. (2012). Novelty detection for topic tracking. *Journal of the American Society for Information Science and Technology (JASIST)*, *63*(4), 777-795.

Arampatzis, A., & Kamps, J. (2009). A signal-to-noise approach to score normalization. In *Proceedings of the 18th ACM Conf. on Information and Knowledge Management (CIKM'09)* (p. 797-806).

Aslam, J. A., & Montague, M. (2001). Models for metasearch. In *Proceedings of the 24th Annual Int'l ACM SIGIR Conf. on Research and Development in IR* (p. 276-284).

Borda, J. C. (1781). Memorie sur les elections au scrutin. In *Histoire de l'academic royale des sciences.*

Cambazoglu, B. B., Altingovde, I. S., Ozcan, R., & Ulusoy, O. (2012). Cache-based query processing for search engines. *ACM Transactions on the Web, 6*(4), 14.

Capannini, G., Nardini, F. M., Perego, R., & Silvestri, F. (2011). Efficient diversification of web search results. *PVLDB, 4*(7), 451-459.

Carbonell, J. G., & Goldstein, J. (1998). The use of MMR, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of the 21st Annual Int'l ACM SIGIR Conf. on Research and Development in IR* (p. 335-336).

Carterette, B., & Chandar, P. (2009). Probabilistic models of ranking novel documents for faceted topic retrieval. In *Proceedings of the 18th ACM Conf. on Information and Knowledge Management (CIKM'09)* (p. 1287-1296).

Clarke, C. L. A., Craswell, N., & Soboroff, I. (2009). Overview of the trec 2009 web track. In *Proceedings of the 18th text retrieval conference (TREC'09).*

Clarke, C. L. A., Craswell, N., Soboroff, I., & Cormack, G. V. (2010). Overview of the trec 2010 web track. In *Proceedings of the 19th text retrieval conference (TREC'10).*

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*(5), 441-465.

Dang, V., & Croft, W. B. (2012). Diversity by proportionality: an election-based approach to search result diversification. In *Proceedings of the 35th Int'l ACM SIGIR Conf. on Research and Development in IR* (p. 65-74).

Dwork, C., Kumar, R., Naor, M., & Sivakumar, D. (2001). Rank aggregation methods for the Web. In *Proceedings of the 10th Int'l Conf. on World Wide Web (WWW'01)* (p. 613-622).

Fernandez, M., Vallet, D., & Castells, P. (2006). Probabilistic score normalization for rank aggregation. In *Proceedings of the 28th European Conf. on IR Research (ECIR'06)* (p. 553-556).

Fox, J. A., & Shaw, E. (1994). Combination of multiple sources: The trec-2 interactive track matrix experiment. In *Proceedings of the 17th Annual Int'l ACM SIGIR Conf. on Research and Development in IR.*

Gollapudi, S., & Sharma, A. (2009). An axiomatic approach for result diversification. In *Proceedings of the 18th Int'l Conf. on World Wide Web (WWW'09)* (p. 381-390).

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 (JASIST)*, *62*(3), 550-571.

Lee, J. H. (1997). Analyses of multiple evidence combination. In *Proceedings of the 20th Annual Int'l ACM SIGIR Conf. on Research and Development in IR* (pp. 267–276).

Macdonald, C., Ounis, I., & Tonellotto, N. (2011). Upper-bound approximations for dynamic pruning. *ACM Transactions on Information Systems*, *29*(4), 17.

Minack, E., Siberski, W., & Nejdl, W. (2011). Incremental diversification for very large sets: a streaming-based approach. In *Proceedings of the 34th Int'l ACM SIGIR Conf. on Research and Development in IR* (p. 585-594).

Radlinski, F., & Dumais, S. T. (2006). Improving personalized web search using result diversification. In *Proceedings of the 29th Annual Int'l ACM SIGIR Conf. on Research and Development in IR* (p. 691-692).

Rafiei, D., Bharat, K., & Shukla, A. (2010). Diversifying web search results. In *Proceedings of the 19th Int'l Conf. on World Wide Web (WWW'10)* (pp. 781–790).

