Learning to Diversify Web Search Results with a Document Repulsion Model

Journal Item

Learning to Diversify Web Search Results with a Document Repulsion Model

Jingfei Li[a], Yue Wu[a], Peng Zhang[a,*], Dawei Song[a,b,*], Benyou Wang[a]

[a] *Tianjin Key Laboratory of Cognitive Computing and Application, School of Computer Science and Technology, Tianjin University, 300350, No.135 Yaguan Road, Haihe Educational Park, Tianjin, P.R.China*
[b] *School of Computing and Communications, the Open University, Walton Hall, Milton Keynes, MK7 6AA. United Kingdom*

Abstract

Search diversification (also called diversity search), is an important approach to tackling the query ambiguity problem in information retrieval. It aims to diversify the search results that are originally ranked according to their probabilities of relevance to a given query, by re-ranking them to cover as many as possible different aspects (or subtopics) of the query. Most existing diversity search models heuristically balance the relevance ranking and the diversity ranking, yet lacking an efficient learning mechanism to reach an optimized parameter setting. To address this problem, we propose a learning-to-diversify approach which can directly optimize the search diversification performance (in term of any effectiveness metric). We first extend the ranking function of a widely used learning-to-rank framework, i.e., LambdaMART, so that the extended ranking function can correlate relevance and diversity indicators. Furthermore, we develop an effective learning algorithm, namely Document Repulsion Model (DRM), to train the ranking function based on a Document Repulsion Theory (DRT). DRT assumes that two result documents covering similar query aspects (i.e., subtopics) should be mutually repulsive, for the purpose of search diversification. Accordingly, the proposed DRM exerts a repulsion force between each pair of similar documents in the learning process, and includes the diver-

[*] Corresponding authors: Dawei Song (Email: dwsong@tju.edu.cn) and Peng Zhang (Email: pzhang@tju.edu.cn). The first two authors have equal contribution to this work.

sity effectiveness metric to be optimized as part of the loss function. Although there have been existing learning based diversity search methods, they often involve an iterative sequential selection process in the ranking process, which is computationally complex and time consuming for training, while our proposed learning strategy can largely reduce the time cost. Extensive experiments are conducted on the TREC diversity track data (2009, 2010 and 2011). The results demonstrate that our model significantly outperforms a number of baselines in terms of effectiveness and robustness. Further, an efficiency analysis shows that the proposed DRM has a lower computational complexity than the state of the art learning-to-diversify methods.

Keywords: Search Diversification, Learning-to-Rank, Document Repulsion Model, Diversity Features

1. Introduction

In recent decades, Information Retrieval (IR) techniques have underpinned a growing number of Web information processing systems (e.g., search engines, recommender systems) that have changed the way people access and interact with information. The core research problem of IR is to rank documents with respect to a given query. Most traditional ranking models follow the Probability Ranking Principle (PRP) [17], which assumes that documents are independently ranked according to their probabilities of relevance to the query.

Despite its great success, the traditional PRP is insufficient to deal with the challenging issue of query ambiguity. Specifically, in Web search, there often exist numerous ambiguous queries that may have more than one interpretations (e.g., a query “apple” can refer to the fruit apple or the Apple corporation) or multiple subtopics (e.g., “program language” contains many different aspects). A PRP-based ranking model tends to first estimate the most probable interpretation (or subtopic) of a query, and then compute the relevance scores of documents with respect to this interpretation, and sort them in a descending order. A consequence is that the retrieval model may return wrong search results (due to the mis-estimation of the query intent) or redundant results for only one subtopic while leaving out relevant information about other subtopics. Such query ambiguity and result redundancy problem can be addressed by diversifying the search results (i.e., the selected relevant document for a lower ranking position should be as dissimilar as possible to the documents that have already ranked at the higher positions), so that the search results can cover multiple subtopics and satisfy the users' diverse information needs.

In the literature, a range of diversity search approaches have been proposed. Essentially, most of them [1, 4, 18, 22, 27, 28] use a greedy algorithm[1] to re-rank the original result list by balancing the query-document relevance score and document-document dissimilarity scores. These approaches usually do not adopt a learning mechanism and are difficult to reach an optimized parameter setting, thus limiting the effectiveness of search diversification.

In this paper, we aim to develop a learning-to-diversify approach by directly optimizing an effectiveness metric, such as α-$nDCG$ [7], within the popular Learning to Rank (LTR) framework. LTR involves learning to optimize a ranking function based on a set of features. In line with the state of the art diversity search models [1, 18, 28], we first define a ranking function for search diversification, which can integrate both relevance features (including query-dependent features, document-dependent features and query-document dependent features [13]) and diversity features (including document-document features that capture the interrelationships between documents) into our proposed learning-to-diversify approach. To do this, we formalize a series of typical diversity features derived from selected diversity models [28, 30]. Note that, not all diversity models can be used, which will be discussed in Section 3.2.5.

Then, the key challenge is how to consider document diversity in the learning process for the defined ranking function. In order to address this problem, we propose a novel Document Repulsion Theory (DRT). Essentially, DRT assumes

[1] For each ranking position, a greedy algorithm computes the diversity scores for all unselected documents and select the one with the highest diversity score.

that (1) two documents in a relevant-irrelevant document pair (i.e., two documents that have different relevance scores, one is more relevant than another.) should be mutually repulsive for the purpose of relevance ranking (i.e., the relevant document should be pushed upwards and the irrelevant document gets pushed downwards in the ranking list); (2) for the purpose of diversity ranking, two documents covering similar query aspects (i.e., subtopics) should also be mutually repulsive. Intuitively, if a pair of topically similar documents can be automatically separated from each other, the final ranking of results will become diversified naturally. Based on the above assumptions, we develop a document repulsion model (DRM) to simulate DRT in the learning process, which not only maximizes the diversity metric but also maintains the quality of relevance.

In order to implement the DRM, we borrow the idea of relevance-irrelevance document repulsion as used in a popular learning-to-rank algorithm, namely LambdaMART. Specifically, for a pair of documents (d_1, d_2), if d_1 is more relevant than d_2, then LambdaMART will exert a repulsion force with size $|\lambda_{12}|$, to push up d_1 and push down d_2. Similarly, we can incorporate an additional repulsion force between two documents sharing similar query subtopics, so that the similar documents can be naturally separated. The direction of movement of separated documents will be determined by the original relevance scores, in order to guarantee that the repulsion operation will not hurt the quality of the relevance ranking.

We have carried out extensive experiments on the TREC diversity track data and Clueweb09B document collection. The experimental results show the effectiveness and robustness of our proposed DRM model. We also theoretically show the efficiency of the proposed model in comparison with various state of the art learning-to-diversify methods through a complexity analysis.

In a nutshell, the major contributions of this paper are summarized as follows:

- First, we propose a novel Document Repulsion Model (DRM) which leads to an improved Learning-to-Rank algorithm for search diversification.

- Second, we prove what diversity features are suitable for DRM, based on which we further formalize a series of novel diversity features that take into account the interrelationship between documents.
- Third, we conduct extensive comparative experiments and gain insightful findings about the proposed model from a range of different perspectives.

The rest of the paper is organized as follows. Section 2 presents a review of the related work, which motivates the proposed document repulsion model as detailed in Section 3. Section 4 reports our experimental setup. The experimental results are reported and discussed in Section 5. In Section 6, we conclude the paper and point out future research directions.

2. Related Work

Search diversification can be used to solve the problems of query ambiguity and result redundancy. Algorithmatically, it can be seen as an instance of the maximum coverage problem [10] which is NP-hard. Most existing search diversification approaches apply an iterative sequential selection process for each ranking position to re-rank the original search results. They can be organized into a two-dimensional taxonomy [20], i.e., diversification strategies and query aspect (subtopic) representation methods.

Two main diversity strategies [16] include extrinsic diversity (coverage-based) and intrinsic diversity (novelty-based). The former aims at retrieving search results by considering all possible interpretations of a query, thus maximizing the coverage of query aspects. The latter aims at avoiding redundancy in the search results. The methods for aspect (subtopic) representation can be grouped into implicit representation and explicit representation. Specifically, implicit representation methods do not mine query aspects explicitly and assume that similar documents cover similar query aspects; while explicit representation methods usually use external information (e.g., query logs) to explicitly model query aspects. Jointly considering the diversity strategies and aspect representation methods, the existing diversity search approaches can be classified into the following categories: implicit coverage-based, explicit coverage-based, implicit novelty-based, explicit novelty-based, and hybrid approaches.

