Result Diversification Based on Query-Specific Cluster Ranking

Jiyin He, Edgar Meij, and Maarten de Rijke
ISLA, University of Amsterdam, Science Park 904, 1098 XH Amsterdam, The Netherlands.
E-mail: {j.he, edgar.meij, derijke}@uva.nl

Result diversification is a retrieval strategy for dealing with ambiguous or multi-faceted queries by providing documents that cover as many facets of the query as possible. We propose a result diversification framework based on query-specific clustering and cluster ranking, in which diversification is restricted to documents belonging to clusters that potentially contain a high percentage of relevant documents. Empirical results show that the proposed framework improves the performance of several existing diversification methods. The framework also gives rise to a simple yet effective cluster-based approach to result diversification that selects documents from different clusters to be included in a ranked list in a round robin fashion. We describe a set of experiments aimed at thoroughly analyzing the behavior of the two main components of the proposed diversification framework, ranking and selecting clusters for diversification. Both components have a crucial impact on the overall performance of our framework, but ranking clusters plays a more important role than selecting clusters. We also examine properties that clusters should have in order for our diversification framework to be effective. Most relevant documents should be contained in a small number of high-quality clusters, while there should be no dominantly large clusters. Also, documents from these high-quality clusters should have a diverse content. These properties are strongly correlated with the overall performance of the proposed diversification framework.

Introduction

Queries submitted to Web search engines are often ambiguous or multi-faceted in the sense that they have multiple interpretations or sub-topics (Allan & Raghavan, 2002). For ambiguous queries, a typical example is the query “jaguar” that can refer to several interpretations including a kind of animal, a car brand, a type of cocktail, an operating system, etc. Multi-faceted queries are even more commonly seen in practice; for example, for the interpretation “jaguar car” of the query “jaguar”, a wide range of sub-topics may be covered: models, prices, history of the company, etc. For such queries we often cannot be certain what the searcher’s underlying information need is because of a lack of context. One retrieval strategy that attempts to cater for multiple interpretations of an ambiguous or multi-faceted query is to *diversify* the search results (Boyce, 1982; Goffman, 1964). Without explicit or implicit user feedback or history, the retrieval system makes an educated guess as to the possible facets of the query and presents as diverse a result list as possible by including documents pertaining to different facets of the query within the top-ranked documents.

Recently, various result diversification methods have been proposed (Agrawal, Gollapudi, Halverson, & Ieong, 2009; Carbonell & Goldstein, 1998; Carterette & Chandar, 2009; Chen & Karger, 2006; Radlinski, Kleinberg, & Joachims, 2008; Santos, Macdonald, & Ounis, 2010; Zhai, Cohen, & Lafferty, 2003). Traditional retrieval strategies such as those based on the Probabilistic Ranking Principle (Robertson, 1997) typically assume that the relevance of a document is independent from the relevance of other documents in the collection. In contrast, in the context of result diversification, the notion of “relevance” usually reflects not only the relation between a document and a given query, but also the relation between the document and other documents retrieved in response to the query. Indeed, most of the proposed diversification methods simultaneously explore *query–document* and *document–document* relations and seek to balance the two in order to address both relevance and diversity in returning retrieval results. A prime example hereof is the Maximal Marginal Relevance (MMR) approach (Carbonell & Goldstein, 1998), which iteratively selects documents that are most similar to the query while at the same time being most dissimilar to the documents already returned. An obvious risk with this type of diversification method is that non-relevant documents may be promoted to the top of a ranked list simply because they are different from the documents presented so far. We illustrate this phenomenon using Figure 1. We use MMR to rank documents for the test queries in the TREC 2009 Web track test collection (Clarke,

Received July 8, 2010; revised October 15, 2010; accepted October 18, 2010

© 2010 ASIS&T • Published online 18 January 2011 in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/asi.21468

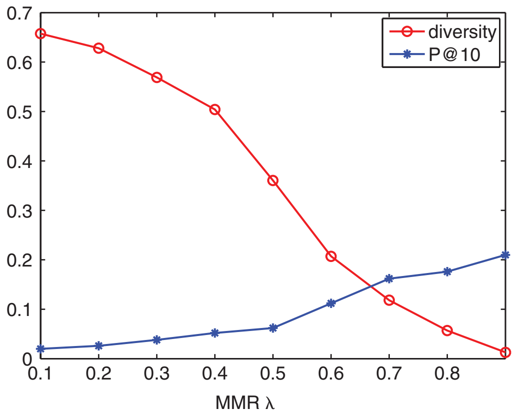


FIG. 1. The trade-off between diversity and precision@10 for the top10 documents retrieved with MMR over different values of λ. The *Y*-axis shows both the precision@10 (P@10) and coherence score; both scores are in the range of [0,1].

Craswell, & Soboroff, 2010). In Figure 1, we plot three things: the change of λ, the parameter in MMR that balances relevance and diversity; Precision@10 to measure relevance; diversity, measured as one minus the Coherence Score (He, Weerkamp, Larson, & de Rijke, 2009).[1] Observe the inverse relation between diversity and relevance of the top 10 documents as we change λ. As λ increases, i.e., the emphasis on relevance is increased, there is an increase in the precision but a drop in the diversity, and vice versa. Ideally, a retrieval system should find the middle ground and present users with a ranked list which is both *relevant* and *diverse*.

Query-specific cluster-based retrieval is the idea of clustering retrieval results for a given query. It has long been proposed for improving retrieval effectiveness (Hearst & Pedersen, 1996; Jardine & van Rijsbergen, 1971; Kurland, 2006; Tombros, Villa, & Van Rijsbergen, 2002). The main intuition behind this approach to retrieval is that relevant documents tend to be clustered together. Retrieval effectiveness will be improved provided that one can place documents from high-quality clusters at the top of the ranked list. Now consider a ranking approach based on query-specific cluster-based retrieval in the context of result diversification. What if we first select a set of high-quality clusters (a relatively large fraction of whose documents is relevant) and then apply diversification only to the documents within these clusters? That is, what happens if we prevent documents in low-quality clusters (with a limited number of relevant documents) from being promoted to the top ranks? We posit that such a strategy should lead to improved results as measured in terms of relevance and diversity because it only diversifies relevant documents. Specifically, we focus on the following question in this article:

Can query-specific clustering be used to improve the effectiveness of result diversification?

To answer this question, we propose the following diversification framework. Given a query, we first cluster top-ranked documents that are retrieved in response to the query. We subsequently rank the clusters according to their estimated relevance to the query and apply a diversification method to the documents belonging to the top-ranked clusters only. Below, we refer to this framework as *diversification with cluster ranking*.

In order to gain insight into the behavior of our proposed diversification framework, a number of specific research questions need to be addressed:

Q-1 What is the impact of the proposed diversification framework on the effectiveness of existing result diversification methods? In other words, how much performance is gained by employing query-specific clustering and applying result diversification to documents contained in the top-ranked clusters only?

Q-2 What is the impact of the two main components, namely, the cluster ranker and the selection of number of top-ranked clusters, on the overall performance of the proposed diversification framework?

Q-3 Further, given that we use top-ranked documents retrieved in response to a query for clustering as well as for diversification, how sensitive is the performance of the proposed framework to the number of documents being selected?

Q-4 What conditions should clusters fulfill in order for diversification with cluster ranking to be effective?

We answer these research questions using empirical methods on Web data that has been made available through the TREC 2009 Web track (Clarke et al., 2010). Several features make this test collection appealing for our task, including the size of the document collection and the fact that the queries are derived from query logs. The most important feature, however, is that the track launched a dedicated diversity task that provides queries as well as relevance judgements that are specifically designed for measuring the performance of retrieval systems in terms of diversity. For a detailed description of the collection, see the section Test collection and queries.

The main contribution of the article is two-fold. We propose a diversification framework that combines cluster-based retrieval and result diversification. The framework significantly improves the effectiveness of several result diversification methods. On top of that, we provide an in-depth analysis of the behavior of our proposed framework as well as the relation between relevance, diversity and query-specific clustering methods. Our analyses do not only help to understand the behavior of diversification with cluster ranking, but also help to direct future work on the proposed framework.

The remainder of the article is organized as follows. In the next section, we discuss related work on result diversification and cluster-based retrieval. We then specify the methods employed for clustering and result diversification in

[1] The coherence score was proposed to measure the "tightness" of a cluster of documents. It takes values between 0 and 1; a high coherence score of a cluster indicates that a large fraction of the documents within the cluster are highly similar to each other. For more details of the coherence score, see (He et al., 2009). Here we use one minus the coherence score of a set of documents so as to measure its "looseness," i.e., diversity. Detailed explanation of the score can be found in the section Preliminaries.

DOI: 10.1002/asi

the section Preliminaries. We introduce our proposed framework for diversification with cluster ranking in the section Result Diversification with Cluster Ranking. We describe our experimental setup in the next section. We report on the effectiveness of diversification with cluster ranking based on our empirical results in the section Experimental Results. We then proceed with two rounds of analysis. We provide a set of sensitivity analysis in the section Sensitivity Analysis. We analyze the impact of the main components of our framework, that is, of methods for ranking and selecting clusters, as well as the impact of the number of documents being used for clustering and for diversification, on the overall performance of the proposed framework. We analyze the conditions that clusters should fulfill in order for our proposed framework to be effective in the section Impact of Clustering Structure. Conclusion is given in the last section.

Background

We survey previous work on result diversification and on query-specific clustering.

Result Diversification

Diversification of search results has been recognized by many as an important issue (Boyce, 1982; Goffman, 1964). Zhai et al. (2003) argue that it is insufficient to simply return a set of relevant documents where relevance of a document is treated independently from other retrieved documents, an observation that gives rise to new evaluation metrics and retrieval strategies that consider dependence among documents. Chen and Karger (2006) investigate the scenario where the user is satisfied with a limited number of relevant documents instead of all relevant documents. They show that in such a scenario, it is more effective to optimize the expected value of a given metric and to rank documents in such a way that the probability of finding at least a relevant document among the top *N* is maximized. On top of that, they find that explicitly aiming to find only one relevant document inherently promotes diversity of documents at the top of a ranked list.

An early diversification method is MMR in which the merit of a document in the ranked list is computed as a linear combination of its similarity to the query and the smallest similarity to documents already returned (Carbonell & Goldstein, 1998). Zhai and Lafferty (2006) propose a risk minimization framework in which loss functions are defined according to different assumptions about relevance so as to minimize the user's average "unhappiness." A probabilistic version of MMR is proposed within this framework, a mixture model of novelty and relevance. Carterette and Chandar (2009) propose a probabilistic facet retrieval model for diversification, with the assumption that users are interested in all facets that are potentially related to the query and thus all hypothesized facets are equally important.

Radlinski et al. (2008) propose a method that learns a diverse ranking of retrieval results from users' clicks. Yue and Joachims (2008) study a learning algorithm based on structure SVM that identifies diverse subsets in a given set of documents.

Agrawal et al. (2009) propose a diversification method, *IA-select*, which uses the Open Directory Project to model facets associated with query and documents. IA-select is interesting in our context because it takes into account the importance of individual user intentions. Below, we employ IA-select as one of the diversification methods we experiment with for our diversification with cluster ranking framework.

Recently, Santos et al. (2010) explore query reformulation for result diversification. Similar to IA-select, during the diversification procedure, merit of a single document is estimated base on its relevance to the query, its coverage of the query aspects and its novelty to other retrieved documents. The difference is that underlying facets associated with a query is uncovered in the form of sub-queries.

Query-Specific Clustering

The *cluster hypothesis* is the hypothesis that *closely associated documents tend to be relevant to the same requests* (Hearst & Pedersen, 1996; Jardine & van Rijsbergen, 1971; van Rijsbergen, 1979). It has given rise to many cluster-based retrieval methods (Croft, 1980; Hearst & Pedersen, 1996; Jardine & van Rijsbergen, 1971; Kurland, 2006, 2008, 2009; Kurland & Domshlak, 2008; Liu & Croft, 2004, 2006a,b, 2008; Tombros et al., 2002; Willett, 1988; Yang, Ji, Zhou, Yu, & Xiao, 2006). These methods were shown to be able to improve the effectiveness of retrieval performance in an ad hoc retrieval setting. The rationale behind cluster-based retrieval is that, since similar documents tend to fulfill similar information requests, relevant documents are likely to be more similar to each other than to non-relevant documents, and are therefore likely to be clustered together. Through clustering, more relevant documents can be found and promoted to the top of the ranked list.

An interesting type of clustering often used in the context of retrieval is query-specific clustering, which aims to improve the retrieval effectiveness via clustering search results in response to a given query (Hearst & Pedersen, 1996; Kurland, 2006; Kurland & Domshlak, 2008; Liu & Croft, 2004; Tombros et al., 2002). Search result clustering may be a suitable solution in specialized environments. For instance, the traditional list-based search interface paradigm does not scale well to mobile devices due to their inherent limitations; here, query-specific clustering may be a viable solution (Carpineto, Mizzaro, Romano, & Snidero, 2009).

As illustrated by Hearst and Pedersen (1996) and Kurland and Domshlak (2008), with a proper clustering algorithm, one can generate clusters such that a large percentage of the relevant documents retrieved are contained in a few *high-quality* clusters. If we would be able to identify those clusters for a given query and place the documents they contain at the top of the ranking, retrieval performance can be substantially improved in terms of early precision. While query-specific clustering methods aim to improve retrieval effectiveness as measured using standard precision and recall-based metrics, we explore the merits of query-specific clustering for result diversification. In particular, while many diversification approaches attempt to strike a balance between relevance and diversity, query-specific clusters are appealing in that they provide documents that are potentially both relevant (as they come from *high-quality* clusters) and diverse (as they come from *different* clusters). Below, we focus on determining how query-specific clustering can be employed for result diversification.

Preliminaries

We start by introducing the notation that we employ in the remainder of the article and then detail the clustering and diversification methods that we consider.

Notation

Let d, q, and D denote a document, query, and set of documents, respectively. Given q, we write D_q^{R} and D_q^{NR} to refer to the explicitly judged relevant documents for q and the explicitly judged non-relevant documents for q, respectively. We write D_q^n for the top n documents retrieved in response to q. In D_q^n we identify a set of K clusters, $C = \{c_k\}_{k=1}^K$. We use the notation $d \in c_k$ to denote the assignment of document d to cluster c_k and write $D_q^{c_k}$ for the set of documents that belong to cluster c_k.