Ravana, S. D., & Moffat, A. (2009). Score aggregation techniques in retrieval experimentation. In *Proceedings of the 20th Australasian Database Conference (ADC'09)* (p. 59-67).

Renda, M. E., & Straccia, U. (2003). Web metasearch: Rank vs. score based rank aggregation methods. In *Proceedings of the 2003 ACM Symposium on Applied Computing (SAC'03)* (p. 841-846).

Robertson, S., Walker, S., Jones, S., Hancock-Beaulieu, M., & Gatford, M. (1994). Okapi at trec-3. In *TREC* (pp. 109–126).

Santos, R. L. T. (2013). *Explicit web search result diversification* (Unpublished doctoral dissertation). University of Glasgow.

Santos, R. L. T., Castells, P., Altingovde, I. S., & Can, F. (2013). Diversity and novelty in information retrieval. In *Proceedings of the 36th Int'l ACM SIGIR Conf. on Research and Development in IR* (p. 1130).

Santos, R. L. T., Macdonald, C., & Ounis, I. (2010). Exploiting query reformulations for web search result diversification. In *Proceedings of the 19th Int'l Conf. on World Wide Web (WWW'10)* (p. 881-890).

Santos, R. L. T., Macdonald, C., & Ounis, I. (2011). Intent-aware search result diversification. In *Proceedings of the 34th Int'l ACM SIGIR Conf. on Research and Development in IR* (p. 595-604).

Vallet, D., & Castells, P. (2012). Personalized diversification of search results. In *Proceedings of the 35th Int'l ACM SIGIR Conf. on Research and Development in IR* (p. 841-850).

Vargas, S., & Castells, P. (2011). Rank and relevance in novelty and diversity metrics for recommender systems. In *Proceedings of the 2011 ACM Conference on Recommender Systems (RecSys'11)* (p. 109-116).

Vargas, S., Castells, P., & Vallet, D. (2012). Explicit relevance models in intent-oriented information retrieval diversification. In *Proceedings of the 35th Int'l ACM SIGIR Conf. on Research and Development in IR* (p. 75-84).

Vieira, M. R., Razente, H. L., Barioni, M. C. N., Hadjieleftheriou, M., Srivastava, D., Jr., C. T., & Tsotras, V. J. (2011). On query result diversification. In *Proceedings of the 27th Int'l Conf. on Data Engineering (ICDE'11)* (p. 1163-1174).

Wang, J., & Zhu, J. (2009). Portfolio theory of information retrieval. In *Proceedings of the 32nd Annual Int'l ACM SIGIR Conf. on Research and Development in IR* (pp. 115–122).

Welch, M. J., Cho, J., & Olston, C. (2011). Search result diversity for informational queries. In *Proceedings of the 20th Int'l Conf. on World Wide Web (WWW'11)* (p. 237-246).

Witten, I. H., Moffat, A., & Bell, T. C. (1999). Managing Gigabytes (2nd ed.): Compressing and Indexing Documents and Images. San Francisco, CA: Morgan Kaufmann Publishers.

Zettair. (2006). *Zettair open-source search engine*. http://www.seg.rmit.edu.au/zettair/.

Zhai, C., Cohen, W. W., & Lafferty, J. (2003). Beyond independent relevance: Methods and evaluation metrics for subtopic retrieval. In *Proceedings of the 26th Annual Int'l ACM SIGIR Conf. on Research and Development in IR* (pp. 10–17).

Zhai, C., & Lafferty, J. D. (2004). A study of smoothing methods for language models applied to information retrieval. *ACM Transactions on Information Systems, 22*(2), 179-214.

Zhai, C., & Lafferty, J. D. (2006). A risk minimization framework for information retrieval. *Inf. Process. Manage.*, *42*(1), 31-55.

Zheng, W., & Fang, H. (2011). A comparative study of search result diversification methods. In *Proceedings of the ECIR 2011 Workshop on Diversity in Document Retrieval.*

Zheng, W., Wang, X., Fang, H., & Cheng, H. (2012). Coverage-based search result diversification. *Information Retrieval* , *15*(5), 433-457.