There exist a number of implicit novelty-based approaches, such as the widely applied Maximal Marginal Relevance (MMR) [4] and others [22, 27, 28, 30]. They are non-learning approaches, and usually use heuristic rules to sequentially select documents from a candidate document set by considering the already selected documents. The current selected document needs to be maximally dissimilar to the documents ranked at the higher positions. Additionally, the Affinity Ranking (AR) approach [28, 25] computes diversity scores of documents based on the "information richness" (derived from the Affinity Graph), using a greedy algorithm to penalize the unselected document with all documents in the selected document set. Similarly, the Quantum Probability Ranking Principle (QPRP) [30] takes a document's relevance score as the original information richness score, but uses a different penalty item. Except for QPRP, all of these approaches have various free parameters that define the trade-off between "query-document" similarity and "document-document" similarity. As non-learning methods, they often use some heuristic methods (e.g., grid search) to tune parameters. In this paper, we use machine learning methods to train the model parameters automatically. Moreover, we adopt a number of diversity features inspired by some aforementioned methods, e.g., AR and QPRP.

Moreover, there exist various learning-based approaches, such as [12, 15, 21, 29] (belonging to categories of implicit coverage-based or hybrid approaches). For example, Radlinski et al.[15] proposed an online learning approach that uses multi-armed bandit and click data to minimize the abandonment activity (i.e., users do not find any satisfied results). However, it requires external data and only solves the maximized coverage of query aspects. Another learning method, presented in [29], does not model the query aspects explicitly, but considers both coverage and novelty problems at the same time. In this hybrid learning method, the training and ranking processes are based on the MMR criterion [4]. It has led to an improvement over the original search results.

Xia et al. [12] proposed a learning approach, which is similar to the work in [29] in term of the ranking process (i.e., MMR based ranking) but differs in the learning process that trains the ranking model by directly optimizing the diversity evaluation metric. Note that, the diversity search approaches in both [12] and [29] apply an iterative sequential ranking function in both learning and ranking processes, which is computationally expensive. Additionally, Yue and Joachims [21] replaced the subtopic coverage with word coverage to solve both coverage and novelty problems within the framework of Support Vector Machines (SVM). However, they do not consider the document relevance.

For explicit approaches, some external information is often required. For example, query logs have been used [5, 14] to mine the query aspects. Santos et al. [18] used query reformulations to represent different query aspects. The Open Directory Project (ODP) has also been used [1]. Up to now, xQuAD[18] and IASelect[1] are considered as the most effective explicit diversification approaches. Similar to MMR, the ranking in these approaches is still a sequential selection process. Differently, xQuAD involves a probabilistic framework to measure the relationships between current document and the already selected documents. Our approach does not use any external information to represent query aspects. Nevertheless, formalizing query aspects explicitly as diversity features for our model is a research direction worth future investigation.

Our proposed model is an implicit approach and focuses on both novelty and coverage. Compared with the existing approaches, the main advantage of our model is that we develop a novel Document Repulsion Theory which then underpins a non-greedy learning process to achieve search diversification. In this way, we gain significant performance improvements with relatively lower computational cost.

3. Document Repulsion Model for Search Diversification

In this section, we propose a Document Repulsion Model (DRM) for learning to diversify, which can directly optimize an effectiveness metric. In the following, we first introduce the learning-to-rank framework. Then, we present a Document Repulsion Theory, based on which the Document Repulsion Model is developed within the learning-to-rank framework.

3.1. Learning to Rank Framework

The learning-to-rank framework is composed of a learning process and a ranking process. In the learning process, the training data is used to learn a ranking model by directly optimizing the evaluation metric. The learned ranking model is then used to re-rank the test data in the ranking process. Therefore, in the following, we first introduce the ranking function with a series of model parameters that need to be trained in the learning process. Then we describe the learning algorithms for training the model parameters. In the present paper, our learning algorithm is extended from a listwise learning-to-rank algorithm, namely LambdaMART [24]. The reason why we choose this approach is that LambdaMART [24] combines a tree-boosting optimization (called MART) [9] and a widely used learning-to-rank algorithm (called LambdaRank) [2]. Therefore, we present the LambdaRank [2] (LambdaMART is the boosted tree version of it) algorithm in this subsection.

3.1.1. Ranking Function

Traditional learning-to-rank models compute documents' ranking scores independently and sort them in a decreasing order. Formally , let $X^i = \{x^i_1, ..., x^i_n\}$, where x^i_j denotes the feature vector of a document j given a query i. The ranking score for each document can be computed as follows:

$$f(x^i_j) = w^T x^i_j \tag{1}$$

where w^T encodes the model parameters which need to be trained. The query-dependent features, document-dependent features and query-document features are used in the ranking function. In the next subsection, we introduce a well known Learning to Rank algorithm, i.e., LambdaRank [3], for training this ranking function.

3.1.2. Introduction to LambdaRank

The LambdaRank algorithm is derived from RankNet [3]. The cost function of RankNet involves penalizing the document pairs that are incorrectly ranked, while rewarding the pairs that are correctly ranked. Specifically, the cost function C is formalized as follows:

$$C = \frac{1}{2}(1 - S_{ij})\sigma(s_i - s_j) + \log(1 + e^{-\sigma(s_i - s_j)}) \tag{2}$$

where s_i and s_j are the model scores of documents i and j respectively; σ is the shape parameter of the sigmoid function; $S_{ij} = 1$ if the relevance label of document i is larger than that of document j, and $S_{ij} = -1$ when the relevance label of document i is smaller than that of document j. The gradient of the cost function with respect to the model score s_i is:

$$\lambda_{ij} \equiv \sigma(\frac{1}{2}(1 - S_{ij}) - \frac{1}{1 + e^{\sigma(s_i - s_j)}}) \tag{3}$$

This gradient can be interpreted as a force. For document i and document j, if i is more relevant than j, this force will push i up with size λ_{ij} and push j down with size λ_{ij}. For each pair of documents which belongs to the set I (I is the set of document pairs $< i, j >$, in which document i is more relevant than document j), the λ_{ij} is computed. Then, for every document i, we can obtain:

$$\lambda_i = \sum_{j:\{i,j\} \in I} \lambda_{ij} - \sum_{j:\{j,i\} \in I} \lambda_{ij} = \sum_{\{i,j\} \rightleftharpoons I} \lambda_{ij} \tag{4}$$

where λ_i is computed from all pairs that contain document i. λ for each document can be regarded as an arrow, the direction of which represents the direction the document will move towards in the next iteration, and the length of which indicates the size of movement.

In order to optimize the evaluation metric directly, some rules are first made [2], and then the gradient is defined to meet the rules through modifying Eq. (3) by simply multiplying the change value of the evaluation metric $|\Delta Z|$ when swapping the rank positions of document i and document j:

$$\lambda_{ij} = \frac{-\sigma}{1 + e^{\sigma(s_i - s_j)}} |\Delta Z| \tag{5}$$

where the document i is more relevant than document j.

The intuitions can be described as follows [2]: it is much easier to make rules to guide the rank order of documents than to directly construct the cost function which desires certain rank order properties. Furthermore, the specified rules can be achieved through defining the gradient of the cost function. Note that, the cost function can be derived by computing the integral of the gradient.

A limitation of the lambdaRank algorithm is that it only considers the document pairs $< i, j >$ in which the relevance of document i is more or less than the relevance of document j. The gradient λ_{ij} can be seen as a repulsion force which can push the more relevant document up and the less relevant document down in the ranking. However, all document pairs whose component documents have the same relevance degrees are likely untouched in the learning process. As a result, documents covering similar query aspects are ranked closely.

In the following, we propose rules (as the Document Repulsion Theory) to guide the rank order of document in the learning process. We also define the gradient of cost function to capture these rules.