Clustering Method

For clustering the documents that have been retrieved in response to a query, we use latent Dirichlet allocation (LDA) (Blei, Ng, & Jordan, 2003). We perform clustering with LDA as follows. First, we train the topic models over D_q^n with a pre-fixed number of K clusters (or latent topics). We then assign each document to a single cluster based on the topic distribution given a document. In other words, a document d is assigned to a cluster c^* such that

$$c^* = \arg\max_c \, p(c|d) \tag{1}$$

where $p(c|d)$ is estimated using the LDA model.

We choose LDA for two main reasons. First, it models the relation between words, documents, and clusters (that is, latent topics) within a theoretically sound probabilistic framework. Once the topic models have been obtained, it is convenient to infer the generation probability of a cluster for an arbitrary piece of text. In our case, we use the trained model to infer the probability of a cluster generating a query as an estimation of the relevance relation between the cluster and the query, see the section Cluster ranking. Second, the latent topics can be seen as the potential "facets" of a query. Although the main purpose of applying query-specific clustering is to gather relevant documents instead of modeling query facets, the latent topic underlying a cluster addresses both. Particularly, we can apply the same LDA model for facet modeling when implementing diversification methods that explicitly model the potential facets of a query, as we will see in the next section. We discuss diversification performance using clustering algorithms other than LDA in the section Impact of Clustering Structure.

Diversification Methods

In our experiments we consider the following diversification methods: MMR, FM-LDA, IA-select, and RR. These will be explained next.

MMR. According to the MMR method (Carbonell & Goldstein, 1998), a document d is selected for inclusion in a ranked list of documents for a given query q such that

$$d = \arg\max_{d_i \in R} \, [\lambda \cdot sim_1(d_i, q) - (1-\lambda) \cdot \max_{d_j \in S} sim_2(d_i, d_j)] \tag{2}$$

where S is the set of documents that have been selected so far and R is the set of candidate documents to be selected; sim_1 is the similarity between query and document and sim_2 is the similarity between two documents. For sim_1 and sim_2 we can use any type of similarity measure; we specify our choices in the section Parameter Settings.

Facet model with LDA (FM-LDA). We also consider the FM-LDA model (Carterette & Chandar, 2009), with marginal likelihood as optimization method. Given a set of documents $D = \{d_i\}_{i=1}^n$, the model uses LDA to capture a set of hypothesized facets $F = \{f_j\}_{j=1}^m$, and a subset of D is selected such that the following likelihood function is maximized:

$$L(y_i|F, D) = \prod_{j=1}^{m} \left(1 - \prod_{i=1}^{n} (1 - p(f_j \in d_i))^{y_i}\right) \tag{3}$$

where $y_i = 1$ if document d_i is selected and $y_i = 0$ otherwise; $p(f_j \in d_i)$ denotes the probability that facet f_j is covered by document d_i. The likelihood function is maximized subject to the constraint $\sum y_i \leq l$, where l is a predefined number of documents that are to be returned in the ranked list. In practice, a greedy approach is applied, which selects a document that maximizes the likelihood function conditioned on all the documents that have already been selected.

Note that FM-LDA identifies facets of a query with LDA, which is very similar to how our document clustering method identifies clusters, cf. section Clustering method. There are two distinguishing differences. First, the underlying assumptions on latent topics are different: in FM-LDA, the trained latent topics are expected to reflect the underlying facets of a query, while in document clustering, we do not care whether the latent topics can accurately reflect the actual facets of a query. Second, in document clustering, we assign each document to a single cluster, cf. Equation (1), while in FM-LDA there is no need for assigning documents to latent topics. In addition, as we see from Equation (3), FM-LDA treats all facets as identified by LDA in the same manner. Contrary to

DOI: 10.1002/asi

our method, FM-LDA does not consider the importance of a facet, that is, some facets may be more relevant than others. If we assume that document clusters reflect the potential facets of a query, our method takes into account the importance of each facet via cluster ranking. We will see the impact of ranking clusters on the diversification result of FM-LDA in the section Experimental Results.

Intent Aware select (IA-select). With IA-select (Agrawal et al., 2009) the selection of a document is determined by its relevance to the query as well as the probability that it satisfies potential facets given that all previously selected documents fail to do so. Given a candidate document set and a set of potential facets F, the algorithm selects the document to be included in the returned set S from a candidate set R that maximizes the *marginal utility* at each step:

$$d^* = \arg\max_{d \in R} \sum_{f \in F} P(f_i|q, S)\ V(d|q, f_i) \tag{4}$$

where $V(d|q,f)$ is a quality value of d that is computed using the retrieval score of d with respect to q, weighted by the likelihood that d belongs to f. Further, $P(f|q,S)$ is the conditional probability that q belongs to f, given that all documents in S failed to provide information on f:

$$P(f|q, S) = (1 - V(d|q, f))P(f|q, S\backslash d) \tag{5}$$

Instead of training a classifier with a taxonomy as implemented in the original IA-select algorithm to obtain $P(f|q,S)$ and the likelihood that d belongs to f, we estimate these probabilities with the topic distribution from the LDA model. Similar to FM-LDA, we use the same LDA model for clustering and facet modeling.

Round-Robin facet selection (RR). This approach naturally arises in the setting of our strategy for diversification with cluster ranking (defined in the next section). For a given set of documents D, we generate a set of K clusters with LDA, and rank the clusters according to a certain ranking criterion, for example, the relevance of the clusters to a given query, which results in a ranked list of clusters $\mathrm{RC} = c_1, \ldots, c_k$, where $c_1 \succ c_2 \cdots \succ c_k$. For each cluster, we rank the documents within that cluster in the order of their original retrieved scores. We then select documents belonging to different clusters in a RR fashion. That is, in each round, we take the top-ranked documents from each of the clusters, and add them to the new ranked list in the order of $c_1, \ldots, c_k$. This selection procedure continues until no documents are left in any of the clusters. The motivation behind this approach is as follows. By clustering documents, we gather documents with similar content within the same cluster, whereas documents from different clusters contain diverse content. Intuitively, we can see the clusters as different facets associated with a given query. Hence, selecting documents from different clusters should potentially result in a diverse result list. On top of that, by selecting the documents in the order of the ranking of the clusters, we take into account the importance of different facets.

Result Diversification with Cluster Ranking

In this section, we introduce our proposed framework for combining query-specific clustering and result diversification. The overall goal of the approach is to *rank clusters with respect to their relevance to the query and to limit the diversification process to documents contained in the top-ranked clusters only, in order to improve the effectiveness of diversification as measured in terms of both relevance and diversity.*

Proposed Framework

Assume that we have a ranking method $cRanker(\cdot)$ that ranks clusters with respect to their relevance to a query and a diversification method $\mathrm{Div}(\cdot)$ that diversifies a given ranked list of documents. We propose the following procedure for diversification. The input of the procedure is the output of *cRanker*, that is, a ranked set of clusters $\mathrm{RC} = c_1, \ldots, c_K$, where $c_1 \succ c_2 \succ \cdots \succ c_K$, and the documents contained in each cluster, D_q^c. A free parameter T is used to indicate the number of top-ranked clusters to be selected for diversification. Furthermore, $dRanker(\cdot)$ is assumed to be a document ranker that ranks documents according to certain criteria, for example, ranking documents in descending order of their retrieval scores. We illustrate the proposed diversification framework in Figure 2.

The pseudocode of our diversification with cluster ranking method is given in Algorithm 1. It applies $\mathrm{Div}(\cdot)$ to the documents assigned to the top T ranked clusters; documents assigned to clusters ranked below the top T are ranked by $dRanker(\cdot)$ and appended to the ranked list of documents obtained from the top T clusters.

Two crucial components of our proposed diversification framework are the function $cRanker(\cdot)$ that ranks the clusters and the selection of T. In the following sub-sections, we discuss our choices for these two components. As for $Div(\cdot)$, we use the diversification approaches introduced in the section Diversification methods.

Cluster Ranking

As we pointed out above, ranking clusters based on their relevance to a query is an important issue, which has been studied in the context of cluster-based retrieval. Since our main purpose is not to develop a new method for ranking clusters, we only discuss two ways to rank clusters that are necessary for investigating the effectiveness of our proposed framework for result diversification.

Query likelihood. For a query q, we rank the clusters in descending order of the probability $p(c|q)$, which is inferred from the LDA model as described in the section Clustering

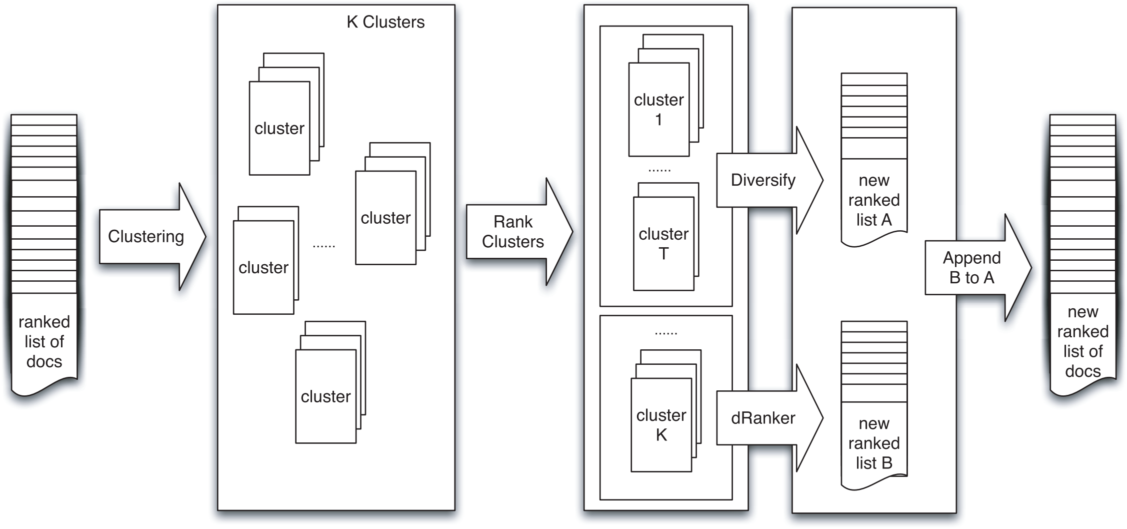


FIG. 2. Diversification with cluster ranking. The input is a ranked list of documents and output is a diversified ranked list of documents. The arrows represent methods applied to the documents, and the boxes show the status of the documents.

Algorithm 1. Diversification with cluster ranking

```
    Input: $Div(\cdot)$, $RC = c_1, \ldots, c_k$, $\{D_q^c\}$, $T$
    Output: re-ranked documents $ranked$
    $ranked = \emptyset$
    $to\_rank = \{D_q^{c_i}\}_{i=1}^{T}$
    $ranked \leftarrow ranked \cup Div(to\_rank)$
    for $i$ in $T+1$ to $k$ do
    $ranked \leftarrow ranked \cup dRanker(D_q^{c_i})$
  end for
  return $ranked$
```

method. In other words, the clusters are ranked according to their likelihood given the query. This is a simple but reasonable approach. Presumably, if a cluster has a high probability to generate a query, the documents contained in this cluster are more likely to be relevant to the query. Hence, the cluster is more likely to contain relevant documents.

Oracle ranker. We also consider an oracle ranker, that is, a ranker that uses information from explicit relevance judgements. Here, the probabilities $p(c|q)$ are estimated using the judgments of retrieved documents in D_q^n. It is computed as:

$$p(c|ora_q) = \frac{|D_q^c \cap D_q^R|}{|D_q^c|}. \tag{6}$$

In words, using $p(c|ora_q)$, we rank clusters according to the number of relevant documents contained in them, normalized by the size of the cluster.

Observe that Equation (6) combines two important factors: the number of relevant documents in c_k and its relative size. Intuitively, we hope that the top-ranked clusters contain most of the relevant documents, which is not achieved by simply assigning most of the documents to a single huge cluster. We discuss this issue, that is, properties of the clustering structure desired by our proposed framework, in more detail in the section Impact of Clustering Structure.

Determining the Cut-Off T

The optimal number of top-ranked clusters whose documents will be used for diversification, T, depends on a number of factors: the diversification method, the total number of clusters (that is, K), the evaluation metric, as well as the query. Similar to our strategy for ranking clusters, we discuss two ways to determine the value of T, namely, automatically determining T with cross-validation and using an oracle.

Automatically determining T using cross-validation over queries. Automatically determining the optimal cut-off T is non-trivial. We typically do not have sufficiently many test queries to learn the optimal value of T, hence we apply leave-one-out cross-validation to find the optimal value of T for each query. Specifically, we optimize T over a set of training queries for a given K and a given diversification method for a given evaluation metric by exhaustive search, i.e., over all possible values of $T = 1, \ldots, K$. Then we apply the learned T on the test query.

Oracle T. To obtain the oracle value of T, for each query, we find the optimal T for each diversification method and each setting of K over each single evaluation metric, i.e., the performance is maximized in terms of a corresponding evaluation metric. For a given cluster ranking approach, the oracle T provides an upper bound for our proposed diversification framework under the given setting, which shows the potential merit of applying cluster ranking and selection for result diversification.

Experimental Setup

In this section, we describe our experimental setup for investigating the effectiveness of diversification with cluster ranking. We begin by recalling the research questions raised in the Introduction. Then we specify the settings for our experiments, including the document collection and test queries, the evaluation metrics and the parameter settings for the retrieval method, diversification methods, and clustering algorithms.

Research Questions and Experiments

The main research question we address in this article is:
Can query-specific clustering be used to improve the effectiveness of result diversification?
More specifically, we investigate the following:

Q-1 What is the impact of diversification with cluster ranking on the effectiveness of existing result diversification methods? In other words, how much performance is gained by employing query-specific clustering and applying result diversification to documents contained in the top-ranked clusters? In particular, given the query likelihood cluster ranker and an automatically determined value of T, what is the effectiveness of the proposed diversification framework?

We apply Algorithm 1 with various diversification methods, i.e., various instances of Div$(\cdot)$, on an initially retrieved ranked list of documents D_q^n where $n = 1000$. We write cX to denote the instance of Algorithm 1 where X is used as Div$(\cdot)$. First, we take the cluster ranker based on query likelihood (section Cluster ranking), and investigate whether the proposed diversification framework is effective even though the ranking of clusters may not be optimal. We set T to different values and compare the results of only diversifying over documents contained in the top T clusters to the result of diversifying over the complete ranked list of documents. Then, in order to evaluate the performance of our framework combined with the query likelihood cluster ranker and the automatically determined T, we use cross-validation as described in section Determining the cut-off T to determine the optimal T for each diversification and a given K. While optimizing T on training queries, we use two evaluation metrics: α-NDCG@10 for α-NDCG-based metrics and IA-P@10 for IA-P based metrics (see the section Evaluation metrics for a description of these evaluation metrics).