Zuccon, G., & Azzopardi, L. (2010). Using the quantum probability ranking principle to rank interdependent documents. In *Proceedings of the 32nd European Conf. on IR Research (ECIR'10)* (pp. 357–369).

Zuccon, G., Azzopardi, L., Zhang, D., & Wang, J. (2012). Top-k retrieval using facility location analysis. In *Proceedings of the 34th European Conf. on IR Research (ECIR'12)* (p. 305-316).

APPENDIX A1

TABLE A1-1. Diversification performance w.r.t. the relevance normalization techniques using the query aspects obtained from the suggestions and BM25 as the retrieval model. The highest scores are shown in boldface.

	Relevance norm.	TREC 2009				TREC 2010			
		λ	ERR-IA	α-nDCG	P-IA	λ	ERR-IA	α-nDCG	P-IA
BM25	-	-	0.1878	0.2757	0.0760	-	0.1947	0.2788	0.1254
org_xQuAD	MinMax	0.83	0.1884	**0.2801**	0.0757	0.60	0.1921	0.2779	0.1235
	Sum	0.20	**0.1902**	0.2792	**0.0822**	0.10	**0.2041**	**0.2963**	**0.1369**
	Virtual	0.97	0.1797	0.2737	0.0763	0.38	0.2012	0.2883	0.1241
IA-Select	MinMax	-	0.1778	**0.2814**	0.0783	-	0.1863	0.2815	0.1103
	Sum	-	**0.1806**	0.2688	**0.0796**	-	**0.2035**	**0.2962**	**0.1300**
	Virtual	-	0.1744	0.2675	0.0779	-	0.2028	0.2966	0.1129
PM2	MinMax	0.25	**0.1937**	**0.2891**	**0.0840**	0.34	0.2021	0.3014	**0.1318**
	Sum	0.25	0.1809	0.2710	0.0798	0	**0.2145**	**0.3028**	0.1297
	Virtual	0.64	0.1692	0.2636	0.0791	0.05	0.2118	0.3006	0.1302

TABLE A1-2. Diversification performance of the xQuAD variants using the query aspects obtained from the suggestions and BM25 as the retrieval model. The highest scores across all methods are shown in boldface.

	Relevance norm.	TREC 2009				TREC 2010			
		λ	ERR-IA	α-nDCG	P-IA	λ	ERR-IA	α-nDCG	P-IA
BM25	-		0.1878	0.2757	0.0760	-	0.1947	0.2788	0.1254
org_xQuAD	MinMax	0.83	0.1884	0.2801	0.0757	0.60	0.1921	0.2779	0.1235
	Sum	0.2	0.1902	0.2792	0.0822	0.10	0.2041	0.2963	0.1369
	Virtual	0.97	0.1797	0.2737	0.0763	0.38	0.2012	0.2883	0.1241
geo_xQuAD	MinMax	0.86	0.1938	**0.2948**	0.0868	0.95	0.2025	0.2971	0.1234
	Sum	0.2	0.1913	0.2828	0.0829	0.1	0.2022	0.2921	0.1396
	Virtual	0.5	0.1938	0.2904	0.0860	0.38	0.2068	0.3005	0.1359
art_xQuAD	MinMax	0.82	**0.1954**	0.2936	**0.0887**	0.66	**0.2070**	0.2968	0.1328
	Sum	0.2	0.1913	0.2828	0.0829	0.1	0.2022	0.2921	**0.1396**
	Virtual	0.5	0.1924	0.2854	0.0836	0.38	**0.2070**	**0.3008**	0.1360

TABLE A1-3. Diversification performance of the score aggregation methods using the query aspects obtained from the suggestions and BM25 as the retrieval model. The highest scores across all methods are shown in boldface.