3.2. A Document Repulsion Model for Learning to Diversity

In this subsection, we describe how to extend the listwise learning-to-rank approach (LambdaMART [24]) to obtain the Document Repulsion Model. At first, we present the ranking function for diversification which considers both relevance ranking and diversity ranking (Section 3.2.1). Then, we propose a Document Repulsion Theory to resolve the problem described in Section 3.1.2. Correspondingly, we define the gradients of cost function for our learning-to-diversity approach and prove the validity of the cost function. Finally, the diversity features used in Document Repulsion Model are also formalized.

3.2.1. Ranking Function for Diversification

Santos et al. [19] used traditional ranking function to rank documents and train the model by directly optimizing the diversity evaluation metrics for search diversification. However, they did not gain good results. A likely reason is that

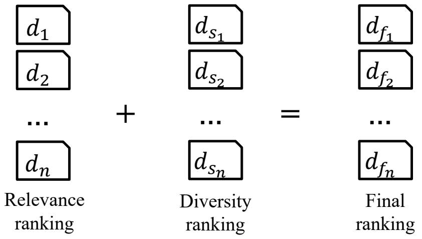


Figure 1: To combine the relevance ranking and diversity ranking as the final ranking.

they do not consider the diversity features in the final ranking function, although they have considered diversity evaluation metrics in the learning process.

In fact, most of the existing diversification models [11, 28] consider both relevance ranking and diversity ranking. The search diversification problem is regarded as a bi-criterion optimisation problem which needs to balance relevance and diversity in the final ranking function. As shown in Figure 1, the relevance ranking maximizes the relevance of top ranked results, while the diversity ranking maximizes the novelty of top ranked results, and the final ranking will consider both relevance and diversity.

Motivated by above discussion, we propose a balanced ranking function by directly adding a diversity part to the original ranking function, as formalized in Eq.6:

$$f(x_j^i, v_j^i) = w_r^T x_j^i + w_s^T v_j^i \tag{6}$$

where x_j^i denotes the relevance feature vector of the document j for query i, v_j^i represents the diversity feature vector of the document j for query i. Then, document scores are computed by this balanced ranking function. Finally, we sort documents in the decreasing order according to the final ranking scores.

After defining this balanced ranking function, our main problems become (i) how to consider diversity when training the model parameters and (ii) how to extract the diversity features, which will be addressed next.

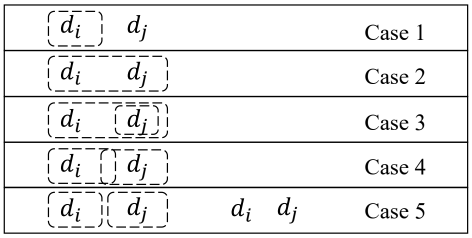


Figure 2: Different cases for document pairs $< i, j >$.

3.2.2. Document Repulsion Theory

The Document Repulsion Theory assumes that two documents in a document pair (i.e., one document is more relevant than the other) should be mutually repulsive for the purpose of relevance ranking (i.e., the relevant document gets pushed upwards and the irrelevant document gets pushed downwards), and two documents covering with similar query aspects (i.e., subtopics) also should be mutually repulsive for the purpose of diversity ranking. Intuitively, if the similar documents can be separated, the final ranking results will be diversified. According to the document repulsion theory, we make a series of repulsion rules for the learning process.

To this end, we define 5 cases, to which a document pair (e.g., $< i, j >$) may belong (as shown in Figure 2). (i) One document in the pair (e.g., document i) covers at least one subtopic, while the other document does not cover any. The original LambdaRank model can handle this case in the learning process. (ii) Document i and document j have the same subtopic coverage. In this case, the two documents in this pair should be separated according to the document repulsion theory. Specifically, the document ranked higher in original result list should be pushed up, while the other one should be pushed down. (iii) If the subtopics covered by document i contain all subtopics covered by document j, the intuition is to push document i up and document j down, since document i contains more subtopics and contribute more diversification to the final ranking. (iv) The document i has some overlap of subtopics with the document j. However, each document has some novelty information which could contribute to the final diversification ranking. Thus, they should not be mutually repulsive. (v) If two documents' subtopics do not have any overlap or they do not cover any query subtopics, there will be no any repulsion force.

Those repulsion forces existing in the first three cases are based on the following intuitive hypotheses: (1) The relevant documents (which cover at least one query subtopic) should rank higher than irrelevant documents (which do not cover any query subtopics). (2) If two documents have the same query subtopics coverage, the first document (which rank higher in the original result list) should be seen as more relevant compared with the other one, so the first document should rank higher than the other and thus needs to be pushed down to make a separation. (3) The documents covering more query subtopics are also regarded as more "relevant" than documents covering less query aspects. Therefore, they should be ranked higher and repulse the less relevant documents to make them apart from each other.

Intrinsically, more cases of document pairs are considered in our model than LambdaRank, which only considers the relevance-irrelevance document pairs. Similar to LambdaMART, our final model combines with MART to produce the boosted tree version. In the next subsection, we will introduce the document repulsion model (DRM), which is an operationalization of the document repulsion theory for diversity search.

3.2.3. Gradients of Cost Function for DRM

In the LambdaRank method, the gradient of weights is computed according to Eq.(4), where the set I only contains the first case illustrated in Figure 2. In our Document Repulsion Model, more cases are considered. As there exist more than one relevance labels for each document in the diversity search task, we use $T^{(i)}$ to replace the $Y^{(i)}$. $Y^{(i)}$ is a one-dimensional vector, where $Y_j^{(i)}$ represents the relevance label of document j for query i. In contrast, $T^{(i)}$ is matrix, in which the j^{th} row vector ($T_j^{(i)}$) represents the relevance label list of document j for query i (each element in the row vector corresponds to a query subtopic).

Then the computation of the gradient for every document can be illustrated by Algorithm 1 (detailed in Appendix), where (1) $T_k^{(i)} > T_l^{(i)}$ represents that document k covers one query subtopic, while document l does not cover any; (2) $T_k^{(i)} \supset T_l^{(i)}$ denotes that document k covers more query subtopics than document l; (3) $T_k^{(i)} \equiv T_l^{(i)}$ represents that two documents have the same query subtopics coverage. Additionally, we replace the Z in the computation function of λ_{kl} with the diversity evaluation metric (e.g., α-$NDCG$[7]) that is to be optimized directly.

3.2.4. Cost Function

We have already specified a set of rules based on the Document Repulsion Theory to determine how to change the rank order of documents, and defined the gradient of cost function to meet the rules. However, we still do not know whether the gradient can be successfully used to the learning process. To guarantee the effectiveness of the gradient, we need to prove the feasibility of the definition. The proof in [2] determines that the condition $\sum_{d_i \in D_Q} \lambda_i = 0$ (d_i is a document in the document set D_Q given a query Q) should be satisfied. If the gradient of cost function is defined, we should guarantee that the modified cost function exists and is convex, so that the proposed learning algorithm can be used.

According to [2], Eq.(7) can be used to determine if there exists a cost function. Furthermore, the cost function is convex if the Jacobian (that is, the matrix $J_{jk} \equiv \partial\lambda_j/\partial s_k$) is positive semidefinite for each j, k.

$$\frac{\partial \lambda_j}{\partial s_k} = \frac{\partial \lambda_k}{\partial s_j} \quad \forall j, k \in \{1, ..., n\} \tag{7}$$

Since the computation process of the λ_i (Eq.(4)) and the computation function of λ_{ij} are similar to those of LambdaRank (the consideration of more kinds of document pairs does not influence the computation of λ_i and λ_{ij}), there exists a convex cost function in our learning algorithm, showing the feasibility of the proposed learning algorithm.