We analyze the effectiveness of diversification with cluster ranking along four dimensions: the cluster rankers used, the cut-off value T, the number of documents used for clustering as well as for diversification, and the clustering algorithms used. In our experiments, the query likelihood cluster ranker and the method to automatically determine T are chosen for simplicity, while many other possibilities exist. Insights into the roles of both components and their interactions within our proposed framework are useful for future work on potentially more effective approaches to ranking clusters and to automatically determining T. In addition, the number of documents being included from the initially retrieved ranked list for clustering and for diversification can be seen as an additional free parameter. We provide a comprehensive analysis for the sensitivity of the proposed framework to the choice of this parameter. In addition, LDA is used for clustering for the reasons as stated in section Clustering method. It is useful to examine the general properties of the sort of clustering structure desired by Algorithm 1, as this provides guidance for choosing suitable clustering algorithms. Specifically, then, we seek to answer the following additional research questions:

Q-2 What is the impact of the two main components, namely, the cluster ranker and the selection of number of top-ranked clusters, on the overall performance of diversification with cluster ranking?

Q-3 Further, given that we use top-ranked documents retrieved in response to a query for clustering as well as for diversification, how sensitive is the performance of the proposed framework to the number of documents being selected?

Q-4 What conditions should clusters fulfill in order for diversification with cluster ranking to be effective?

In order to answer Q-2, we conduct a set of "oracle" runs in three settings. First, we analyze the impact of T by comparing the diversification results using the oracle T and the predicted T determined by cross-validation. Then, we analyze the impact of the cluster ranker by comparing the diversification performance using the oracle *cRanker*$(\cdot)$ as described in the section Cluster ranking to that of the query likelihood-based cluster ranker. In addition, we combine the oracle cluster ranker and the oracle T so as to identify an upper bound on the improvement of diversification with cluster ranking over diversification without cluster ranking and selection; See the section Impact of the cluster ranker and T.

In order to answer Q-3, we continue using the oracle cluster ranker and conduct a set of three experiments with varying number of documents for clustering and for diversification. Given an initially retrieved ranked list of documents, let us refer to the documents used for clustering, i.e., training the LDA model, as D_q^C and the documents on which we apply Algorithm 1 as D_q^D. The settings of the three experiments can be described as follows.

Setting 1. Set $C = 100{,}300{,}500$ and $D = 1{,}000$. In this setting, we fix the number of documents used for applying Algorithm 1 and compare the impact of LDA models trained on different number of top-ranked documents on our proposed diversification framework.

Setting 2. Set $C = 500$ and $D = 100{,}300{,}1{,}000$. In this setting, we fix the number of documents used for training the LDA model and analyze the effect of applying Algorithm 1 on different number of top-ranked documents.

Setting 3. Set $C = 100{,}300{,}500$ and $D = 100{,}300{,}1{,}000$, respectively. In this setting, we check the performance of our proposed framework by varying the number of documents for both clustering and for diversification simutanuously.

Note that in each setting, when $C = 500$ and $D = 1{,}000$, it is the default parameter setting of the experiments discussed above (see the section Parameter settings) and we use the results of this parameter setting as baselines in our analysis. See the section Length effect for details.

In order to answer Q-4, we hypothesize conditions that should be fulfilled by the clustering structure generated by a cluster algorithm based on the literature in cluster-based retrieval as well as the characteristic of the diversification task. On top of that, we include hierarchical clustering as an alternative clustering algorithm such that it generates a clustering structure different from that generated by LDA. We examine the impact of the conditions on clustering structure by comparing the properties of the two types of clustering structure and the end performance of our diversification with cluster ranking framework. See the section Impact of Clustering Structure for details.

Test Collection and Queries

As our test collection we use the Category B subset of the ClueWeb09 dataset.[2] It consists of 50 million English pages and was employed as the test collection at the TREC 2009 Web Track (Clarke et al., 2010). As our queries, we use the TREC 2009 Web Track query set from the diversity task, which contains 50 queries, each of which comes with a set of subtopics created from query logs to reflect different facets associated with the query. While relevance judgements were made with respect to each subtopic, retrieval systems only receive a keyword query as input, i.e., short queries that usually consist of one or a few words.

Evaluation Metrics

For evaluation, we use α-NDCG (Clarke et al., 2008), which adapts the NDCG measure to address both relevance and diversity. The parameter α denotes the probability that a user is still interested in a document given that the facet associated with the document is already covered by previously seen documents. By default, we set α to 0.5. In addition, we use the IA-P measure (Clarke et al., 2010) with a uniform distribution for judged facets.

A one-sided paired t-test is used for testing the significance of the difference between run results as indicated in the captions: $^{\triangle}$ ($^{\triangledown}$) indicates that an improvement (decline) is significant with $\alpha < 0.05$; $^{\triangle}$ ($^{\triangledown}$) indicates that an improvement (decline) is significant with $\alpha < 0.01$.

Parameter Settings

Settings for retrieval. For our baseline retrieval method, we use the Markov Random Field (MRF) retrieval model (Metzler & Croft, 2005); we use the full dependency model implemented by the indri search engine[3] with default parameter settings. All follow-up clustering and re-ranking methods that involve an initially retrieved list of documents use the results generated by the MRF model.

Settings for clustering. For each query in our test set, we retrieve a ranked list of 1,000 documents using MRF and identify clusters with LDA, as described previously. We set the number of clusters, K, to 10, 30, and 50, in order to check for the effectiveness of our diversification with cluster ranking using different numbers of clusters. For training the latent topic models, following Carterette and Chandar (2009) we use the top 500 documents as D_q^n to estimate the LDA model parameters with Gibbs sampling (Geman & Geman, 1984; Griffiths & Steyvers, 2004) and then infer the latent topic generation probabilities for all 1,000 documents.

Settings for diversification. The diversification methods that we consider come with the following model parameters:

MMR. For sim_1 we normalize retrieval scores into [0,1] (see below); for sim_2, we use cosine similarity. To determine λ, we performed a simple parameter sweep by applying MMR without cluster ranking and use α-NDCG@10 as the optimization metric, that is, we chose the λ that generates the best result in terms of α-NDCG@10; λ was found to be 0.9. Optimization is performed with respect to diversification with entire ranked list.

IA-select. We model the distribution of facets of a query with the cluster distribution inferred by LDA (see the section Diversification methods). Specifically, the importance of a cluster, that is, facet, for a query q is determined by $p(c|q)$, which is inferred from the trained LDA model.

FM-LDA. Similar to IA-select, facets of a query are discovered by LDA; the only parameter is the number of facets.

RR. We order clusters by descending value of $p(c|q)$ which is inferred in the same way as for IA-select.

Score normalization. For MMR and IA-select, the original retrieval scores are involved for diversification. In our experiments, we normalize those scores into range [0, 1] in order to combine scores with different ranges. Since the original retrieval score is usually in the log domain, we first transform it back to its original domain, and then for the score of each document s_d in the ranked list D_q^n, we normalize it using $\text{norm}(s_d) = s_d / \sum_{i \in D_q^n} s_i$

Experimental Results

In this section, we discuss the results of our experiments that aim to answer the main research question: *Can query-specific clustering be used to improve the effectiveness of result diversification using diversification with cluster ranking?*

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

[3] http://www.lemurproject.org/indri/

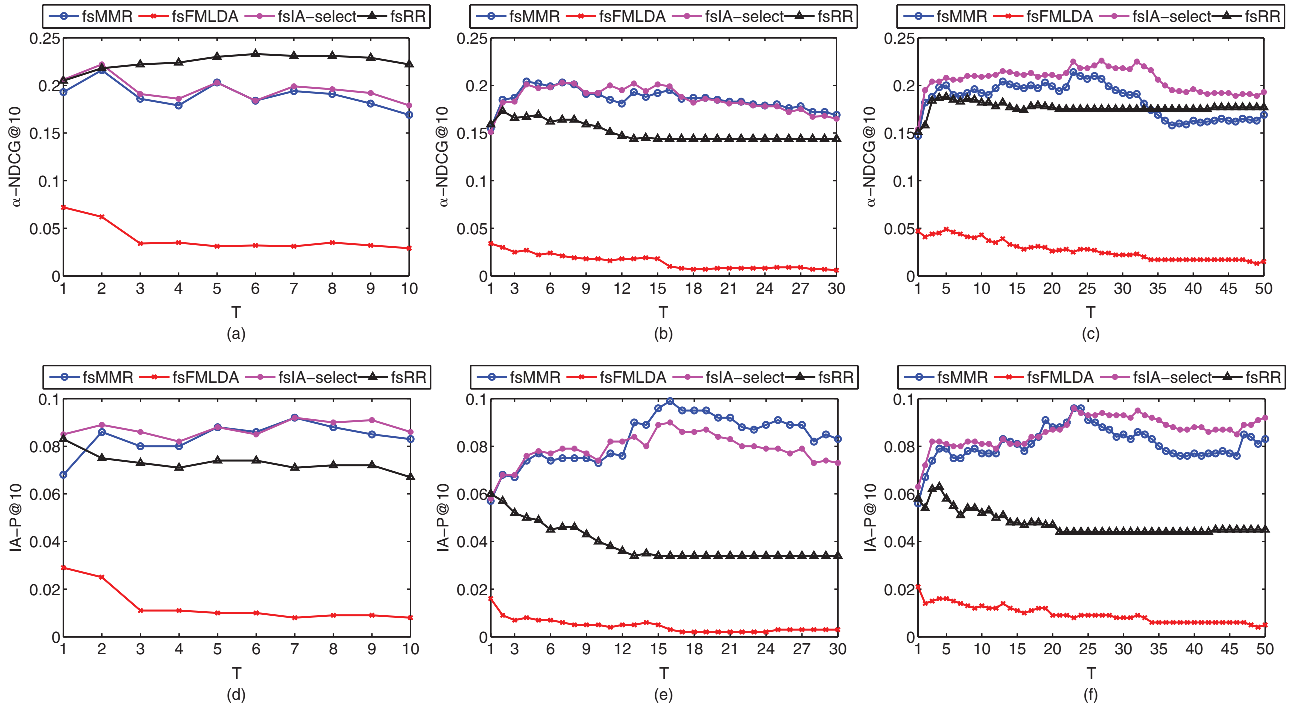


FIG. 3. Diversification with cluster ranking using query likelihood as *cRanker*(·) over different numbers of selected top-ranked clusters (T). The evaluation metrics are α-NDCG@10 (top row) and IA-P@10 (bottom row). The total number of clusters K is set to 10 (3(a) and 3(d)), 30 (3(b) and 3(e)) and 50 (3(c) and 3(f)). Note that the plots have different scales on the Y-axis for different evaluation metrics.

Effectiveness of Diversification With Cluster Ranking

How does diversification with cluster ranking compare with diversification over the complete ranked list of documents? Figure 3 shows the trends of the performance of each diversification method with cluster ranking (cMMR, cFM-LDA, cIA-select, and cRR) across values of T, the number of top-ranked clusters whose documents are used for diversification. For each method, when $T = K$, diversification with cluster ranking is equivalent to diversifying the complete list of initially retrieved documents. Here, we only show the results measured using α-NDCG@10 and IA-P@10 for $K = 10$, 30, and 50; a similar trend can be observed for α-NDCG@X and IA-P@X, for $X = 5,20$.

For all methods, the plots in Figure 3 show that diversification does not benefit from, or is even hurt by, selecting all clusters, that is, by diversifying the complete ranked list of documents. In addition, for each method there is an optimal value of T that maximizes the performance of the method, which is smaller than the total number of clusters, that is, for which the optimal value of T satisfies $T < K$. If we could accurately find this optimal T, the diversification performance is bound to be more effective than diversification over the complete ranked list of documents. We conclude from this observation that, for a given cluster ranker, the proposed framework has the potential to improve the diversification effectiveness if a proper T is chosen. In the following sections, we will further examine whether the difference between diversification with entire ranked list and diversification with selected T clusters is significant, where the selected T can be determined through cross-validation as well as set by oracle.

We therefore investigate the effectiveness of diversification with cluster ranking based on the query-likelihood cluster ranker combined with the predicted T next.

Diversification With the Query Likelihood-Based Cluster Ranker and Predicted T

Now let us look at the performance of diversification using query likelihood for ranking clusters and using cross-validation to predict the number T of top-ranked clusters to be considered for diversification. Tables 1–4 compare diversification with cluster ranking against diversifying the complete list of retrieved documents. As before, *cX* indicates the runs with cluster ranking and selection, where X is the name of a diversification method; in each table, K is the total number of clusters. We also list the average predicted value of T. On top of that, we include the performance achieved by each method when T is optimal, which is indicated by T^*, e.g., the peak points in Figure 3. Note that T^* is different from the oracle T: in the case of oracle T, the value of T is optimized for each query, while T^* is optimized for the average performance over all queries.

We see that for different diversification methods, diversification with cluster ranking outperforms the original algorithms in nearly all cases, even though query likelihood is not a perfect ranker for ranking clusters and T has not been

TABLE 1. Results of MMR vs. cMMR.

		α-NDCG@5		α-NDCG@10		IA-P@5		IA-P@10	
K	Method	Score	avg. T	Score	avg. T	Score	avg. T	Score	avg.T
–	MMR	0.122	–	0.169	–	0.066	–	0.083	–
10	cMMR	**0.191**$^{\triangle}$	1.98	**0.216**	2.00	0.070	2.44	0.069	6.82
	cMMR$^{T^*}$	**0.191**$^{\triangle}$	2	**0.216**	2	**0.090**	2	**0.092**	7
30	cMMR	0.157	4.42	0.171	4.76	0.077	13.54	0.090	15.94
	cMMR$^{T^*}$	**0.179**$^{\triangle}$	4	**0.204**	4	**0.085**	16	**0.099**	16
50	cMMR	0.178$^{\triangle}$	21.80	**0.214**$^{\triangle}$	23.00	0.090	23.00	**0.096**	23.00
	cMMR$^{T^*}$	**0.179**$^{\triangle}$	23	**0.214**$^{\triangle}$	23	**0.092**	23	**0.096**	23

Note. For each K and each evaluation metric, the performance of cMMR is compared with the corresponding performance of MMR. Boldface indicates the best score achieved for a given K. For cMMR$^{T^*}$, the avg. T is the value of T^*.