	Relevance norm.	TREC 2009				TREC 2010			
		λ	ERR-IA	α-nDCG	P-IA	λ	ERR-IA	α-nDCG	P-IA
BM25	-	-	0.1878	0.2757	0.0760	-	0.1947	0.2788	0.1254
Mix_CombMNZ	MinMax	1	0.1906	0.2811	0.0800	0.1	0.2012	0.2838	0.1257
	Sum	1	0.1817	0.2735	0.0819	0	0.1947	0.2788	0.1254
	Virtual	0.6	0.1879	0.2795	0.0818	0.1	0.2224	0.3073	0.1303
Mix_CombSUM	MinMax	0.9	0.1953	0.2871	**0.0881**	0.6	0.1919	0.2840	0.1300
	Sum	0.2	0.1914	0.2829	0.0829	0.1	0.2033	0.2942	0.1400
	Virtual	0.25	**0.2004**	**0.2913**	0.0847	0.3	**0.2161**	**0.3123**	**0.1499**

TABLE A1-4. Diversification performance of the rank aggregation methods using the query aspects obtained from the suggestions and BM25 as the retrieval model. The highest scores across all methods are shown in boldface.

	TREC 2009				TREC 2010			
	λ	ERR-IA	α-nDCG	P-IA	λ	ERR-IA	α-nDCG	P-IA
BM25		0.1878	0.2757	0.0760	-	0.1947	0.2788	0.1254
mix_SV	0.95	0.1738	0.2671	0.0808	0.85	**0.2271**	**0.3027**	0.1277
mix_BV	1	0.1932	0.2851	**0.0844**	0.9	0.2127	0.2991	**0.1374**
mix_MC1	-	0.1873	0.2815	0.0795	-	0.2059	0.2902	0.1243
mix_MC2	-	0.1914	0.2858	0.0799	-	0.2060	0.2976	0.1214
mix_MC3	-	0.1911	0.2854	0.0798	-	0.2016	0.2885	0.1252
mix_MC4	-	**0.2014**	**0.2937**	0.0801	-	0.2068	0.2904	0.1286

APPENDIX A2

TABLE A2-1. Diversification performance w.r.t. the relevance normalization techniques using the query aspects obtained from the official sub-topics and LM as the retrieval model. The highest scores are shown in boldface.

	Relevance norm.	TREC 2009				TREC 2010			
		λ	ERR-IA	α-nDCG	P-IA	λ	ERR-IA	α-nDCG	P-IA
LM	-	-	0.1738	0.2645	0.0930	-	0.1959	0.2842	0.1406
org_xQuAD	MinMax	1	0.2240	**0.3311**	0.0920	1.00	0.2517	0.3499	0.1496
	Sum	0.44	0.2078	0.3065	0.0939	0.56	**0.2634**	**0.3689**	**0.1562**
	Virtual	0.92	**0.2242**	0.3283	**0.0940**	0.79	0.2584	0.3514	0.1409
IA-Select	MinMax	-	**0.2240**	**0.3311**	**0.0920**	-	0.2517	0.3499	**0.1496**
	Sum	-	0.2113	0.3096	0.0874	-	0.2547	0.3618	0.1451
	Virtual	-	0.2143	0.3148	0.0774	-	**0.2631**	**0.3624**	0.1291
PM2	MinMax	0.66	**0.2160**	**0.3259**	**0.0923**	0.76	**0.2679**	**0.3751**	0.1314
	Sum	0.44	0.2110	0.3111	0.0896	0.71	0.2469	0.3547	**0.1326**
	Virtual	0.1	0.2094	0.3076	0.0888	0.8	0.2662	0.3674	0.1331

TABLE A2-2. Diversification performance of the xQuAD variants using the query aspects obtained from the official sub-topics and LM as the retrieval model. The highest scores across all methods are shown in boldface.

	Relevance norm.	TREC 2009				TREC 2010			
		λ	ERR-IA	α-nDCG	P-IA	λ	ERR-IA	α-nDCG	P-IA
LM	-		0.1738	0.2645	0.0930	-	0.1959	0.2842	0.1406
Org_xQuAD	MinMax	1	0.2240	**0.3311**	0.0920	1.00	0.2517	0.3499	0.1496
	Sum	0.44	0.2078	0.3065	0.0939	0.56	0.2634	0.3689	0.1562
	Virtual	0.92	**0.2242**	0.3283	0.0940	0.79	0.2584	0.3514	0.1409
Geo_xQuAD	MinMax	0.96	0.2163	0.3238	0.0918	1	0.2521	0.3530	0.1478
	Sum	0.4	0.2048	0.3038	0.0940	0.79	0.2568	0.3527	0.1453
	Virtual	0.95	0.2238	0.3281	**0.1006**	0.78	**0.2721**	**0.3792**	**0.1614**
art_xQuAD	MinMax	0.97	0.2166	0.3235	0.0978	0.91	0.2604	0.3687	0.1576
	Sum	0.4	0.2048	0.3038	0.0940	0.79	0.2568	0.3527	0.1453
	Virtual	0.95	0.2240	0.3284	**0.1006**	0.78	**0.2721**	**0.3792**	**0.1614**