Correspondingly, the cost function derived from the defined gradient can be

Algorithm 1 : Computation of gradients.

```
Input: $(q_i, X^{(i)}, T^{(i)}), x_j^{(i)} \in X^{(i)}$
    // $X^{(i)}$: feature vector set, $T^{(i)}$: relevance label matrix
Output: $\lambda^{(i)}, \lambda_j^{(i)} \in \lambda^{(i)}$
for $k = 1, ..., n$ do
    for $l = k+1, ..., n$ do
        if $T_k^{(i)} > T_l^{(i)}$ then
            $\lambda_{kl}^{(i)} = \frac{-\sigma}{1+e^{\sigma(s_k - s_l)}}|\triangle Z|$
            $\lambda_k^{(i)} + = \lambda_{kl}^{(i)}$
            $\lambda_l^{(i)} - = \lambda_{kl}^{(i)}$
        else if $T_k^{(i)} < T_l^{(i)}$ then
            $\lambda_{kl}^{(i)} = \frac{-\sigma}{1+e^{\sigma(s_l - s_k)}}|\triangle Z|$
            $\lambda_l^{(i)} + = \lambda_{kl}^{(i)}$
            $\lambda_k^{(i)} - = \lambda_{kl}^{(i)}$
        else if $T_k^{(i)} \supset T_l^{(i)}$ then
            $\lambda_{kl}^{(i)} = \frac{-\sigma}{1+e^{\sigma(s_k - s_l)}}|\triangle Z|$
            $\lambda_k^{(i)} + = \lambda_{kl}^{(i)}$
            $\lambda_l^{(i)} - = \lambda_{kl}^{(i)}$
        else if $T_k^{(i)} \subset T_l^{(i)}$ then
            $\lambda_{kl}^{(i)} = \frac{-\sigma}{1+e^{\sigma(s_l - s_k)}}|\triangle Z|$
            $\lambda_l^{(i)} + = \lambda_{kl}^{(i)}$
            $\lambda_k^{(i)} - = \lambda_{kl}^{(i)}$
        else if $T_k^{(i)} \equiv T_l^{(i)}$ then
            $\lambda_{kl}^{(i)} = \frac{-\sigma}{1+e^{\sigma(s_k - s_l)}}|\triangle Z|$
            $\lambda_k^{(i)} + = \lambda_{kl}^{(i)}$
            $\lambda_l^{(i)} - = \lambda_{kl}^{(i)}$
        end if
    end for
end for
return $\lambda^{(i)}$
```

formalized as follows:

$$C = \sum_{\{d_i,d_j\} \rightleftharpoons G} |\triangle Z| \log(1 + e^{-\sigma(s_i - s_j)}) \tag{8}$$

where G contains all document pairs with respect to the first three cases mentioned above. The meaning of the notation $\rightleftharpoons$ is similar to that in Eq.(4).

3.2.5. Formalization of Diversity Features

To conduct search diversification, we need to formalize a set of diversity features which are combined with the relevance features to determine the final ranking. To this end, we come up with an intuitive idea of using selected existing diversity models to form the diversity features directly. However, not all the existing diversity models are applicable in the learning-to-diversify framework, because the required model score of a document for diversity feature should reflect the diversity ranking order. Specifically, if document d_i ranks before the document d_j in diversity ranking list, the model score of the document d_i should be larger than that of the document d_j. Only this kind of model score can be used as diversity feature. In this paper, we select the diversity model score as a diversity feature according to the above rule (we call it "score-rank consistency" rule).

However, most diversity models apply the iterative sequential selection process to re-rank the initial ranking list, and the model score of a document may not necessarily meet the above rule (since the diversity score of a unselected document is updated for each iteration, we regard the diversity score in final selection iteration as the model score of the document).

Let us look at the MMR approach [4] as an example: for each document d_i in the unselected document set D_q, the diversity scores $f(q, d_i, D_q)$ can be computed by the Eq.(9):

$$f_{MMR}(q, d_i, D_q) = \lambda f_1(q, d_i) - (1 - \lambda) max_{d_j \in D_q} f_2(d_i, d_j) \tag{9}$$

where $f_1(q, d_i)$ denotes the relevance score of document d_i, and $f_2(d_i, d_j)$ represents the similarity score of documents d_i and d_j. However, this model does not meet the above score-rank consistency rule, as illustrated below.

Let document d_i be the i^{th} document in the final diversity ranking list. In the re-ranking process, the unselected document set D_q contains documents d_i and d_{i+1} when the selected document set already has $i-1$ documents. The reason why the model selects d_i for the position i is that the diversity score s_{d_i} of document d_i is the largest in the unselected document set D_q. Therefore, we can obtain the model score of document d_i in this iteration ($m_{d_i}{=}s_{d_i}$) according to the definition of model score. Similarly, we can obtain the model score $m_{d_{i+1}}$ of d_{i+1} in the next iteration which is the biggest diversity score in the unselected document set $D_q \setminus d_i$. However, the model score m_{d_i} (the biggest diversity score within the unselected document set D_q in the previous iteration) is not necessarily bigger than the model score $m_{d_{i+1}}$, because of the absence of the clear relation between the score m_{d_i} and the score $m_{d_{i+1}}$. Thus the "score-rank consistency" rule may be violated by the MMR model.

Among the existing diversity ranking approaches, we find that AR [28] and QPRP [30] satisfy this rule, as detailed below.

For the AR approach [28], a directed link graph, namely Affinity Graph, is used to produce the information coverage score ($InfoRich(d_i)$) for each document i. Then an iterative sequential selection algorithm is used to re-rank the result list by the novelty information coverage score ($AR(d_i)$). Specifically, the document with the highest novelty information score is selected at each rank position. The novelty information score for the document i in the unselected document set is computed by the following equation:

$$f_{AR}(q, d_i, D_q) = InfoRich(d_i) - \sum_{d_j \in D_q} \widehat{M}_{ji} InfoRich(d_j) \tag{10}$$

where $InfoRich(d_i)$ and $\widehat{M}_{ji}$ are produced by the Affinity Graph, and D_q is the set of already selected documents. Additionally, $\sum_{d_j \in D_q} \widehat{M}_{ji} InfoRich(d_j)$ is the penalty term, which exerts a penalty score for the candidate document by all the selected documents in D_q.

Zuccon and Azzopardi [30] proposed the quantum probability ranking principle (QPRP) which extends the probability ranking principle (PRP) by considering the influence of other documents when scoring a candidate document. They used Eq.(11) to compute the document ranking score, which can be seen as the novelty information score for a candidate document d_i compared with the selected documents D_q for the query q.

$$\begin{aligned} & f_{QPRP}(q, d_i, D_q) \\ & = p(d_i) - 2 \sum_{d_j \in D_q} \sqrt{p(d_i)}\sqrt{p(d_j)} f(d_i, d_j) \end{aligned} \tag{11}$$

where $p(d_i)$ denotes the relevance score of document d_i being relevant to the query, and $f(d_i, d_j)$ denotes the similarity score between document d_i and document d_j. The interference term $2\sum_{d_j \in D_q} \sqrt{p(d_i)}\sqrt{p(d_j)} f(d_i, d_j)$ is used to penalize the information redundancy of a candidate document compared with the selected documents.

The above two methods also use the iterative sequential selection process to rank documents. However, for each iteration, they impose a penalty to the diversity score of the previous iteration to update the current diversity score, rather than combine the original score and dissimilarity score. Here, we give a brief proof to show that the selected diversity models (i.e., AR and QPRP) satisfy the “score-rank consistency” rule.

For this purpose, we need to prove that the model score for the document at position k is greater than that at position $k+1$, which is formalized as $m_{d_k} \geq m_{d_{k+1}}$. When selecting a document for the rank position k, the unselected document set D_q contains the documents d_k and d_{k+1}. The reason why d_k is selected for the position k is that d_k has the maximum diversity score in D_q, so the diversity score of d_k is larger than the diversity score of d_{k+1}, formalized as $s_{d_k} \geq s_{d_{k+1}}$. In addition, we can obtain the model score m_{d_k} of document d_k in this iteration ($m_{d_k} = s_{d_k}$). This proof process is the same as that for the MMR model. However, the next step is different, which determines the suitability of the AR and QPRP models for diversity features.