TABLE 2. Results of FM-LDA vs. cFM-LDA.

		α-NDCG@5		α-NDCG@10		IA-P@5		IA-P@10	
K	Method	Score	avg. T	Score	avg. T	Score	avg. T	Score	avg.T
10	FM-LDA	0.027	–	0.029	–	0.011	–	0.008	–
	cFM-LDA	**0.058**	1.00	**0.072**$^{\triangle}$	1.00	**0.031**$^{\triangle}$	1.00	**0.029**$^{\triangle}$	1.00
	cFM-LDA$^{T^*}$	**0.058**	1	**0.072**$^{\triangle}$	1	**0.031**$^{\triangle}$	1	**0.029**$^{\triangle}$	1
30	FM-LDA	0.000	–	0.006	–	0.000	–	0.003	–
	cFM-LDA	0.020$^{\triangle}$	2.06	0.027$^{\triangle}$	1.02	0.009$^{\triangle}$	1.00	**0.016**$^{\triangle}$	1.96
	cFM-LDA$^{T^*}$	**0.022**$^{\triangle}$	2	**0.034**$^{\triangle}$	1	**0.010**$^{\triangle}$	2	**0.016**$^{\triangle}$	1
50	FM-LDA	0.008	–	0.015	–	0.004	–	0.005	–
	cFM-LDA	0.020	1.32	0.026	4.60	**0.021**$^{\triangle}$	1.00	**0.021**$^{\blacktriangle}$	1.00
	cFM-LDA$^{T^*}$	**0.038**$^{\blacktriangle}$	1	**0.049**$^{\triangle}$	5	**0.021**$^{\blacktriangle}$	1	**0.021**$^{\blacktriangle}$	1

Note. For each K, the results of cFM-LDA are compared to the corresponding results of FM-LDA. Boldface indicates the best score achieved for a given K. For cFM-LDA$^{T^*}$, the avg. T is the value of T^*.

TABLE 3. Results of IA-select vs. cIA-select.

		α-NDCG@5		α-NDCG@10		IA-P@5		IA-P@10	
K	Method	Score	avg. T	Score	avg. T	Score	avg. T	Score	avg.T
10	IA-select	0.125	–	0.179	–	0.069	–	0.086	–
	cIA-select	**0.199**$^{\triangle}$	2.00	0.221	2.00	0.053	3.30	0.056	6.58
	cIA-select$^{T^*}$	**0.199**$^{\triangle}$	2	**0.222**	2	**0.096**	7	**0.092**	7
30	IA-select	0.116	–	0.165	–	0.063	–	0.073	–
	cIA-select	0.145	7.00	0.158	7.64	0.079	14.36	0.077	16.00
	cIA-select$^{T^*}$	**0.185**$^{\triangle}$	7	**0.203**	7	**0.094**$^{\triangle}$	16	**0.090**$^{\triangle}$	16
50	IA-select	0.146	–	0.193	–	0.078	–	0.092	–
	cIA-select	0.181$^{\triangle}$	15.06	0.208	27.14	0.100	31.36	0.092	23.54
	cIA-select$^{T^*}$	**0.199**$^{\triangle}$	9	**0.226**$^{\triangle}$	27	**0.105**$^{\triangle}$	32	**0.096**	23

Note. For each K, the results of cIA-select are compared with the corresponding reults of IA-select. Boldface indicates the best score achieved for a given K. For cIA-select$^{T^*}$, the avg. T is the value of T^*.

fully optimized. If we take the optimal T with respect to the average performance over all queries, i.e., T^*, we see further improvements, and more improvements are statistically significant compared with that of the predicted T. In some cases, the average predicted T is very close to the T^* and results in similar performance. However, small difference between the average predicted T and T^* does not necessarily lead to small difference between diversification results. This may because the difference between the average predicted T and the T^* does not reflect the per-query difference, which can in fact lead to very different results.

Below, we take a close look at the performance of individual diversification methods, focusing on the results obtained using automatically determined T. Results obtained

TABLE 4. Results of RR vs. cRR.

		α-NDCG@5		α-NDCG@10		IA-P@5		IA-P@10	
K	Method	Score	avg. T	Score	avg. T	Score	avg. T	Score	avg.T
10	RR	0.198	–	0.222	–	0.079	–	0.067	–
	cRR	0.199	2.68	**0.233**$^{\triangle}$	6.00	0.085	2.00	**0.083**	1.00
	cRR$^{T^*}$	**0.204**	2	**0.233**$^{\triangle}$	6	**0.091**	2	**0.083**	1
30	RR	0.137	–	0.144	–	0.049	–	0.034	–
	cRR	0.151	2.94	0.168	2.06	0.065	2.90	**0.060**$^{\triangle}$	1.00
	cRR$^{T^*}$	**0.152**	2	**0.173**$^{\triangle}$	2	**0.068**$^{\triangle}$	2	**0.060**$^{\triangle}$	1
50	RR	0.157	–	0.177	–	0.057	–	0.045	–
	cRR	0.160	5.00	0.172	4.86	0.067	3.20	0.056	3.92
	cRR$^{T^*}$	**0.176**$^{\triangle}$	5	**0.188**	5	**0.072**	3	**0.063**$^{\triangle}$	4

Note. For each K, the results of cRR are compared with the corresponding results of RR. Boldface indicates the best score achieved for a given K. For cRR$^{T^*}$, the avg. T is the value of T^*.

by T^* are listed for completeness, but not discussed further.

For MMR (Table 1), we see that in all cases except when $K = 10$ and for IA-P@10, the performance of diversification with cluster ranking improves over the original diversification algorithm, although the improvements are not always statistically significant.

For FM-LDA (Table 2), we see that in all cases, diversification with cluster ranking improves over diversification without cluster ranking; in most cases the improvement is statistically significant. In addition, we notice that the average number of selected top-ranked clusters in each case is small compared with other methods (that is, cIA-select, cMMR, and cRR). In other words, when more clusters are included for diversification, the performance of FM-LDA drops quickly. This phenomenon suggests that FM-LDA may be very sensitive to non-relevant documents: including more clusters increases the chance of including more non-relevant documents for diversification and the performance of FM-LDA decreases in this situation.

For IA-select (Table 3), we see that in most cases, the performance is improved by applying diversification with cluster ranking. Exception includes the following cases: $K = 10$ using IA-P@5 and IA-P@10, $K = 30$ using α-NDCG@10 and $K = 50$ using IA-P@10 where the performance stays the same.

For RR (Table 4), in all cases except when $K = 50$ using α-NDCG@10, diversification with cluster ranking outperforms the original method. Note that ranking clusters is inherent for RR and the only difference between RR and cRR is that cRR applies RR on top T selected clusters. The improvement of cRR over RR shows that eliminating from the diversification process clusters that are likely to be non-relevant to the query can effectively improve the result diversification performance.

Finally, we have a look at cases where diversification with cluster ranking does not outperform their original counterparts. Let us use cIA-select as an example. If we look at the corresponding plots in Figures 3(b), (d) and (f) for the cases where cIA-select loses against IA-select, we see that the performance curves of cIA-select across different cut-off values T fluctuate frequently and on each curve, several local maximums exist and the differences between those local maximums are small. On the one hand this may create difficulties for the cross-validation approach to find a global optimal T; on the other hand, this indicates that the ranking of clusters needs to be improved. Similar observations can be made for cMMR and cRR.

Additional Remarks

Although not directly related to our experimental objectives, in Table 5 we show the performance of the initial retrieval result generated by the MRF model, as measured using diversification metrics. We compare its performance to that of applying diversification methods, and of the results of diversification with cluster ranking. For the results of diversification with cluster ranking, we only show the runs with best performance among different K values, in terms of α-NDCG@10 and IA-P@10. Note that the value of K that results in the best performance may differ for different diversification methods, which suggests that for optimizing performance and a careful model selection, K should be tuned separately for each diversification method and metric.

Diversification with cluster ranking outperforms diversification over the complete ranked list of documents, but does not always outperform the baseline, that is, the initial ranked list returned by MRF. The performance of diversification with cluster ranking is closely related to the performance of the underlying diversification methods: diversification methods that perform better, e.g., IA-select and RR, result in better performance with cluster ranking.[4] The performance of FM-LDA is low in general, which may be due to the fact that it retrieves too few relevant documents after diversification, as was also found by Carterette and Chandar (2009).

[4] The performance of IA-select and RR and their cluster ranking versions is between the median and the best of systems taking part in the diversity task at TREC 2009 Web Track in terms of αNDCG@10 (best: 0.526; median: 0.175) and IA-P@10 (best: 0.244; median: 0.073).

TABLE 5. The performance of the initially retrieved ranked list of documents (MRF) in terms of diversity and the optimal performance of diversification methods and the corresponding cluster ranking versions.

Methods	α-NDCG@5	α-NDCG@10	K	IA-P@5	IA-P@10	K
MRF (baseline)	0.118	0.170	–	0.069	0.088	–
MMR	**0.122**	0.169	–	0.066	0.083	–
cMMR	**0.191**△	**0.216**	10	**0.090**	**0.096**	50
FM-LDA	0.027▼	0.029▼	10	0.011▼	0.008▼	10
cFM-LDA	0.058▼	0.072▼	10	0.031▼	0.029▼	10
IA-select	**0.146**△	**0.193**△	50	**0.078**	**0.092**	50
cIA-select	**0.199**▲	**0.221**△	10	**0.100**△	**0.092**	50
RR	**0.198**▲	**0.222**△	10	**0.079**▲	0.067▽	10
cRR	**0.200**▲	**0.233**▲	10	**0.085**▲	0.083	10

Note. Clusters are ranked with query likelihood. Bold face indicates improved performance over the baseline, i.e., MRF. Significance is tested against the MRF baseline.

In Table 5 we notice that RR and its cluster-based version cRR, while simple, are very effective comparing with other diversification methods. The effectiveness of RR and cRR may be due to the following reasons. By applying RR, we first need to rank the clusters, which potentially improves the early precision. On top of that, we select documents from different clusters in an RR fashion, which promotes diversity. cRR, on top of that, cut the clusters at top T, further prevents potentially non-relevant clusters being included for diversification.

Answer to the Main Research Question

We turn to our main research question Q-1, for which we have obtained the following answers. First, with an imperfect cluster ranker, diversification using documents from a carefully selected number of top-ranked clusters can be more effective than diversification using all documents in the initial retrieved list. Second, in general, the query likelihood-based cluster ranker and the predicted T are effective for improving the performance of the diversification methods discussed in this article. In addition, as discussed in the section Additional remarks, the performance of diversification with cluster ranking is closely related to the performance of the underlying diversification method (that is, without cluster ranking).

Sensitivity Analysis

In this section, we offer a first of two rounds of analysis into the effectiveness of diversification with cluster ranking. In general, the analysis in this section provides insights into the sensitivity of our proposed framework to various parameter settings. Specifically, we aim to answer research question Q-2 and Q-3.

Impact of the Cluster Ranker and T

The aim of this section is to answer research question Q-2: What is the impact of the two main components, namely, the cluster ranker and the selection of the number of top-ranked clusters, on the overall performance of diversification with cluster ranking?

To answer this question, we use a set of oracle experiments based on the oracle cluster ranker; we run the experiments with oracle parameter settings as described in the section Research questions and experiments.

Figure 4 shows the trends of the performance of each diversification method across values of T with the oracle cluster ranker. If we compare Figure 4 with Figure 3, we see that in Figure 3 the retrieval performance fluctuates a lot as T increases, that is, with many local maximums, whereas in Figure 4, the performance curves are relatively smooth: they remain the same or decrease once an initial maximum has been reached. This implies that, with a near perfect ranking of clusters, we can find the global optimal T by simply adding documents belonging to a cluster ranked next, until the performance starts to decrease. On top of that, we clearly see that the optimal results are achieved by selecting a small number of top-ranked clusters. In addition, we notice that the oracle cluster ranker has a different impact on different diversification methods. For example, in Figure 3, cIA-select has a similar performance as cMMR in most cases, whereas in Figure 4, cIA-select consistantly outperforms other methods.

Now let us take a close look at the results of the oracle experiments, which use oracle information for ranking clusters or determining T, or both. Tables 6–9 show the oracle performance of diversification in three settings. First, T is selected using an oracle and clusters are ranked with query likelihood. Second, T is automatically determined and the clusters are ranked with the oracle cluster ranker. And, third, both T and the cluster ranker use oracle information. For comparison, we also include results of the following experiments: diversification over the complete ranked list of documents, and diversification with cluster ranking but without oracle information using the predicted T and query likelihood cluster ranker.

Note that for IA-select and RR, since the importance of clusters is taken into account in the original algorithms when ranking with the oracle cluster ranker, their baselines change as well. For the baselines of IA-select and RR with oracle ranker, we use the oracle information to rank the clusters for

DOI: 10.1002/asi

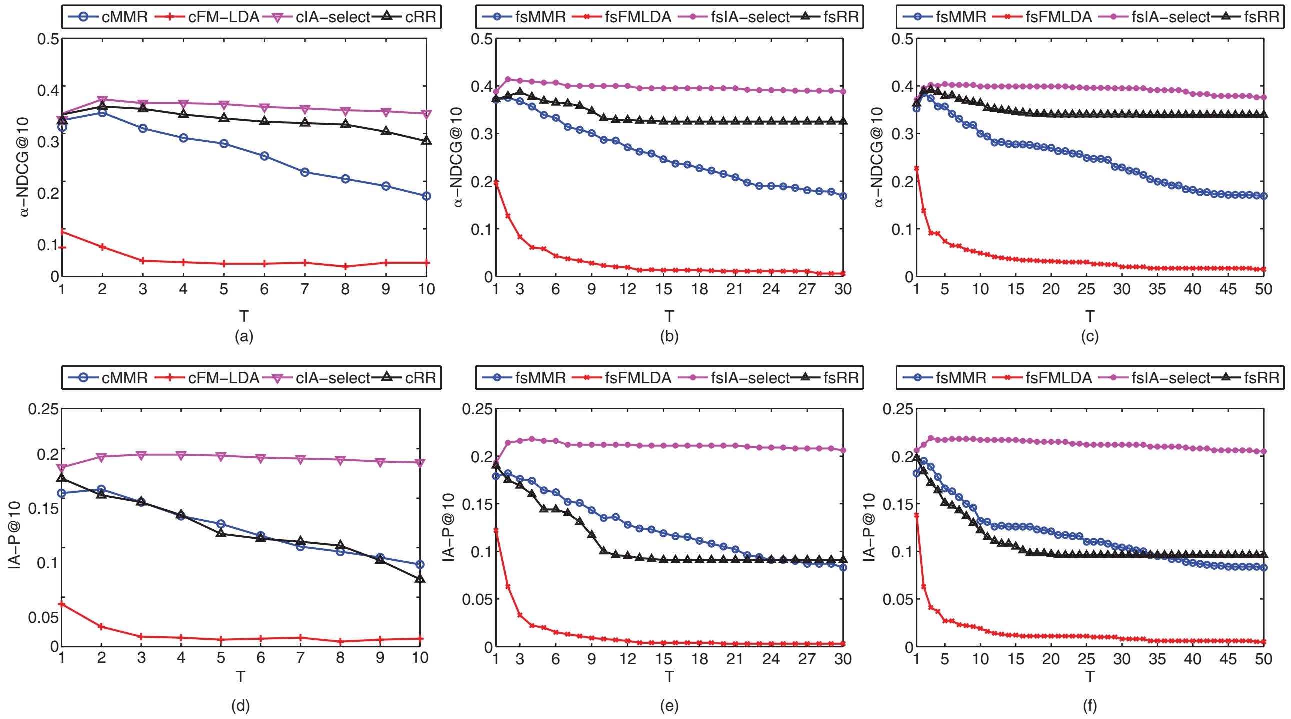


FIG. 4. Diversification with cluster ranking using oracle information as $cRanker(\cdot)$ over different numbers of selected top-ranked clusters (T). The evaluation metrics are α-NDCG@10 (top row) and IA-P@10 (bottom row). The number of clusters K is set to 10 (3(a) and 3(d)), 30 (3(b) and 3(e)) and 50 (3(c) and 3(f)). Note that the plots have different scales at the Y-axis for different evaluation metrics.