TABLE A2-3. Diversification performance of the score aggregation methods using the query aspects obtained from the official sub-topics and LM as the retrieval model. The highest scores across all methods are shown in boldface.

	Relevance norm.	TREC 2009				TREC 2010			
		λ	ERR-IA	α-nDCG	P-IA	λ	ERR-IA	α-nDCG	P-IA
LM	-	-	0.1738	0.2645	0.0930	-	0.1959	0.2842	0.1406
Mix_CombMNZ	MinMax	0.45	**0.2343**	**0.3334**	0.1022	0.45	0.2552	0.3604	0.1549
	Sum	0.55	0.2138	0.3077	0.0948	0.15	0.2597	0.3618	0.1560
	Virtual	0.8	0.2273	0.3242	0.1007	0.25	0.2506	0.3533	0.1564
mix_CombSUM	MinMax	0.85	0.2193	0.3207	**0.1036**	0.8	0.2639	0.3682	**0.1648**
	Sum	0.4	0.2047	0.3037	0.0943	0.8	0.2546	0.3508	0.1449
	Virtual	0.95	0.2232	0.3253	0.1033	0.75	**0.2712**	**0.3780**	0.1639

TABLE A2-4. Diversification performance of the rank aggregation methods using the query aspects obtained from the official sub-topics and LM as the retrieval model. The highest scores across all methods are shown in boldface.

	TREC 2009				TREC 2010			
	λ	ERR-IA	α-nDCG	P-IA	λ	ERR-IA	α-nDCG	P-IA
LM		0.1738	0.2645	0.0930	-	0.1959	0.2842	0.1406
mix_SV	0.8	0.2119	0.3137	**0.0983**	0.7	0.2421	0.3452	0.1535
mix_BV	0.85	0.2066	0.3071	0.0967	0.8	0.2362	0.3421	**0.1608**
mix_MC1	-	**0.2222**	0.3242	0.0950	-	0.2525	0.3606	0.1483
mix_MC2	-	**0.2222**	**0.3282**	0.0944	-	**0.2645**	**0.3714**	0.1394
mix_MC3	-	0.2185	0.3213	0.0952	-	0.2591	0.3677	0.1479
mix_MC4	-	0.2134	0.3117	0.0921	-	0.2484	0.3594	0.1497

APPENDIX A3

TABLE A3-1. Diversification performance w.r.t. the relevance normalization techniques using the query aspects obtained from the suggestions and LM as the retrieval model. The highest scores are shown in boldface.

	Relevance norm.	TREC 2009				TREC 2010			
		λ	ERR-IA	α-nDCG	P-IA	λ	ERR-IA	α-nDCG	P-IA
LM	-	-	0.1738	0.2645	0.0930	-	0.1959	0.2842	0.1406
org_xQuAD	MinMax	0.97	0.1923	**0.2929**	0.0941	0.99	0.2057	0.3020	0.1398
	Sum	0.57	**0.1928**	**0.2929**	**0.1014**	0.46	**0.2164**	**0.3147**	**0.1486**
	Virtual	0.97	0.1895	0.2905	0.0945	0.78	0.2127	0.2997	0.1393
IA-Select	MinMax	-	0.1913	**0.2916**	0.0908	-	0.1997	0.2956	0.1294
	Sum	-	**0.1985**	0.2881	**0.0953**	-	0.2021	0.3020	**0.1424**
	Virtual	-	0.1890	0.2847	0.0882	-	**0.2106**	**0.3078**	0.1195
PM2	MinMax	0.74	**0.1891**	**0.2870**	0.0921	0.46	**0.2039**	**0.3033**	0.1344
	Sum	0.9	0.1721	0.2702	0.0893	0	0.1956	0.2913	0.1353
	Virtual	0.87	0.1697	0.2670	**0.0950**	0.2	0.1954	0.2867	**0.1367**

TABLE A3-2. Diversification performance of the xQuAD variants using the query aspects obtained from the suggestions and LM as the retrieval model. The highest scores across all methods are shown in boldface.