For the next rank position $k+1$, the current document is selected according to equation $d_{k+1} = arg\max_{d_i \in \{D_q \setminus d_k\}} \{s_{d_i} - penalty(d_k, d_i)\}$, where s_{d_i} is the diversity score in the last iteration, $penalty(d_k, d_i) \geq 0$ is a penalty score of the current document considering the previously selected document k. Specifically, $penalty(d_k, d_i) = \hat{M}_{ki} InfoRich(d_k)$ in AR and $penalty(d_k, d_i) = \sqrt{p(d_i)}\sqrt{p(d_k)} f(d_i, d_k)$ in QPRP. Therefore, the model score $m_{d_{k+1}}$ of document d_{k+1} equals to $s_{d_k+1} - penalty(d_k, d_k + 1)$. Then, we have $m_{d_k} = s_{d_k}$, $s_{d_k} \geq s_{d_{k+1}}$, $m_{d_{k+1}} = s_{d_k+1} - penalty(d_k, d_k + 1)$ and $penalty(d_k, d_k + 1) \geq 0$, so $m_{d_k} \geq m_{d_{k+1}}$. Therefore, both the AR and QPRP models satisfy the "score-rank consistency" rule and the model scores can be used as the diversity features. Note that, the computation of aforementioned diversity features do not involve free parameters.

4. Experimental Setup

In this section, we describe the experimental setup, including data sets, diversification approaches for comparison, feature extraction, and the details of model testing.

4.1. Data Sets

Our experiments are conducted on TREC (Text REtrieval Conference[2]) diversity tasks, including TREC 2009 Web Track (50 topics), TREC 2010 Web Track (48 topics), and TREC 2011 Web Track (50 topics). For each topic (query), TREC assessors identify 2 to 8 subtopics (or aspects). In Figure 3, we report the distribution of queries over different numbers of subtopics. The relevance judgments for documents are conducted at the subtopic level. Specifically, TREC assessors label a relevance degree for a document with respect to each identified subtopic. We use the ClueWeb09 category-B as the document collection[3] which comprises 50 million English documents. The collection

[2] http://trec.nist.gov/tracks.html

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

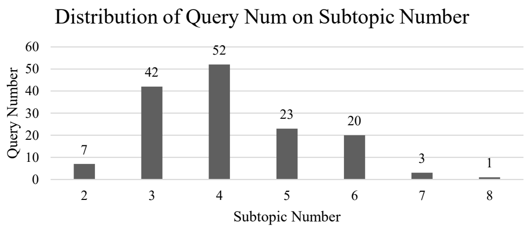


Figure 3: Distribution of query number on the number of subtopic number queries contain.

is indexed by the Indri toolkit (version 5.6)[4]. The indexing process involves basic pre-processing, including word stemming (with the Porter stemmer) and stopword removal (with standard English stopwords).

4.2. Diversification Approaches for Comparison

We evaluate our our proposed model (denoted as DRM), in comparison with a baseline language model (LM) used for initial relevance-based ranking and a number of state of the art diversity search models, including MMR [4], RankScoreDiff [11], QPRP [30], AR [28] and LambdaMART (with diversity optimization target) [3]. They are described as follows:

- LM is the initial ranking model which is implemented by the Indri search engine. All the following diversity models are achieved by re-ranking the initial results returned by LM.

- DRM is the proposed Document Repulsion Model which is extended from the LambdaMART approach.

- MMR is an implicit novelty-based approach, which considers both relevance and similarity factors of documents for ranking. The ranking process of MMR is implemented with the greedy algorithm, i.e., an iterative sequential selection of documents for each ranking position from a candidate document set considering the influence of previously selected documents. We choose it as a baseline for comparison because it is the first

[4] http://lemurproject.org/indri

implicit novelty-based diversification approach in the literature and is a representative diversity search approach.

- RankScoreDiff is an implicit coverage-based approach, which combines the initial relevance ranking list and the diversity ranking list. The diversity ranking is based on the difference between the initial rank scores (e.g., the query likelihood score) of adjacent documents. Note that, this is an non-greedy approach. The combination of the relevance features and diversity features in our approach is inspired by this method. Moreover, our final ranking function (DRM) is also non-greedy. Therefore, we select it as another baseline.

- QPRP is an implicit novelty-based approach which considers the inter-relationships between documents for the re-ranking purpose. It is also a greedy method which computes the ranking probability for each document by considering the penalty given by all documents in the selected document set.

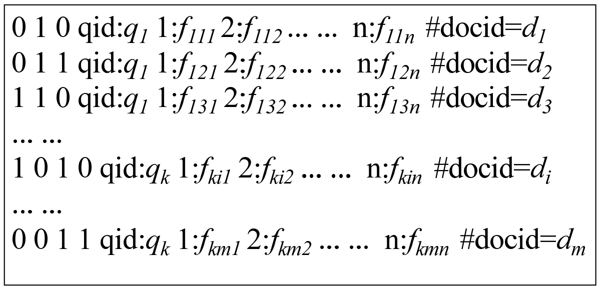
- AR is an implicit hybrid-based approach which utilizes a document-document relationship graph to compute the information coverage score for each document. The diversity score of a document is obtained by combining the information coverage score and the penalty scores exerted by all documents in the set of already selected document. For the purpose of ranking, AR combines the initial relevance score and diversity score together as the final rank score. Some diversity features used in our approach are extracted based on AR, therefore it is also used as a baseline.

- LambdaMART is a successful listwise learning-to-rank algorithm to deal with the ranking problem, which can directly optimize any IR evaluation metric. Here, we use the $\alpha\text{-}NDCG$ as the optimization target for the diversity retrieval task in this paper. Our model is extended from this approach, so we select it as a baseline.

Note that we choose a number of representative implicit diversity ranking approaches, which are closely related to ours, as baselines in the evaluation, since our method is also an implicit model. The explicit approaches are not empirically compared in this study. Indeed, there exist some recent learning methods (e.g., R-LTR [29] and PAMM [12]) that have achieved a good performance. However, the ranking function of them is still an iterative sequential selection process which is different from ours. Moreover, they exploit numerous external resources in features. Therefore, we do not conduct comparative experiments with them, but we analyze their difference from our approach based on the results reported in the corresponding papers.

4.3. Feature Extraction

In order to train and test our document repulsion model, we represent each query-document pair as a feature vector with n feature elements. All the features are pre-extracted offline and stored into a text file, and each row corresponds to a query-document pair. Figure 4 shows an example of extracted features stored in the feature file. The first K columns before "qid" correspond to the relevance judgments for each subtopic of the query (e.g., query q_1 contains $K = 3$ subtopics and q_k contains $K = 4$ subtopics). "qid:q_1" represents the query ID, "n:f_{kin}" is the n^{th} feature value for document d_i with respect to the query q_k, and "#docid=d_i" represents the document ID.



0 1 0 qid:q_1 1:f_{111} 2:f_{112} ... ... n:f_{11n} #docid=d_1
0 1 1 qid:q_1 1:f_{121} 2:f_{122} ... ... n:f_{12n} #docid=d_2
1 1 0 qid:q_1 1:f_{131} 2:f_{132} ... ... n:f_{13n} #docid=d_3
... ...
1 0 1 0 qid:q_k 1:f_{ki1} 2:f_{ki2} ... ... n:f_{kin} #docid=d_i
... ...
0 0 1 1 qid:q_k 1:f_{km1} 2:f_{km2} ... ... n:f_{kmn} #docid=d_m

Figure 4: An example of feature file which contains k queries.

The feature vector contains both relevance features and diversity features. We extract them as follows. The relevance features include various commonly

Table 1: Relevance features. For more details, please refer to [13].

Feature	Description
QueryTF	Sum of query term frequency in a document
DocLen	Number of words in a document
DocTF-Sum	Sum of document term frequency in the collection
DocTF-Min	Min of document term frequency in the collection
DocTF-Max	Max of document term frequency in the collection
DocTF-Mean	Mean of document term frequency in the collection
DocTF-Var	Variance of document term frequency in the collection
DocTFIDF-Sum	Sum of document tfidf in the collection
DocTFIDF-Min	Min of document tfidf in the collection
DocTFIDF-Max	Max of document tfidf in the collection
DocTFIDF-Mean	Mean of document tfidf in the collection
DocTFIDF-Var	Variance of document tfidf in the collection
TFIDF	TF×IDF score
BM25	BM25 score
LMIR-ABS	LMIR with ABS smoothing
LMIR-DIR	LMIR with DIR smoothing
LMIR-JM	LMIR with JM smoothing

used features in the literature [13], as summarized in Table 1. The diversity features shown in the Table 2 are extracted based on QPRP [30] and Affinity graph [28] (which are detailed in Section 3.2.5. The QPRP based features are computed using Eq.(11), where $f(d_i, d_j)$ is the Cosine similarity between documents d_i and d_j, represented as TF-IDF vectors. The other features are extracted based on Affinity graph [28].