TABLE 6. Results of MMR, cMMR, and the oracle versions of cMMR.

Method	T_o	R_o	α-NDCG@5			α-NDCG@10			IA-P@5			IA-P@10		
MMR	–	–	0.122			0.169			0.066			0.083		
			$K=10$	$K=30$	$K=50$	$K=10$	$K=30$	$K=50$	$K=10$	$K=30$	$K=50$	$K=10$	$K=30$	$K=50$
cMMR	–	–	0.191△	0.157	0.178△	0.216	0.171	0.214△	0.070	0.077	0.090	0.069	0.090	0.096
	+	–	0.281▲	0.284▲	0.264▲	0.313▲	0.307▲	0.311▲	0.140▲	0.147▲	0.144▲	0.138	0.138▲	0.146▲
	–	+	0.312▲	0.331▲	0.357▲	0.344▲	0.344▲	0.385▲	0.146▲	0.212▲	0.204▲	0.142▲	0.178▲	0.195▲
	+	+	0.369▲	0.406▲	0.417▲	0.401▲	0.432▲	0.440▲	0.204▲	0.234▲	0.233▲	0.195▲	0.217▲	0.217▲

Note. For each K and each evaluation metric, the performance of the oracle runs is compared with the corresponding performance of MMR. Columns T_o and R_o list whether the oracle T and R are used.

the two algorithms, but apply diversification on the whole ranked list.

Two observations can be made. First, for each diversification method, both oracle T and the oracle cluster ranker significantly improve the effectiveness of result diversification over their corresponding baselines. Moreover, in the case of automatically determined T, while not all cases are improved over the baselines when using the query likelihood-based cluster ranker, the proposed approach outperforms the baselines in all cases when using the oracle cluster ranker, and many of the improvements are statistically significant. That is, the prediction of T is more effective when an oracle cluster ranker is used.

Second, using the oracle cluster ranker results in better performance in terms of the diversification metrics than using an oracle to determine T in all cases except in the case of FM-LDA with $K=10$ of α-NDCG@5 and IA-P@5; this suggests that the oracle cluster ranker has a larger impact on the diversification results than the oracle T. On top of that, combining the oracle cluster ranker and the oracle T always results in improved performance.

In addition, for methods such as IA-select and RR, the oracle information of cluster distribution, as defined in the section Experimental results, helps in both cases, with and without cluster ranking and selection, as these methods take into account the importance of clusters, and the oracle information provides a good approximation of the importance of clusters. In the case of MMR and FM-LDA, where importance of clusters is not considered, the oracle information of cluster distribution only helps when cluster ranking is applied.

DOI: 10.1002/asi

TABLE 7. Results of FM-LDA, cFM-LDA, and the oracle versions of cFM-LDA.

Method	T_o	R_o	α-NDCG@5			α-NDCG@10			IA-P@5			IA-P@10		
			$K=10$	$K=30$	$K=50$	$K=10$	$K=30$	$K=50$	$K=10$	$K=30$	$K=50$	$K=10$	$K=30$	$K=50$
FM-LDA	–	–	0.027	0.000	0.008	0.029	0.006	0.015	0.011	0.000	0.004	0.008	0.003	0.005
FM-LDA	–	–	0.058	0.020$^{\triangle}$	0.020	0.072$^{\triangle}$	0.027$^{\triangle}$	0.026	0.031$^{\triangle}$	0.009$^{\triangle}$	0.021$^{\blacktriangle}$	0.029$^{\triangle}$	0.016$^{\triangle}$	0.021$^{\blacktriangle}$
	+	–	0.081$^{\blacktriangle}$	0.036$^{\blacktriangle}$	0.069$^{\blacktriangle}$	0.092$^{\blacktriangle}$	0.044$^{\blacktriangle}$	0.077$^{\blacktriangle}$	0.041$^{\blacktriangle}$	0.018$^{\blacktriangle}$	0.032$^{\blacktriangle}$	0.035$^{\blacktriangle}$	0.025$^{\blacktriangle}$	0.034$^{\blacktriangle}$
	–	+	0.069$^{\triangle}$	0.152$^{\blacktriangle}$	0.192$^{\blacktriangle}$	0.094$^{\blacktriangle}$	0.197$^{\blacktriangle}$	0.227$^{\blacktriangle}$	0.036$^{\triangle}$	0.098$^{\blacktriangle}$	0.118$^{\blacktriangle}$	0.043$^{\blacktriangle}$	0.122$^{\blacktriangle}$	0.138$^{\blacktriangle}$
	+	+	0.096$^{\blacktriangle}$	0.164$^{\blacktriangle}$	0.214$^{\blacktriangle}$	0.119$^{\blacktriangle}$	0.206$^{\blacktriangle}$	0.246$^{\blacktriangle}$	0.047$^{\blacktriangle}$	0.103$^{\blacktriangle}$	0.125$^{\blacktriangle}$	0.051$^{\blacktriangle}$	0.123$^{\blacktriangle}$	0.140$^{\blacktriangle}$

Note. For each K, the results of the oracle runs are compared with the corresponding results of FM-LDA.

TABLE 8. Results of IA-select, cIA-select, and the oracle versions of cIA-select.

			α-NDCG@5			α-NDCG@10			IA-P@5			IA-P@10		
Method	T_o	R_o	$K=10$	$K=30$	$K=50$	$K=10$	$K=30$	$K=50$	$K=10$	$K=30$	$K=50$	$K=10$	$K=30$	$K=50$
IA-select	–	–	0.125	0.116	0.146	0.179	0.165	0.193	0.069	0.063	0.078	0.086	0.073	0.092
cIA-select	–	–	0.199$^{\triangle}$	0.145	0.181$^{\triangle}$	0.221	0.158	0.208	0.053	0.079	0.100	0.056	0.077	0.092
cIA-select	+	–	0.287$^{\blacktriangle}$	0.252$^{\blacktriangle}$	0.262$^{\blacktriangle}$	0.317$^{\blacktriangle}$	0.285$^{\blacktriangle}$	0.291$^{\blacktriangle}$	0.153$^{\blacktriangle}$	0.130$^{\blacktriangle}$	0.137$^{\blacktriangle}$	0.150$^{\blacktriangle}$	0.123$^{\blacktriangle}$	0.127$^{\blacktriangle}$
IA-select	–	+	0.316	0.362	0.361	0.342	0.388	0.376	0.186	0.212	0.214	0.186	0.206	0.205
cIA-select	–	+	0.347$^{\blacktriangle}$	0.389$^{\triangle}$	0.372	0.372$^{\blacktriangle}$	0.407	0.392	0.197$^{\triangle}$	0.216	0.223	0.193$^{\triangle}$	0.210	0.213
cIA-select	+	+	0.374$^{\blacktriangle}$	0.424$^{\blacktriangle}$	0.416$^{\blacktriangle}$	0.394$^{\blacktriangle}$	0.443$^{\blacktriangle}$	0.429$^{\blacktriangle}$	0.218$^{\blacktriangle}$	0.245$^{\triangle}$	0.246$^{\triangle}$	0.209$^{\blacktriangle}$	0.232$^{\blacktriangle}$	0.231$^{\blacktriangle}$

Note. For each K, the results of cIA-select and cIA-select with oracle T are compared with that of IA-select, and the cIA-select runs where the oracle cluster ranker is used are compared with the IA-select run with oracle cluster ranker.

TABLE 9. Results of RR, cRR, and the oracle versions of cRR.

			α-NDCG@5			α-NDCG@10			IA-P@5			IA-P@10		
Method	T_o	R_o	$K=10$	$K=30$	$K=50$	$K=10$	$K=30$	$K=50$	$K=10$	$K=30$	$K=50$	$K=10$	$K=30$	$K=50$
RR	–	–	0.198	0.137	0.157	0.222	0.144	0.177	0.079	0.049	0.057	0.067	0.034	0.045
cRR	–	–	0.199	0.151	0.160	0.233$^{\triangle}$	0.168	0.172	0.085	0.065	0.067	0.083	0.060$^{\triangle}$	0.056
cRR	+	–	0.225	0.197$^{\blacktriangle}$	0.202$^{\blacktriangle}$	0.274$^{\triangle}$	0.230$^{\blacktriangle}$	0.243$^{\blacktriangle}$	0.107$^{\triangle}$	0.095$^{\blacktriangle}$	0.096$^{\blacktriangle}$	0.125$^{\blacktriangle}$	0.085$^{\blacktriangle}$	0.096$^{\blacktriangle}$
RR	–	+	0.296	0.334	0.346	0.284	0.325	0.339	0.121	0.146	0.153	0.068	0.091	0.096
cRR	–	+	0.339$^{\blacktriangle}$	0.362$^{\blacktriangle}$	0.361	0.357$^{\blacktriangle}$	0.387$^{\blacktriangle}$	0.384	0.179$^{\blacktriangle}$	0.205$^{\blacktriangle}$	0.202$^{\blacktriangle}$	0.170$^{\blacktriangle}$	0.190$^{\blacktriangle}$	0.198$^{\blacktriangle}$
cRR	+	+	0.382$^{\blacktriangle}$	0.413$^{\blacktriangle}$	0.422$^{\blacktriangle}$	0.409$^{\blacktriangle}$	0.442$^{\blacktriangle}$	0.442$^{\blacktriangle}$	0.223$^{\blacktriangle}$	0.239$^{\blacktriangle}$	0.239$^{\blacktriangle}$	0.208$^{\blacktriangle}$	0.225$^{\blacktriangle}$	0.226$^{\blacktriangle}$

Note. For each K, the results of cRR and cRR with oracle T are compared with that of RR, and the cRR runs where the oracle cluster ranker is used are compared with the RR run with oracle cluster ranker.

In summary, as an answer to research question Q-2, we find that both the cluster ranker and the cut-off value T are important to the effectiveness of our proposed diversification with cluster ranking framework. The oracle information for either the cluster ranker or the cut-off value T, or both, improve the performance of the proposed framework. This indicates that the performance of each component has a large impact on the overall performance of our framework. The cluster ranker has a larger impact than the cut-off value T on the effectiveness of the proposed framework.

Length Effect

Now we turn to research question Q-3: Given that we use top-ranked documents retrieved in response to a query for clustering as well as for diversification, how sensitive is the performance of the proposed framework to the length of the list of documents being selected?

We conduct the analysis experiments as described in the section Research questions and experiments, where we vary the number of documents for clustering and for applying Algorithm 1 in three settings. In order to summarize the massive amount of experimental results generated by the three settings along with variations of other parameters, such as the number of clusters K, the diversification method used (Div($\cdot$)) and the number of top-ranked clusters selected (T), we use the following three types of scores: Min, Max, and Avg. Specifically, for a given experimental setting, a given K and a given Div($\cdot$), we apply Algorithm 1 with all possible values of $T \in \{1, \ldots, K\}$ with the oracle cluster ranker.

TABLE 10. Results of Setting 1: $C = 100,300,500$; $D = 1000$.

Method	Score	$K=10$			$K=30$			$K=50$		
		$C100$	$C300$	$C500$	$C100$	$C300$	$C500$	$C100$	$C300$	$C500$
cMMR	Min.	**0.169**	**0.169**	**0.169**	**0.169**	**0.169**	**0.169**	**0.169**	**0.169**	**0.169**
	Max.	**0.351**	0.340	0.344	0.379	**0.399**	0.375	**0.405**	0.403	0.385
	Avg.	0.258	**0.262**	0.259	**0.267**	0.256	0.256	**0.258**	0.257	0.248
cFM-LDA	Min.	0.025	**0.039**	0.021	**0.022**	0.011	0.006	0.011	0.008	**0.015**
	Max.	**0.115**	0.097	0.094	0.226	**0.237**	0.197	0.269	**0.271**	0.227
	Avg.	0.050	**0.053**	0.038	**0.056**$^{\triangle}$	0.036	0.031	**0.041**	0.031	0.038
cIA-select	Min.	0.334	0.334	**0.342**	0.389	**0.405**	0.388	0.386	**0.407**	0.371
	Max.	0.369	0.371	**0.372**	0.407	**0.425**	0.414	0.409	**0.431**	0.404
	Avg.	0.347	0.350	**0.355**	0.396	**0.416**	0.397	0.393	**0.414**	0.393
cRR	Min.	0.283	0.281	**0.284**	0.351	**0.357**	0.325	**0.372**$^{\triangle}$	0.356	0.339
	Max.	0.358	**0.367**	0.357	0.395	**0.409**	0.387	**0.423**$^{\triangle}$	0.416	0.392
	Avg.	0.324	**0.328**	0.328	0.361	**0.370**	0.339	**0.379**$^{\triangle}$	0.367	0.348

Note. In each block, scores from columns $C100$ and $C300$ are compared with their corresponding scores in column $C500$; statistically significant difference between the scores from $C100$ ($C300$) and that from $C500$ is annotated by $^{\triangle}$. The highest scores among different settings of C for a given K, a given diversification method, and a given type of score (Min., Max., or Avg.) are shown in boldface.