	Relevance norm.	TREC 2009				TREC 2010			
		λ	ERR-IA	α-nDCG	P-IA	λ	ERR-IA	α-nDCG	P-IA
LM	-		0.1738	0.2645	0.0930	-	0.1959	0.2842	0.1406
org_xQuAD	MinMax	0.97	0.1923	0.2929	0.0941	0.99	0.2057	0.3020	0.1398
	Sum	0.57	0.1928	0.2929	**0.1014**	0.46	**0.2164**	**0.3147**	0.1486
	Virtual	0.97	0.1895	0.2905	0.0945	0.78	0.2127	0.2997	0.1393
geo_xQuAD	MinMax	0.85	0.1934	0.2939	0.0986	0.94	0.2064	0.3091	0.1422
	Sum	0.57	0.1913	0.2871	0.0992	0.46	0.2160	0.3130	**0.1488**
	Virtual	0.76	**0.1938**	**0.2941**	0.1001	0.62	0.2146	0.3122	0.1464
art_xQuAD	MinMax	0.97	0.1923	0.2927	0.0957	0.84	0.2123	0.3090	0.1441
	Sum	0.57	0.1913	0.2871	0.0992	0.46	0.2160	0.3130	**0.1488**
	Virtual	0.77	0.1929	0.2931	0.0992	0.62	0.2139	0.3117	0.1469

TABLE A3-3. Diversification performance of the score aggregation methods using the query aspects obtained from the suggestions and LM as the retrieval model. The highest scores across all methods are shown in boldface.

	Relevance norm.	TREC 2009				TREC 2010			
		λ	ERR-IA	α-nDCG	P-IA	λ	ERR-IA	α-nDCG	P-IA
LM	-	-	0.1738	0.2645	0.0930	-	0.1959	0.2842	0.1406
mix_CombMNZ	MinMax	0.65	0.1941	0.2860	0.0963	0.5	0.2033	0.2983	0.1343
	Sum	0.35	0.1838	0.2763	0.0987	0.1	0.2117	0.3062	0.1326
	Virtual	0.15	0.1929	0.2865	0.0973	0.1	0.2001	0.2967	0.1341
mix_CombSUM	MinMax	0.85	**0.1992**	**0.2967**	0.0965	0.85	0.2116	0.3075	0.1452
	Sum	0.7	0.1913	0.2821	0.0983	0.5	**0.2149**	**0.3124**	0.1480
	Virtual	0.75	0.1871	0.2857	**0.1001**	0.55	0.2127	0.3114	**0.1485**

TABLE A3-4. Diversification performance of the rank aggregation methods using the query aspects obtained from the suggestions and LM as the retrieval model. The highest scores across all methods are shown in boldface.

	TREC 2009				TREC 2010			
	λ	ERR-IA	α-nDCG	P-IA	λ	ERR-IA	α-nDCG	P-IA
LM		0.1738	0.2645	0.0930	-	0.1959	0.2842	0.1406
mix_SV	0.95	0.1790	0.2711	**0.0944**	0.65	**0.2098**	0.2963	0.1370
mix_BV	0.35	0.1773	0.2682	0.0923	0.7	0.2063	0.2981	**0.1382**
mix_MC1	-	0.1872	0.2808	0.0918	-	0.2039	0.2936	0.1286
mix_MC2	-	**0.1932**	**0.2887**	0.0913	-	0.2024	**0.2984**	0.1230
mix_MC3	-	0.1873	0.2816	0.0929	-	0.2023	0.2953	0.1286
mix_MC4	-	0.1887	0.2786	0.0900	-	0.1889	0.2828	0.1294