4.4. Comparative Models

The official evaluation metrics for the diversity search task (α-$NDCG$ [7], ERR-IA [6] and $NRBP$ [8]) are adopted to evaluate the diversity models. The common idea of those metrics is to reward top ranked diversified and relevant results. Meanwhile they penalize the redundancy in search results by assigning

Table 2: Diversity features (QPRP based features are computed based on Eq. 11)

Feature	Description
QPRP-TF	QPRP value based on TF-IDF
QPRP-BM25	QPRP value based on BM25 score
QPRP-ABS	QPRP value based on LMIR with ABS smoothing
QPRP-DIR	QPRP value based on LMIR with DIR smoothing
QPRP-JM	QPRP value based on LMIR with JM smoothing
InfoRich	Information richness computed as in [28]
ARScore	AR score computed as in [28]

an increased probability of stopping browsing the results when users find the desired information. We set the related parameters α (for computing α-$nDCG$) and β (for computing $NRBP$) to 0.5, in order to guarantee the consistency with the official TREC evaluation methodology. Additionally, all the metrics are computed over the top-k ranked search results (k=20).

In order to augment the size of the training data for our learning model, we combine all the queries in the TREC Web Tracks from 2009 to 2011. The combined dataset contains 148 queries. All approaches are tested by re-ranking the original top 1000 documents retrieved by the Indri search engine (implemented with the query likelihood Language Model (LM)) for each query. For all approaches with free parameters, 5-fold cross-validation is conducted through optimizing the α-$NDCG$ ($k = 20$). The average performance over all test folds is reported. The significance test (t-test) has been performed for all the comparative diversity ranking approaches compared with the LM baseline.

5. Results and Discussions

In this section, we report and analyze the experiment results from different angles. We first report the overall average diversification performance on all queries in TREC Web Tracks 2009, 2010 and 2011, followed by average performance for different years separately to observe performances of different diversity models on different TREC data. The re-ranking performance of all models for different queries with different numbers of subtopics are also reported and analyzed, from which we can gain insights about the application scope of diversification models. Moreover, a robust analysis is conducted. We then carry out a component analysis to investigate how different components of our model contribute to the final diversification performance.

5.1. Overall Diversification Performance

The overall diversification performance of different models with respect to three official evaluation metrics are reported in Table 3. The relative improvements of all diversification models over the LM model are shown in the parentheses. As shown in the table, our diversification model (DRM) significantly outperforms the baseline LM model with respect to all evaluation metrics, by 33.58% for the α-$NDCG$, 52.31% for the ERR-IA and 72.94% for the $NRBP$ respectively. This result shows that our model is effective for the diversity search task.

The proposed model also outperforms the other diversification baselines significantly. For example, the widely used MMR model in general does not improve the original ranked results and even brings some harm to the diversification results. Similarly, the re-ranking performance of QPRP performs worse than LM. MMR and QPRP are rule-based greedy ranking algorithms. The poor diversification performance shows that simple rule-based methods have a limitation for web search diversification. RankScoreDiff and AR are diversification models that combine the relevance ranking and diversity ranking into the final ranking. From the experiment results, we find that they gain some improvements over the originally ranked results returned by LM. However, the improvements are not significant for all evaluation metrics. Overall, the diversification performances for all the above models (non-learning models) are rather poor, showing the limitation of non-learning algorithms. Compared with the non-learning approaches, LambdaMART gains a much larger improvement over the LM baseline by 15.1% for the α-$NDCG$, 27.41% for the ERR-IA and

Table 3: Overall performance on all queries in TREC 2009-2011. Significance Test has been conducted for all of the diversity models compared with the baseline LM model with t-test, where † means $p < 0.05$ and ‡ means $p < 0.01$.

Runs	α-$NDCG$	ERR-IA	$NRBP$
LM	0.2695	0.1751	0.1371
MMR	0.2681 (-0.52%)	0.1715 (-2.06%)	0.1313 (-4.23%)
QPRP	0.1663 (-38.31%)	0.1266 (-27.69%)	0.1109 (-19.11%)
RankScoreDiff	0.2705 (+0.37%)	0.1767 (+0.91%)	0.1392 (+1.53%)†
AR	0.2711 (+0.59%)	0.1765 (+0.79%)	0.1372 (0.07%)
LambdaMART	0.3102 (+15.10%)‡	0.2231 (+27.41%)‡	0.192 (+40.04%)‡
DRM	0.36 (+33.58%)‡	0.2667 (+52.31%)‡	0.2371 (+72.94%)‡

40.04% for the $NRBP$ respectively. Even through LambdaMART has achieved significant improvements, our DRM model still largely outperforms it. The result shows a superior performance of our proposed model.

5.2. Performance on Different TREC data

In the above subsection, we report the average evaluation results on all diversity tasks of TREC Web Track 2009-2011. Furthermore, we would like to find out the diversification performance on different subsets of the data, one for a specific year of TREC. The diversity tasks in different years have different properties. To this end, we report the re-ranking performance for TREC 2009 (Table 4), 2010 (Table 5) and 2011 (Table 6) respectively.

As shown in the three tables, we find that the relative trend of diversification performance for different models is consistent with the overall results reported in previous subsection. DRM is still the most effective one compared with other baseline diversification models. However, we can still observe some meaningful phenomenon by comparing the results for different years of TREC. The performances of all re-ranking models on TREC 2009 are largely better than those on TREC 2010 and 2011. Even the MMR, which did not perform well on the whole dataset, has got an improvement over the LM baseline on TREC 2009.

Table 4: Performance on TREC 2009 diversity tasks.

Runs	α-$NDCG$	ERR-IA	$NRBP$
LM	0.2028	0.1050	0.0743
MMR	0.2080 (+2.56%)†	0.1065 (+1.43%)†	0.0750 (+0.94%)
QPRP	0.1289 (-36.44%)	0.0809 (-22.95%)	0.0676 (-9.02%)
RankScoreDiff	0.2127 (+4.88%)†	0.1114 (+6.10%)†	0.0815 (+9.69%)†
AR	0.2123 (+4.68%)†	0.1168 (+11.23%)‡	0.0876 (+17.90%)‡
LambdaMART	0.2374 (17.06%)‡	0.1447 (37.81%)‡	0.1180 (58.81%)‡
DRM	0.2952 (+45.56%)‡	0.1958 (+86.48%)‡	0.1775 (+138.90%)‡

Table 5: Performance on TREC 2010 diversity tasks.

Runs	α-$NDCG$	ERR-IA	$NRBP$
LM	0.2078	0.1251	0.0896
MMR	0.2094 (+0.77%)	0.1219 (-2.56%)	0.0834 (-6.92%)
QPRP	0.1314 (-36.77%)	0.0918 (-26.62%)	0.0755 (-15.74%)
RankScoreDiff	0.2094 (+0.77%)	0.1288 (+2.96%)†	0.0944 (+5.36%)†
AR	0.2179 (+4.86%)†	0.1306 (+4.40%)†	0.0935 (+4.35%)†
LambdaMART	0.2356 (+13.37%)‡	0.1513 (+20.94%)‡	0.1185 (+32.25%)‡
DRM	0.3036 (+46.10%)‡	0.2247 (+79.62%)‡	0.1963 (+119.08%)‡

Table 6: Performance on TREC 2011 diversity tasks.

Runs	α-$NDCG$	ERR-IA	$NRBP$
LM	0.3954	0.2932	0.2457
MMR	0.3847 (-2.71%)	0.2842 (-3.07%)	0.2336 (-4.92%)
QPRP	0.2371 (-40.04%)	0.2055 (-29.91%)	0.1885 (-23.28%)
RankScoreDiff	0.3871 (-2.10%)	0.2881 (-1.74%)	0.2401 (-2.28%)
AR	0.3812 (-3.59%)	0.2804 (-4.37%)	0.2287 (-6.92%)
LambdaMART	0.4545 (+14.95%)‡	0.3703 (+26.29%)‡	0.3366 (+36.99%)‡
DRM	0.4782 (+20.94%)‡	0.3780 (+28.92%)‡	0.3360 (+36.75%)‡

The performance for TREC 2011 is not as good as for the other two years. All the non-learning models become worse after re-ranking the original results of LM, and the performance of LambdaMART and DRM also become less effective than that in TREC 2009 and 2010. The possible reason is that the diversity tasks in TREC 2011 are more difficult than that in TREC 2009 and 2010.