TABLE 11. Results of Setting 2: $C = 500$, $D = 100,300,1000$.

Method	Score	$K=10$			$K=30$			$K=50$		
		$D100$	$D300$	$D1{,}000$	$D100$	$D300$	$D1{,}000$	$D100$	$D300$	$D1{,}000$
cMMR	Min.	**0.171**	0.169	0.169	**0.171**	0.169	0.169	**0.171**	0.169	0.169
	Max.	**0.378**$^{\triangle}$	0.361	0.344	**0.422**$^{\triangle}$	0.416$^{\triangle}$	0.375	**0.405**	0.385	0.385
	Avg.	0.260	**0.262**	0.259	0.254	**0.259**	0.256	**0.252**	0.250	0.248
cFM-LDA	Min.	**0.132**$^{\triangle}$	0.089$^{\triangle}$	0.021	**0.095**$^{\triangle}$	0.027$^{\triangle}$	0.006	**0.099**$^{\triangle}$	0.042$^{\triangle}$	0.015
	Max.	**0.341**$^{\triangle}$	0.237$^{\triangle}$	0.094	**0.395**$^{\triangle}$	0.340$^{\triangle}$	0.197	**0.357**$^{\triangle}$	0.320$^{\triangle}$	0.227
	Avg.	**0.198**$^{\triangle}$	0.132$^{\triangle}$	0.038	**0.157**$^{\triangle}$	0.074$^{\triangle}$	0.031	**0.144**$^{\triangle}$	0.076$^{\triangle}$	0.038
cIA-select	Min.	0.327	**0.352**	0.342	0.359	**0.392**	0.388	0.366	**0.385**	0.371
	Max.	0.369	**0.374**	0.372	0.422	**0.432**	0.414	0.402	**0.404**	**0.404**
	Avg.	0.347	**0.362**	0.355	0.376	**0.407**	0.397	0.381	**0.396**	0.393
cRR	Min.	**0.315**$^{\triangle}$	0.287	0.284	0.341	**0.344**	0.325	**0.341**	0.331	0.339
	Max.	**0.376**	0.363	0.357	**0.425**$^{\triangle}$	0.418$^{\triangle}$	0.387	**0.421**	0.403	0.392
	Avg.	**0.349**	0.333	0.328	**0.368**$^{\triangle}$	0.362$^{\triangle}$	0.339	**0.369**	0.347	0.348

Note. In each block, scores from columns $D100$ and $D300$ are compared with their corresponding scores in column $D1{,}000$; statistically significant difference between scores from $D100$ ($D300$) and that from $D1{,}000$ is annotated by $^{\triangle}$. The highest scores among different settings of C for a given K, a given diversification method and a given type of score (Min., Max., or Avg.) are shown in boldface.

For simplicity, we only use α-NDCG@10 as the evaluation metric. Then for each T we evaluate the results as the average α-NDCG@10 scores over all 50 queries. If we write the evaluation result as $E(T)$, i.e., as a function of T, we have

$$\text{Min} = \arg\min_T E(T), \quad \text{Max} = \arg\max_T E(T) \quad \text{and}$$
$$\text{Avg} = \sum_T E(T)/K$$

In other words, we compare the results from different settings in their worst performance, best performance, and average performance under different values of T, in terms of α-NDCG@10 which is averaged over 50 queries. For each setting, as described in the section Research questions and experiments, we compare the results of different settings of C and D, i.e., number of documents used for training the LDA model and the number of documents used for applying Algorithm 1, respectively, to the result of our baseline setting, i.e., $C = 500$ and $D = 1000$. We use two-sided paired t-test for significance test, where the significance level is set to 0.05.[5] Tables 10–12 show the results.

In Table 10 we see that, in general, the differences between different settings of C are not significant. There are two exceptions: cRR with $K = 50$, $C = 100$, which significantly outperforms $C = 500$ in all three types of scores; and cFM-LDA with $K = 30$, $C = 100$, where the performance difference is significant in terms of Avg scores. However, these occasional significant differences between performance may due be to various reasons; no clear pattern emerges in the

[5] Here, two-sided t-test is used, since we are interested in whether the performance is different, where the difference can be either “greater” or “less.”

TABLE 12. Results of Setting 3: *top*100 denotes $C = 100$, $D = 100$, *top*300 denotes $C = 300$, $D = 300$, and *top*500 denotes $C = 500$, $D = 1{,}000$.

		$K = 10$			$K = 30$			$K = 50$		
Method	Score	*top*100	*top*300	*top*500	*top*100	*top*300	*top*500	*top*100	*top*300	*top*500
cMMR	Min.	**0.171**	0.169	0.169	**0.171**	0.169	0.169	**0.171**	0.169	0.169
	Max.	**0.381**$^{\triangle}$	0.363	0.344	**0.434**$^{\triangle}$	0.430$^{\triangle}$	0.375	0.411	**0.424**	0.385
	Avg.	**0.270**	0.266	0.259	0.259	**0.261**	0.256	0.241	**0.255**	0.248
cFM-LDA	Min.	**0.101**$^{\triangle}$	0.069$^{\triangle}$	0.021	**0.055**$^{\triangle}$	0.040$^{\triangle}$	0.006	**0.050**$^{\triangle}$	0.037	0.015
	Max.	**0.327**$^{\triangle}$	0.225$^{\triangle}$	0.094	**0.398**$^{\triangle}$	0.363$^{\triangle}$	0.197	**0.355**$^{\triangle}$	0.354$^{\triangle}$	0.227
	Avg.	**0.171**$^{\triangle}$	0.107$^{\triangle}$	0.038	**0.123**$^{\triangle}$	0.082$^{\triangle}$	0.031	**0.094**$^{\triangle}$	0.072$^{\triangle}$	0.038
cIA-select	Min.	0.342	**0.348**	0.342	0.382	**0.407**	0.388	0.381	**0.395**	0.371
	Max.	0.376	**0.386**	0.372	0.420	**0.440**	0.414	0.415	**0.427**	0.404
	Avg.	0.360	**0.362**	0.355	0.400	**0.428**	0.397	0.390	**0.409**	0.393
cRR	Min.	**0.303**	0.275	0.284	**0.370**	0.363	0.325	0.328	**0.346**	0.339
	Max.	**0.386**	0.368	0.357	**0.438**$^{\triangle}$	0.436$^{\triangle}$	0.387	0.408	**0.439**	0.392
	Avg.	**0.349**	0.328	0.328	**0.394**$^{\triangle}$	0.380$^{\triangle}$	0.339	0.356	**0.367**	0.348

Note. In each block, scores from columns *top*100 and *top*300 are compared with their corresponding scores in column *top*500; statistically significant difference between scores from *top*100 (*top*300) and that from *top*500 is annotated by $^{\triangle}$. The highest scores among different settings of C for a given K, a given diversification method, and a given type of score (Min., Max., or Avg.) are shown in boldface.

overall performance when using different numbers of document for training the LDA models under our diversification framework.

From Table 11 we make two observations. First, we see that in general, smaller Ds (i.e., $D = 100{,}300$) are preferred to $D = 1{,}000$, as in all cases, none of the $D = 1{,}000$ outperform their $D = 100{,}300$ counterparts in terms of absolute values of evaluation score. Second, for each diversification method, we see certain patterns in their performance with different settings of D. For cMMR, cFM-LDA, and cRR, in general, $D = 100$ is preferred, as it achieves best performance in 24 out of 27 cases. Particularly, in terms of Max scores, for all three diversification methods, $D = 100$ results in best performance. In addition, we see that for cFM-LDA, all the differences between the $D = 100$, 300 and $D = 1{,}000$ are statistically significant. On the other hand, cIA-select is an interesting exception among other diversification methods: it does not show significant difference between different settings of D in any of cases. However, cIA-select seems to slightly prefer $D = 300$, as it results in best scores for all cases.

In Table 12 we see a similar pattern as in Table 10 for cMMR and cFM-LDA. That is, small numbers of documents ($C = 100$, $D = 100$ and $C = 300$, $D = 300$) are preferred over a large number of documents ($C = 500$, $D = 1{,}000$). In addition, the observation that significant difference between different settings of C and D occur under similar condition as in Table 11 suggests that the results of Setting 3 are under the impact of D, the number of documents on which Algorithm 3 is applied. Besides, cIA-select still shows no significant difference between different settings of C and D, with a slight preference toward $C = 300$, $D = 300$.

In summary, we have the following conclusions for answering research question Q-3. We find that the number of documents used for clustering does not have a significant and systematic impact on the overall performance of our proposed diversification framework. On the other hand, the number of documents for applying Algorithm 1 shows a systematic impact on the overall performance of the proposed diversification framework. For all diversification methods, a smaller number of documents, e.g., 100, 300, are preferred over a large number, which is set to 1000 in our experiments. In addition, we find that for cIA-select, both parameters do not show significant impact on the final diversification results.

Impact of Clustering Structure

Now let us turn to research question Q-4: What conditions should clusters fulfill in order for diversification with cluster ranking to be effective?

Since our prime motivation for applying query-specific clustering and cluster ranking to result diversification is its effect on promoting relevance, we first check the type of properties that makes query-specific clustering effective in promoting precision. From previous work on query-specific clustering, we know that the main reason why query-specific clustering can improve early precision is that among the document clusters, there exist a few high-quality clusters such that most of the relevant documents are contained in these clusters (Hearst & Pedersen, 1996; Kurland & Domshlak, 2008). Working on TREC-3 data (Harman, 1995), Hearst and Pedersen (1996) show that if for each query one clusters the top-ranked documents into five clusters, then *"the top-ranked cluster always contains over 50% of the relevant documents retrieved, ... The third, fourth and fifth-ranked clusters usually contain 10% or fewer."* If documents from those high-quality clusters are placed at the top of a ranked list, it is very likely that many of the relevant documents are promoted to the top of the ranked list, hence improving early precision.

On the other hand, from the diversification perspective, we expect that documents contained in those top-ranked clusters, while relevant to the general topic of a given query, cover multiple facets or sub-topics of the general topic. Intuitively, if the documents contained in the top-ranked clusters exclusively

TABLE 13. Comparison of three linkage types of the agglomerative hierarchical clustering method.

		Largest cluster			Other clusters			Uniform
Linkage type	*K*	Avg.	Std.	Perc. (%)	Avg.	Std.	Perc. (%)	(%)
UPGMA	10	943.78	126.04	95.9	4.31	14.22	0.4	10
	30	913.66	125.17	92.6	2.37	7.41	0.3	3.3
	50	887.28	124.54	89.7	1.94	5.45	0.2	2.0
Single linkage	10	963.74	138.65	97.9	2.1	23.30	0.2	10
	30	943.94	137.98	95.7	1.33	12.73	0.1	3.3
	50	924.14	137.32	93.5	1.19	9.59	0.1	2.0
Complete linkage	10	356.96	118.50	36.4	69.52	43.80	7.0	10
	30	158.0	53.20	15.9	28.43	19.24	2.8	3.3
	50	99.58	35.17	10.0	18.02	11.86	1.8	2.0

Note. *Largest cluster* shows the average size (Avg.), standard deviation (Std.), and the average percentage (Perc.) of the documents assigned to the largest cluster, calculated over the 50 test queries. *Other clusters* shows the same statistics for the rest of the clusters. *Uniform* shows the percentage of documents that should be assigned to each of the cluster if we have a uniform cluster size distribution.

focus on a single narrow topic, diversification will not be effective due to the lack of diverse content.

In summary, we expect that the clusters generated by a query-specific clustering algorithm should satisfy the following conditions to make diversification with cluster ranking effective:

> *Condition 1.* Among all clusters, there exist a small number of clusters, which we call *high-quality clusters*, that contain most of the *relevant* documents;
>
> *Condition 2.* The union of *high-quality clusters* should contain documents associated with multiple facets of a query, or in other words, documents whose contents are sufficiently different.

In the following sub sections, we examine the impact of the above two conditions on the effectiveness of our diversification with cluster ranking framework.

Preliminaries

Measuring the two conditions. In order to examine how the two conditions mentioned above are reflected by different types of clustering structures, we need measures that are able to capture the characteristics of a given clustering structure with respect to these two conditions.

We translate Condition 1 into the Precision score, which on the one hand, measures the amount of relevant documents contained in a given set of documents, and on the other hand, limits the size of the set of documents. That is, we do not want to have a set of clusters containing most of the relevant documents merely due to the fact that most documents are assigned to them.

For Condition 2, we propose to use an adapted version of the Coherence Score (He et al., 2009), which reverses the score so as to reflect "diversity" instead of "coherence." It is defined as

$$rCoh(D) = 1 - \frac{\sum_{i \neq j \in D} \delta(d_i, d_j)}{\frac{1}{2}|D|(|D| - 1)} \tag{7}$$

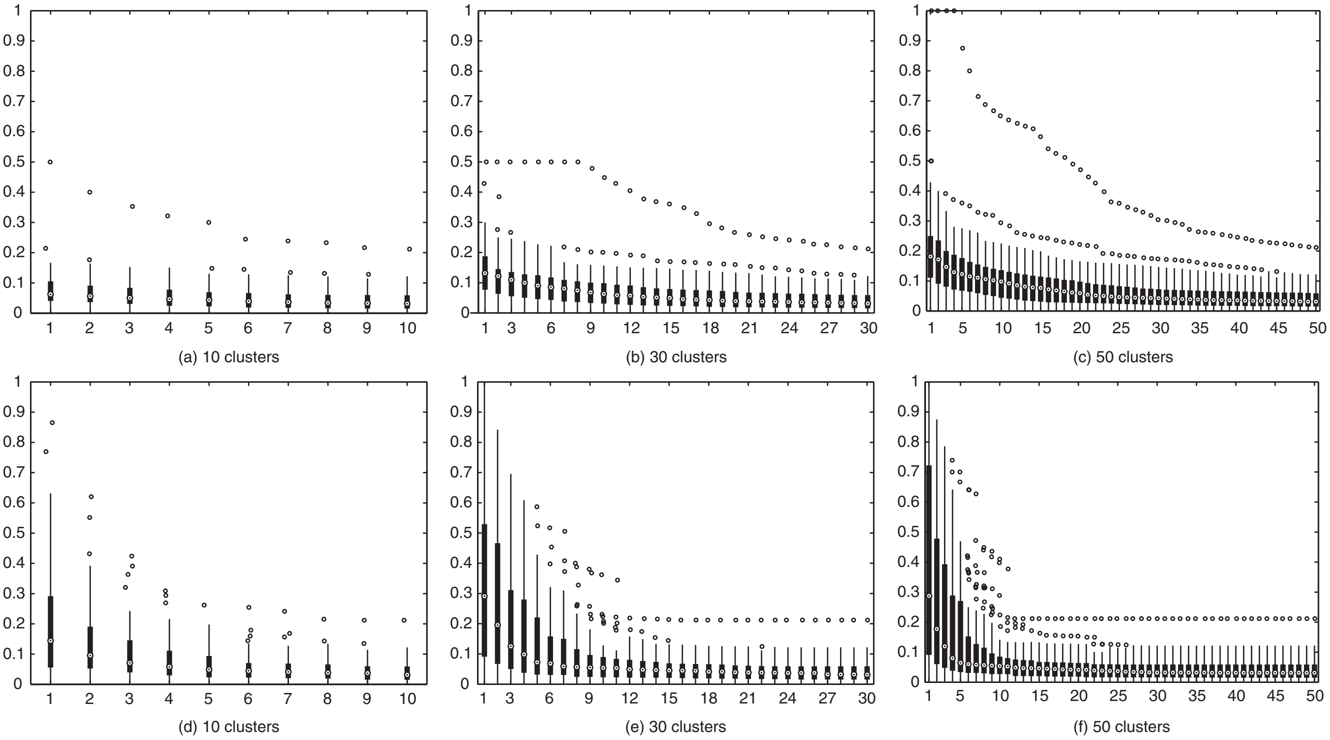
where for $i \neq j \in D$, $\delta(d_i, d_j) = 1$ if $sim(d_i, d_j) \geq \tau$, and 0 otherwise. We use cosine similarity for $sim(\cdot)$. Here, τ is a threshold indicating an "exceptionally high" similarity between two documents if they are randomly drawn from the collection; τ can be obtained by repeatedly sampling documents from the collection. In our experiments, the sampled τ turns out to be 0.024. The coherence score gives a higher value to a structured dataset than to a random set, and among structured datasets it gives higher values to sets with fewer clusters. In our case, the reversed Coherence score gives a high score to a set of documents if it has a rich sub-cluster structure; a low score if documents within the set are highly similar.

Hierarchical clustering. In order to generate a clustering structure different from those generated by the LDA models, we consider hierarchical clustering. Hierarchical clustering is different from LDA in nature: it is non-probabilistic and uses a vector-space representation for terms and documents. Potentially, these theoretical differences will lead to a different clustering structure.

We conduct hierarchical clustering as follows. For a query q, we create a set of clusters C, on D_q^n with agglomerative hierarchical clustering. For simplicity, we use cosine similarity to measure similarity between documents and use term TF.IDF for document representation. We consider different linkage types, including single-linkage, complete linkage, and group average (or unweighted pair group method with arithmetic mean (UPGMA)) (Sneath & Sokal, 1973).

For our experiments, we use the 50 test queries from the TREC 2009 Web track and the D_q^n are the documents returned by the MRF model per query, where $n = 1{,}000$. The number of clusters, K, is set to 10, 30, and 50.

In Table 13 we describe the properties of the clusters produced by agglomerative hierarchical clustering using three types of linkage. We see that on average, the largest cluster generated by single linkage and UPGMA constantly taking up over 90% of the documents, and each of the rest of the clusters has less than 1% of the documents, far from a uniform distribution of the cluster sizes. This type of results is



(a) 10 clusters

(b) 30 clusters

(c) 50 clusters

(d) 10 clusters

(e) 30 clusters

(f) 50 clusters

FIG. 5. Distribution of accumulated Precision scores among clusters. Figures 5(a)–(c) show the accumulated precision scores for clusters generated by hierarchical clustering, over 50 queries. Figures 5(d)–(f) show the same scores for clusters generated by LDA. In each box, the '⊙' at the central position is the median, the edges of the box are the 25th and 75th percentiles, the whiskers extend to the most extreme data points not considered outliers, and outliers are plotted individually as '○'.

undesirable for our task, as the dominant clusters are very likely to be considered as high-quality clusters due to the fact that they contain most of the relevant documents, but selecting a cluster with over 90% of the documents in the original ranked list probably does not make much difference from simply using the entire ranked list for diversification. On the other hand, complete linkage generates clusters whose sizes are relatively equal, compared with the other two linkage types. For example, when $K = 10$, if the size of clusters is uniformly distributed, each cluster should contain approximately 10% of the documents. As we see in Table 13 in the case of hierarchical clustering with complete linkage, the average percentage of documents assigned to each of the rest of the clusters is 7.0%, which is much closer to the uniform size distribution than that of single linkage and UPGMA (0.2 and 0.4%, respectively). Similar observations can be made for $K = 30$ and 50 as well. In addition, the largest clusters are not as dominant as those generated by single linkage and UPGMA. Based on the above observation, we decide to continue experiments with clusters generated with complete linkage and drop those generated by single linkage and UPGMA.

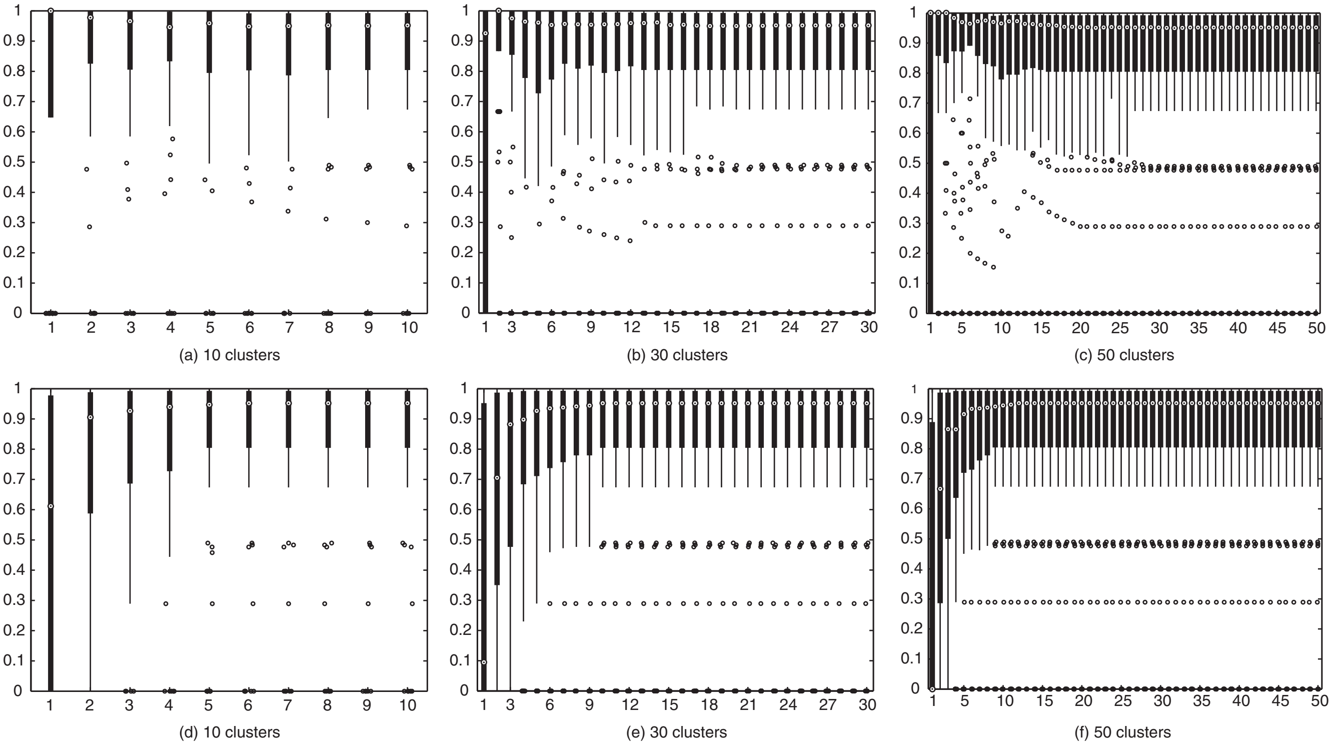
Clustering Structure

Now let us look at the clustering structure generated by the LDA models and hierarchical clustering with complete linkage, in terms of Precision and reversed Coherence scores.

Note that in Algorithm 1, given a ranked list of clusters, the diversification procedure is applied to the union of the documents contained in the top T clusters. Accordingly, the Precision and reversed Coherence scores are also calculated on the union of documents belonging to the top T clusters, which we refer to as accumulated Precision and accumulated reversed Coherence Scores, as the measures are taken on accumulated documents from a set of selected clusters.

To illustrate the cluster structure with respect to Condition 1, we first rank the clusters using the oracle cluster ranker as described in 4.2, which is equivalent to ranking with accumulated Precision scores. Then we plot the distribution of the accumulated Precision scores and reversed Coherence Scores for documents in the top T clusters, where $T = 1, \ldots, K$. Figure 5 shows the distribution of accumulated Precision scores and Figure 6 shows the distribution of accumulated reversed Coherence Scores among documents from the top T clusters. We see an interesting difference between the two clustering algorithms, namely, LDA and hierarchical clustering with complete linkage.

In Figure 5 we see that the early Precision scores of clusters generated by LDA are higher than those generated by the hierarchical clustering on average, but also have a larger variance. Note that the accumulated Precision score for the two clustering algorithms should converge to the same value at some point, as the same initial ranked list is used for both clustering procedures. For LDA, as the number of clusters being included increases, the accumulated Precision scores



(a) 10 clusters (b) 30 clusters (c) 50 clusters

(d) 10 clusters (e) 30 clusters (f) 50 clusters

FIG. 6. Distribution of accumulated reversed Coherence scores among clusters. Figures 6(a)–(c) show the accumulated reversed Coherence scores for clusters generated by hierarchical clustering, over 50 queries. Figures 6(d)–(f) show the same scores for clusters generated by LDA. In each box, the '⊙' at the central position is the median, the edges of the box are the 25th and 75th percentiles, the whiskers extend to the most extreme data points not considered outliers, and outliers are plotted individually as '○'.

decrease quickly, while for hierarchical clustering, the change is not very obvious, especially in the case of 10 clusters. The above observations suggest that clusters generated by LDA are more likely to satisfy Condition 1 than clusters generated by hierarchical clustering.

In Figure 6 we see that top-ranked clusters generated by hierarchical clustering with complete linkage have higher accumulated reversed Coherence Scores than those generated by LDA. In other words, the clusters generated by LDA are more likely to focus on single or few sub-topics, whereas clusters generated by hierarchical clustering are more likely to contain documents associated with multiple sub-topics or with diverse content. These observations suggest that clusters generated by hierarchical clustering are more likely to satisfy Condition 2 than those generated by LDA, that is, containing more diverse material.

Impact on the Performance of the Proposed Diversification Framework

Now that we have seen that the clusters generated by LDA and hierarchical clustering have a difference in clustering structure, let us examine whether the difference in clustering structure has an impact on the overall performance of our proposed diversification framework.

Figure 7 shows the results of diversification with cluster ranking with hierarchical clustering and LDA, in terms of α-NDCG@10 and IA-P@10. All clusters are ranked with oracle cluster ranker, so that we see how the clustering structure influences the performance under a perfect ranking. To incorporate hierarchical clustering into our proposed diversification framework, for cRR and cMMR, we simply apply Algorithm 1 with the clusters generated by hierarchical clustering. For cFM-LDA and cIA-select, we use hierarchical clustering to generate the clusters, and select top-ranked clusters for diversification. While applying Algorithm 1, we still use LDA for modeling the sub-topics of a query. That is, hierarchical clustering is only used for selecting documents to be diversified. In addition, in Table 14 we show the Pearson correlation between the end performance of our proposed framework and the Precision scores and reversed Coherence scores, which are calculated as in the previous section (section Clustering structure). All the correlations are significant except in the case for cFM-LDA, where the correlation between the reversed Coherence score and diversification result is not significant.

We notice that different diversification methods show different behaviors given different clustering algorithms. Let us refer to a diversification with cluster ranking procedure based on LDA as "the LDA version," and a procedure based on hierarchical clustering with complete linkage as "the HC version."

For cMMR and cFM-LDA, we see that initially, the LDA versions outperform their corresponding HC versions in all three settings of K, number of clusters, set to 10, 30, and 50. As T increases, the HC versions can outperform the LDA

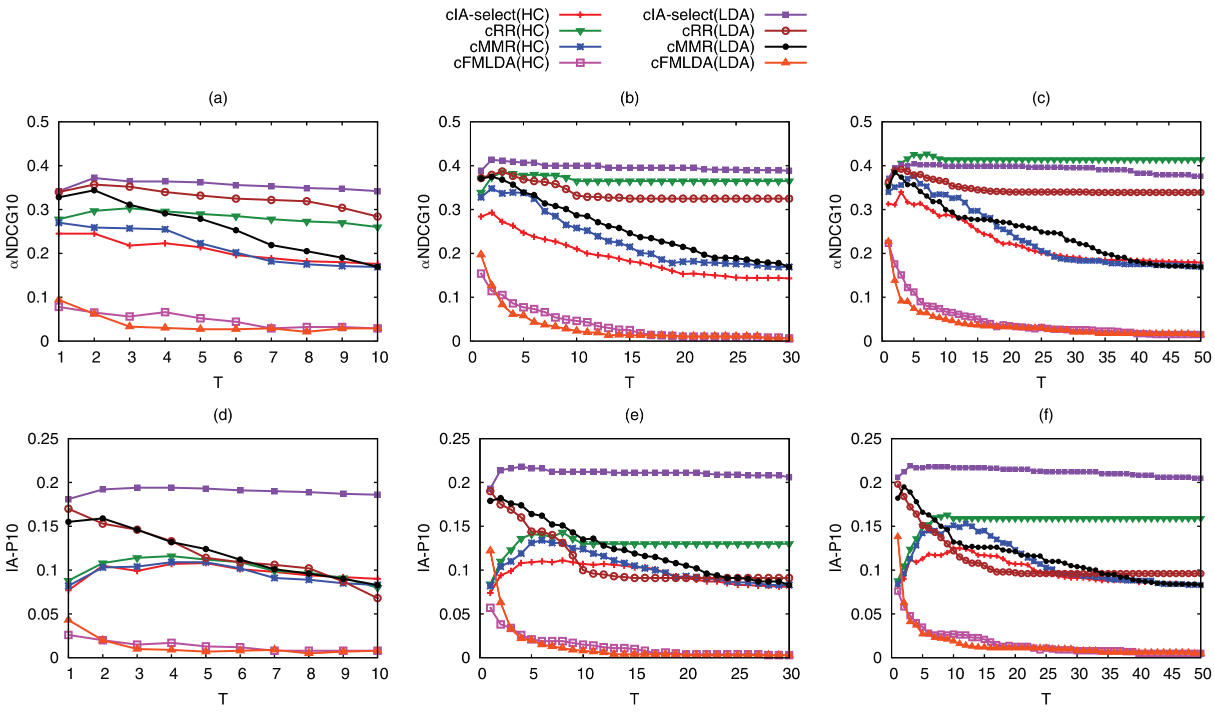


FIG. 7. Comparison of diversification results using the clusters generated by hierarchical clustering to clusters generated by LDA. In both cases, the clusters are ranked by oracle cluster ranker.