5.3. Performance on Different Queries

Table 7: Performance for different queries with different numbers of subtopics with respect to *α-NDCG*.

Subtopic Num	$num = 2$	$num = 3$	$num = 4$	$num = 5$	$num \geq 6$
LM	0.3816	0.3725	0.2404	0.2559	0.1325
MMR	0.3610	0.3672	0.2439	0.2473	0.1403
	(-5.40%)	(-1.42%)	(1.45%)†	(-3.36%)	(5.89%)†
QPRP	0.2270	0.2200	0.1611	0.1520	0.0794
	(-40.51%)	(-40.93%)	(-32.98%)	(-40.60%)	(-40.07%)
RankScoreDiff	0.3958	0.3694	0.2420	0.2426	0.1496
	(3.72%)†	(-0.83%)	(0.66%)	(-5.19%)	(12.90%)‡
AR	0.4076	0.3578	0.2469	0.2520	0.1507
	(6.81%)†	(-3.94%)	(2.70%)†	(-1.52%)	(13.73%)‡
LambdaMART	0.5129	0.3813	0.2826	0.2563	0.2382
	(34.41%)‡	(2.36%)†	(17.55%)‡	(0.16%)	(79.77%)‡
DRM	0.5407	0.4498	0.3172	0.2944	0.3043
	(41.69%)‡	(20.75%)‡	(31.95%)‡	(15.04%)‡	(129.99%)‡

In this subsection, we report and analyze the diversification performance on different queries with different numbers of subtopics. Intuitively, if a query has more subtopics, the query tends to be more ambiguous and would need more diversification. The results are reported in Table 7. DRM outperforms all other diversification models significantly on all queries. For each model, we find that the largest improvement over the baseline LM model is obtained when the number of subtopic is larger than 6 ($num \geq 6$), and the least improvement of performance is obtained when subtopic number is 5. This is an

Table 8: The robustness analysis of diversification performance for TREC 2009-2011.

Runs	α-$NDCG$	ERR-IA	$NRBP$
MMR	63/55	59/59	55/59
QPRP	32/96	20/108	43/85
RankScoreDiff	69/55	66/58	68/55
AR	75/50	70/55	74/50
LambdaMART	80/51	76/55	82/49
DRM	87/41	86/42	86/42

unexpected phenomenon. From Table 7, we observe that the performance of initial search results decreases with the increase of the number of subtopics except for $num = 5$. This shows that the more subtopics a query has, the more difficult to return diversified results based on the original LM. If the original performance is low (e.g., $num \geq 6$), there will be much room for improvement. If the original performance is already good (e.g., $num = 5$), there is little room for improvement that the diversification models can lead to.

5.4. Robustness Analysis

In addition to the effectiveness of diversification models that have been analyzed in the above subsections, we believe the robustness also needs to be analyzed. We use the Wins/Losses to measure the robustness of performance [23]. $Wins$ is the number of queries which gain improvements over LM, and $Losses$ is the number of queries whose performance are worse than LM. The queries with no difference in performance from LM were not considered. As shown in Table 8, our model is the most robust with respect to all evaluation metrics.

5.5. Components Analysis

Our diversity search model consists of three components, i.e., the optimization of diversity metric, the diversity features and the document repulsion algorithm. In this subsection, we analyze how different components contribute

Table 9: Component analysis for our proposed model on TREC 2009-2011.

Runs	α-$NDCG$	ERR-IA	$NRBP$
LM	0.2695	0.1751	0.1371
LambdaMART	0.3102 (+15.10%)‡	0.2231 (+27.41%)‡	0.192 (+40.04%)‡
LambdaMART$_{DR}$	0.3021 (+12.10%)‡	0.2066 (+17.98%)‡	0.1736 (+26.62%)‡
LambdaMART$_{DF}$	0.3386 (+25.64%)‡	0.2493 (+42.37%)‡	0.2186 (+59.44%)‡
DRM	0.36 (+33.58%)‡	0.2667 (+52.31%)‡	0.2371 (+72.94%)‡

to the final diversification performance. To this end, we test four sub-models composed of different components. They are "LambdaMART" (LambdaMART model with diversity evaluation metric α-$NDCG$ as the optimization target, without diversity features), "LambdaMART$_{DR}$" (LambdaMART with document repulsion learning algorithm), "LambdaMART$_{DF}$" (LambdaMART with diversity features) and "DRM" (the complete DRM diversification approach).

From Table 9, we can find that "LambdaMART$_{DF}$" significantly outperforms LambdaMART, which shows that adding extra diversity features to the basic learning-to-rank model is beneficial. In addition, we can further improve the diversification performance by implementing the document repulsion model (DRM). However, we find "LambdaMART$_{DR}$", which does not consider the diversity features, does not outperform the initial LambdaMART model. This is a meaningful phenomenon, which reveals that the proposed learning algorithm based on Document Repulsion Theory works only when diversity features are considered in the ranking function. To conclude, the combination of diversity features and document repulsion learning algorithm is the major contributor to improvement of the diversity search performance.

5.6. Discussion: Comparison with Recent Learning-to-Rank Approaches

We have shown the superiority of our approach compared with a number of implicit baseline approaches. These comparative approaches all belong to the same class of diversity search model with implicit aspect representation (See

Table 10: Comparison with some recent learning approaches with respect to α-$NDCG$@20.

Runs	TREC 2009	TREC 2010	TREC 2011
LM (Baseline)	0.2028	0.2078	0.3954
DRM	0.2954 (+45.56%)	0.3036 (+46.10%)	0.4782 (+20.94%)
QL (Baseline)	0.269	0.302	0.453
R-LTR	0.3964 (+47.21%)	0.4924 (+62.91%)	0.6297 (+39.07%)
R-LTR-NTN$_{doc2vec}$	0.4503 (+67.40%)	0.5376 (+78.01%)	0.6555 (+44.70%)
PAMM	0.4271 (+58.74%)	0.524 (+73.51%)	0.643 (+41.94%)
PAMM+NTN$_{doc2vec}$	0.4555 (+69.33%)	0.5407 (+79.04%)	0.6566 (+44.94%)

Section 2 - Related Work, for detailed classification). It is worth noting that recently there have been various learning-to-diversify approaches that make use of explicit aspect representation, i.e., R-LTR [29] and PAMM [12]. More recently, Xia et al. [26] proposed to model the document novelty with neural tensor network and applied it to existing learning framework for search result diversification. Our proposed approach is intrinsically different from them in the taxonomy of search diversification approaches (i.e., ours is an implicit approach, while R-LTR and PAMM are explicit approaches). Therefore, we do not directly conduct comparative experiments with them in the same experimental environment. Instead, we look at the experimental results reported in their papers [12][29][26]. As shown in Table 10 (in which, LM is the baseline used in this paper. Query Likelihood (QL) model is the baseline used in [12][29][26].), there is still a room for our approach to improve, compared with these recent explicit and learning based approaches. However, we would like to highlight their fundamental differences from ours as follows:

(i) The design of the ranking model is different. Specifically, they use a greedy ranking function based on the iterative sequential selection principle in both learning and testing processes, which leads to a high computational cost. Specifically, they need to update the weight values for a large number of times in learning process, and for each time, it is an iterative sequential selection process. On the other hand, we only use one greedy process to extract the diversity features, instead of using the unpredictable greedy process repeatedly, so that the learning and ranking in our approach would be more efficient. The time complexity for computing document scores in the learning process for R-LTR [29] and PAMM [12] is $O(M \cdot \frac{(N+1)N}{2})$, while ours is $O(M \cdot N)$, where M is the number of iterations[5], and N is the number of documents. Thus, our approach is more efficient than existing learning-to-diversify methods. Note that, we use the non-learning approaches (e.g., MMR, QPRP and AP, etc.) as diversity features and use the training algorithm to learn their weights, so that the computational complexity of our approach is larger than that of the non-learning approaches.