TABLE 14. Pearson correlation coefficients.

Measure	Comp. Type	cRR	cIA-select	cFM-LDA	cMMR
αNDCG@10	Precision	0.2968	0.2384	0.6117	0.3862
	rCoh	0.4062	0.2714	0.0549	0.3212
IA-P@10	Precision	0.4180	0.4135	0.6863	0.3612
	rCoh	0.2391	0.2528	0.0133^{A}	0.1650

Note. All correlation scores are statistically significant ($p < 0.01$) except the one with the A sign. Precision refers to the accumulated Precision scores for top T clusters, where $T = 1, \ldots, K$ and rCoh refers to the accumulated reversed Coherence scores.

versions, and vice versa; when $T = K$, since both versions are applied on the same initial ranked list, the performance ends up as the same.

In Table 14 we see that for cMMR and cFM-LDA, the correlation scores between the diversification results (measured by α-NDCG@10 and IA–P@10) and the Precision score are stronger than that between the diversification results and the reversed Coherence scores. This may be the reason why the initial performance of the LDA versions is better than that of the HC versions. As with small T, the early precision of top-ranked clusters has a larger impact on the performance of the proposed diversification framework. Recall that in the section Experimental results we noticed that the cFM-LDA selects relatively small T and we hypothesized that cFM-LDA is very sensitive to the non-relevant documents included when more clusters are included for diversification. The high correlation between the performance of cFM-LDA and the Precision scores, as we see from Table 14, further suggests that the gain of cFM-LDA by applying diversification with cluster ranking comes from the increased precision at the top-ranked clusters.

For cRR and cIA-select, we see that in Table 14, α-NDCG@10 is found to have a stronger correlation with the reversed Coherence scores than with the Precision scores, while IA-P@10 has a stronger correlation with the Precision scores than with the reversed Coherence scores. In Figure 7, correspondingly, we see that for IA-P@10, the LDA versions greatly outperform the HC versions at small Ts, which may be caused by the high early precision of the LDA versions. For α-NDCG@10, where the correlation between the diversification performance and the precision is not as strong, we see that for cRR, the LDA versions only slightly outperform the HC versions for small Ts and for cIA-select, the HC versions outperform the LDA versions. We also notice that for these two diversification methods, for larger Ts in the case of $K = 30$ and 50, the HC versions outperform the LDA versions

DOI: 10.1002/asi

in terms of both evaluation metrics, which suggests that the HC version may have achieved a better balance between Conditions 1 and 2 than the LDA version at larger Ts for these two methods.

Finally, it seems that the evaluation measures, α-NDCG and IA-P, have different preferences concerning relevance and diversity. In particular, IA-P has a bias toward precision as it consistently has a higher correlation with precision than with reversed coherence.

Conclusions

In summary, in this section, we posit that the clusters generated by a clustering algorithm should fulfill two conditions with respect to precision and diversity for our proposed diversification framework to be effective. Empirical results show that for most diversification methods, both conditions are significantly correlated with the overall performance of the framework. The impact of the two conditions on the overall performance, however, is dependent on the type of diversification method used, which suggests that when choosing a specific clustering algorithm, one should take into account the properties of individual diversification method.

Conclusions and Further Discussions

We investigated whether and how query-specific clustering can be used for improving the effectiveness of result diversification. More specifically, our aim was to take advantage of cluster-based retrieval methods for promoting relevance and restricting result diversification to a select set of high-quality clusters that contain large amounts of relevant documents so as to improve the effectiveness of diversification in terms of both relevance and diversity.

Our main findings can be summarized as follows. First, we proposed a diversification framework based on query-specific clustering with cluster ranking and selection, in which the diversification procedure is restricted to documents associated with clusters that potentially contain large amount of relevant documents. The framework was shown to improve the performance, as measured by α-NDCG and IA-P, of several types of diversification methods using a query likelihood-based cluster ranker and a cluster cut-off value T which is automatically determined via cross-validation.

On top of that, we analyzed the effectiveness of the proposed diversification framework with respect to four aspects: the cluster rankers, the cluster cut-off value T, the length effect of the initial retrieved ranked list, as well as the clustering structure generated by clustering algorithms. We showed that both the performance of the cluster ranker and the choice of the cluster cut-off value T are crucial to the overall performance of our diversification framework. In addition, the overall performance of the proposed framework is under the influence of the length of the initial ranked list of documents. Based on the lessons learnt from previous study in cluster-based retrieval as well as the characteristics of the result diversification task, we posited two conditions that the clusters generated by a clustering algorithm should fulfill in order for the diversification with cluster ranking to be effective. Our empirical results have shown that these conditions have a strong correlation with the overall performance, but the strength of the impact of each condition depends on the specific diversification method that is used. In addition, the question of "which clustering algorithm can effectively generate the desired clustering structure" remains, which we leave for the future work.

Our findings are interesting for developing new diversification methods as well as for cluster-based retrieval models for faceted queries. At the same time, various options for further analyses within our proposed diversification framework remain. In this article, we have only experimented with a simple strategy for ranking clusters and the oracle experiments show that there is sufficient room for the improvement with more sophisticated ranking approaches. Similarly, we have shown that there exists an optimal value of T, with which the effectiveness of diversification can be maximized. Clearly, more sophisticated learning methods should be explored for this purpose.

Acknowledgments

We are very grateful to our reviewers. Their detailed comments and suggestions helped us to improve this article.

This research was supported by the European Union's ICT Policy Support Programme as part of the Competitiveness and Innovation Framework Programme, CIP ICT-PSP under grant agreement nr 250430, by the DuOMAn project carried out within the STEVIN programme which is funded by the Dutch and Flemish Governments under project nr STE-09-12, by the Center for Creation, Content, and Technology (CCCT), and by the Netherlands Organisation for Scientific Research (NWO) under project nrs 612.066.512, 612.061.814, 612.061.815, 640.004.802.

References

Agrawal, R., Gollapudi, S., Halverson, A., & Ieong, S. (2009). Diversifying search results. In R.A. Baeza-Yates, P. Boldi, B.A. Ribeiro-Net, & B.B. Cambazoglu (Eds.), Proceedings of the Second ACM International Conference on Web Search and Data Mining (WSDM '09) (pp. 5–14). New York: ACM Press.

Allan, J., & Raghavan, H. (2002). Using part-of-speech patterns to reduce query ambiguity. In Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '02) (pp. 307–314). New York: ACM Press.

Blei, D.M., Ng, A.Y., & Jordan, M.I. (2003). Latent Dirichlet allocation. Journal of Machine Learning Research, 3, 993–1022.

Boyce, B.R. (1982). Beyond topicality: A two stage view of relevance and the retrieval process. Information Processing & Management, 18(3), 105–109.

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 (SIGIR '98) (pp. 335–336). New York: ACM Press.

Carpineto, C., Mizzaro, S., Romano, G., & Snidero, M. (2009). Mobile information retrieval with search results clustering: Prototypes and evaluations.

Journal of American Society for Information Science and Technology, 60(5), 877–895.

Carterette, B., & Chandar, P. (2009). Probabilistic models of ranking novel documents for faceted topic retrieval. In D.W.-L. Cheung, I.-Y. Song, W.W. Chu, X. Hu, and J.J. Lin (Eds.), Proceedings of the 18th ACM Conference on Information and Knowledge Management (CIKM '09) (pp. 1287–1296). New York: ACM Press.

Chen, H., & Karger, D.R. (2006). Less is more: Probabilistic models for retrieving fewer relevant documents. In E.N. Efthimiadis, S.T. Dumais, D. Hawking, & K. Järvelin (Eds.), Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '06) (pp. 429–436). New York: ACM Press.

Clarke, C., Craswell, N., & Soboroff, I. (2010, November). Overview of the TREC 2009 Web track. Paper presented the 18th Text REtrieval Conference (TREC 2009), Gaithersburg, MD.

Clarke, C.L., Kolla, M., Cormack, G.V., Vechtomova, O., Ashkan, A., Büttcher, S., & MacKinnon, I. (2008). Novelty and diversity in information retrieval evaluation. In S.-H. Myaeng, D.W. Oard, F. Sebastiani, T.-S. Chua, & M.-K. Leong (Eds.), Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '08) (pp. 659–666). New York: ACM Press.

Croft, W.B. (1980). A model of cluster searching based on classification. Information Systems, 5(3), 189–195.

Geman, S., & Geman, D. (1984). Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images. IEEE Transactions on Pattern Analysis and Machine Intelligence, 6, 721–741.

Goffman, W. (1964). A searching procedure for information retrieval. Information Storage and Retrieval, 2(2), 73–78.

Griffiths, T.L., & Steyvers, M. (2004). Finding scientific topics. Proceedings of the National Academy of Sciences of the United States of America, 101, 5228–5235.

Harman, D. (Ed.). (1995, November). Overview of the Third Text Retrieval Conference (TREC-3). Paper presented at the Third Text Retrieval Conference (TREC-3), Gaithersburg, MD.

He, J., Weerkamp, W., Larson, M., & de Rijke, M. (2009). An effective coherence measure to determine topical consistency in user-generated content. International Journal on Document Analysis and Recognition, 12, 185–203.

Hearst, M.A., & Pedersen, J.O. (1996). Reexamining the cluster hypothesis: scatter/gather on retrieval results. In H.-P. Frei, D. Harman, P. Schäuble, & R. Wilkinson (Eds.), Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '96) (pp. 76–84). New York: ACM Press.

Jardine, N., & van Rijsbergen, C. (1971). The use of hierarchic clustering in information retrieval. Information Storage and Retrieval, 7(5), 217–240.

Kurland, O. (2006). Inter-document similarities, language models, and ad hoc information retrieval (Doctoral dissertation.), Cornell University.

Kurland, O. (2008). The opposite of smoothing: A language model approach to ranking query-specific document clusters. In S.-H. Myaeng, D.W. Oard, F. Sebastiani, T.-S. Chua, & M.-K. Leong (Eds.), Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '08) (pp. 171–178). New York: ACM Press.

Kurland, O. (2009). Re-ranking search results using language models of query-specific clusters. Information Retrieval, 12(4), 437–460.

Kurland, O., & Domshlak, C. (2008). A rank-aggregation approach to searching for optimal query-specific clusters. In S.-H. Myaeng, D.W. Oard, F. Sebastiani, T.-S. Chua, & M.-K. Leong (Eds.), Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '08) (pp. 547–554). New York: ACM Press.

Liu, X., & Croft, W.B. (2004). Cluster-based retrieval using language models. In M. Sanderson, K. Järvelin, J. Allan, & P. Bruza (Eds.) Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '04) (pp. 186–193). New York: ACM Press.

Liu, X., & Croft, W.B. (2006a). Experiments on retrieval of optimal clusters. Technical Report, Center for Intelligent Information Retrieval (CIIR), University of Massachusetts.

Liu, X., & Croft, W.B. (2006b). Representing clusters for retrieval. In E.N. Efthimiadis, S.T. Dumais, D. Hawking, & K. Järvelin (Eds.), Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (pp. 671–672). New York: ACM Press.

Liu, X., & Croft, W.B. (2008). Evaluating text representations for retrieval of the best group of documents. In C. Macdonald, I. Ounis, V. Plachouras, I. Ruthven, & R.W. White (Eds.), Proceedings of the 30th European Conference on Advances in Information Retrieval (ERIC) (pp. 454–462). Berlin, Germany: Springer.

Metzler, D., & Croft, W.B. (2005). A Markov random field model for term dependencies. In R.A. Baeza-Yates, N. Ziviani, G. Marchionini, A. Mofat, & J. Tait (Eds.), Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '05) (pp. 472–479). New York: ACM Press.

Radlinski, F., Kleinberg, R., & Joachims, T. (2008). Learning diverse rankings with multi-armed bandits. In W.H. Cohen, A. McCallum, & S.T. Roweis (Eds.), Proceedings of the 25th International Conference on Machine Learning (pp. 784–791). Madison, WI: Omnipress.

Robertson, S.E. (1997). The probability ranking principle in IR. In Readings in Information Retrieval (pp. 281–286). Los Altos, CA: Morgan Kaufmann Publishers Inc.

Santos, R.L.T., Macdonald, C., & Ounis, I. (2010). Exploiting query reformulations for Web search result diversification. In Proceedings of the 19th International Conference on the World Wide Web (WWW '10) (pp. 881–890). New York: ACM Press.

Sneath, P.H.A., & Sokal, R.R. (Eds.) (1973). Numerical Taxonomy. San Francisco, CA: Freeman.

Tombros, A., Villa, R., & Van Rijsbergen, C.J. (2002). The effectiveness of query-specific hierarchic clustering in information retrieval. Information Processing & Management, 38(4), 559–582.

van Rijsbergen, C. (1979). Information retrieval. London: Butterworth.

Willett, P. (1988). Recent trends in hierarchic document clustering: A critical review. Information Processing & Management, 24(5), 577–597.

Yang, L., Ji, D.-H., Zhou, G., Yu, N., & Xiao, G. (2006). Document re-ranking using cluster validation and label propagation. In Proceedings of the 15th ACM International Conference on Information and Knowledge Management (CIKM '06) (pp. 690–697). New York: ACM Press.

Yue, Y., & Joachims, T. (2008). Predicting diverse subsets using structural SVMs. In Proceedings of the 25th International Conference on Machine Learning (ICML '08) (pp. 1224–1231). Madison, WI: Omni Press.

Zhai, C., & Lafferty, J. (2006). A risk minimization framework for information retrieval. Information Processing & Management, 42(1), 31–55.

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 (SIGIR '03) (pp. 10–17). New York, ACM Press.