(ii) They extract a large number of features from various external sources of explicit knowledge (e.g., ODP, pagerank and anchor text), while we only use some representative features from queries and documents without requiring any external knowledge. Moreover, from the results reported in their papers [12, 29], we find that the initial relevance-based baseline search performance is different from ours, possibly due to different experimental setups and different pre-processing methods of the document collection, etc. In this sense, a direct empirical comparison of our model with the models proposed in [12, 29] would not be applicable nor meaningful. However, we are inspired to further improve our approach by utilizing some good features from these models.

[5] We assume that the number of iterations for R-LTR, PAMM and our DRM are the same, since they all use the gradient descent algorithm to reach a convergence for the cost function.

6. Conclusions and Future Work

In this paper, we have proposed a novel learning-to-diversify approach that directly optimizes the diversity metric to improve the effectiveness, robustness and efficiency of search diversification. A Document Repulsion Theory (DRT) is proposed, which assumes that two documents covering similar query aspects should be mutually repulsive. To implement DRT, an efficient learning algorithm is developed. Based on DRT and by extending a widely used learning-to-rank framework, i.e., LambdaMART, we propose a document repulsion model (DRM) for search results diversification. The inter-relationships between documents are captured by the diversity features, which are then combined with traditional relevance features to balance relevance and diversity in document ranking. Extensive experiments have shown that our approach is effective and robust in comparison with a number of existing learning-to-diversify approaches that also build upon LambdaMART.

Our model can be seen as an implicit diversity ranking approach based on the assumption that the similar documents cover similar query aspects. There have been recent work in explicitly modelling query aspects as diversity features. We are inspired to incorporate the explicit aspects into the DRM in the future.

Acknowledgements. This work is supported in part by the Chinese National Program on Key Basic Research Project (973 Program, grant No. 2014CB744604, 2013CB329304), the Chinese 863 Program (grant No. 2015AA015403), the Natural Science Foundation of China (grant No. U1636203, 61272265, 61402324), the Tianjin Research Program of Application Foundation and Advanced Technology (grant no. 15JCQNJC41700), and the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 721321.

Author Contributions. Theoretical study and proof: Yue Wu, Jingfei Li, Peng Zhang and Dawei Song; Conceived of and designed the experiments: Yue Wu, Jingfei Li and Benyou Wang; Performed the experiments: Yue Wu, Jingfei

Li and Benyou Wang; Analysed the data: Yue Wu and Jingfei Li; Wrote the manuscript: Yue Wu, Jingfei Li, Peng Zhang and Dawei Song.

References

[1] Agrawal, R., Gollapudi, S., Halverson, A., & Ieong, S. (2009). Diversifying search results. In Proceedings of the second ACM international conference on web search and data mining (pp. 5-14). ACM.

[2] Burges, C. J., Ragno, R., & Le, Q. V. (2006). Learning to rank with nonsmooth cost functions. In NIPS (Vol. 6, pp. 193-200).

[3] Burges, C., Shaked, T., Renshaw, E., Lazier, A., Deeds, M., Hamilton, N., & Hullender, G. (2005). Learning to rank using gradient descent. In Proceedings of the 22nd international conference on Machine learning (pp. 89-96). ACM.

[4] Carbonell, J., & Goldstein, J. (1998). The use of MMR, diversity-based reranking for reordering documents and producing summaries. In Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval (pp. 335-336). ACM.

[5] Capannini, G., Nardini, F. M., Perego, R., & Silvestri, F. (2011). Efficient diversification of web search results. Proceedings of the VLDB Endowment, 4(7), 451-459.

[6] Chapelle, O., Metlzer, D., Zhang, Y., & Grinspan, P. (2009). Expected reciprocal rank for graded relevance. In Proceedings of the 18th ACM conference on Information and knowledge management (pp. 621-630). ACM.

[7] Clarke, C. L., Kolla, M., Cormack, G. V., Vechtomova, O., Ashkan, A., Bttcher, S., & MacKinnon, I. (2008). Novelty and diversity in information retrieval evaluation. In Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval (pp. 659-666). ACM.

[8] Clarke, C. L., Kolla, M., & Vechtomova, O. (2009). An effectiveness measure for ambiguous and underspecified queries. In Conference on the Theory of Information Retrieval (pp. 188-199). Springer Berlin Heidelberg.

[9] Friedman, J. H. (2001). Greedy function approximation: a gradient boosting machine. Annals of statistics, 1189-1232.

[10] Hochba, D. S. (1997). Approximation algorithms for NP-hard problems. ACM SIGACT News, 28(2), 40-52.

[11] Kharazmi, S., Sanderson, M., Scholer, F., & Vallet, D. (2014, July). Using score differences for search result diversification. In Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval (pp. 1143-1146). ACM.

[12] Xia, L., Xu, J., Lan, Y., Guo, J., & Cheng, X. (2015). Learning maximal marginal relevance model via directly optimizing diversity evaluation measures. In Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval (pp. 113-122). ACM.

[13] Qin, T., Liu, T. Y., Xu, J., & Li, H. (2010). LETOR: A benchmark collection for research on learning to rank for information retrieval. Information Retrieval, 13(4), 346-374.

[14] Radlinski, F., & Dumais, S. (2006). Improving personalized web search using result diversification. In Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval (pp. 691-692). ACM.

[15] Radlinski, F., Kleinberg, R., & Joachims, T. (2008). Learning diverse rankings with multi-armed bandits. In Proceedings of the 25th international conference on Machine learning (pp. 784-791). ACM.

[16] Radlinski, F., Bennett, P. N., Carterette, B., & Joachims, T. (2009). Redundancy, diversity and interdependent document relevance. In ACM SIGIR Forum (Vol. 43, No. 2, pp. 46-52). ACM.

[17] Robertson, S. E. (1977). The probability ranking principle in IR. Journal of documentation, 33(4), 294-304.

[18] Santos, R. L., Macdonald, C., & Ounis, I. (2010). Exploiting query reformulations for web search result diversification. In Proceedings of the 19th international conference on World wide web (pp. 881-890). ACM.

[19] Santos, R. L., Macdonald, C., & Ounis, I. (2011). On the suitability of diversity metrics for learning-to-rank for diversity. In Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval (pp. 1185-1186). ACM.

[20] Santos, R. L., Macdonald, C., & Ounis, I. (2015). Search result diversification. Foundations and Trends in Information Retrieval, 9(1), 1-90.

[21] Yue, Y., & Joachims, T. (2008). Predicting diverse subsets using structural SVMs. In Proceedings of the 25th international conference on Machine learning (pp. 1224-1231). ACM.

[22] Wang, J., & Zhu, J. (2009). Portfolio theory of information retrieval. In Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval (pp. 115-122). ACM.

[23] Wang, L., Bennett, P. N., & Collins-Thompson, K. (2012). Robust ranking models via risk-sensitive optimization. In Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval (pp. 761-770). ACM.

[24] Wu, Q., Burges, C. J., Svore, K. M., & Gao, J. (2010). Adapting boosting for information retrieval measures. Information Retrieval, 13(3), 254-270.

[25] Wu, Y., Li, J., Zhang, P., & Song, D. (2016). Learning to Improve Affinity Ranking for Diversity Search. In Information Retrieval Technology (pp. 335-341). Springer International Publishing.

[26] Xia, L., Xu, J., Lan, Y., Guo, J., & Cheng, X. (2016). Modeling Document Novelty with Neural Tensor Network for Search Result Diversification. In Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval (pp. 395-404). ACM.

[27] Zhai, C. X., Cohen, W. W., & Lafferty, J. (2003). Beyond independent relevance: methods and evaluation metrics for subtopic retrieval. In Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval (pp. 10-17). ACM.

[28] Zhang, B., Li, H., Liu, Y., Ji, L., Xi, W., Fan, W., ... & Ma, W. Y. (2005). Improving web search results using affinity graph. In Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval (pp. 504-511). ACM.

[29] Zhu, Y., Lan, Y., Guo, J., Cheng, X., & Niu, S. (2014). Learning for search result diversification. In Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval (pp. 293-302). ACM.

[30] Zuccon, G., & Azzopardi, L. (2010). Using the quantum probability ranking principle to rank interdependent documents. In European Conference on Information Retrieval (pp. 357-369). Springer Berlin Heidelberg.