Search Result Diversification for Enterprise Data

Wei Zheng, Hui Fang
Dept. of Electrical and Computer Engineering
University of Delaware
Newark, DE USA
{zwei,hfang}@udel.edu

Conglei Yao, Min Wang
HP Labs China
Haidian District
Beijing, China
{conglei.yao, min.wang6}@hp.com

ABSTRACT

Search result diversification aims to return a list of diversified relevant documents in order to satisfy different user information needs. Most of the efforts focused on Web Search, and few studies have considered another important search domain, i.e., enterprise search. Unlike Web search, enterprise search deals with both unstructured and structured data. In this paper, we propose to integrate the structured and unstructured data to discover meaningful query subtopics in search result diversification. Experimental results show that integrating structured and unstructured information allows us to discover high quality query, which are effective in diversifying the retrieval results.

Categories and Subject Descriptors

H.3.3 [**Information Search and Retrieval**]: Retrieval Models

General Terms

Algorithm

Keywords

enterprise search, diversification, query subtopics

1. INTRODUCTION

Traditional search engines rank documents mainly based on their relevance scores. As a result, top ranked results often contain relevant yet redundant information and may not be able to satisfy different user information needs. To overcome this limitation, it is necessary to diversify search results so that the returned documents would cover different pieces of relevant information, i.e. query subtopics. One of the key challenges is to identify meaningful query subtopics [16]. Almost all the existing studies on search result diversification focus on Web search, and the query subtopics are often extracted from the unstructured text and query logs [2, 6, 13]. However, less attention has been paid to solving the problem in enterprise search. Enterprise search is important because poor search results within enterprises could cause significant loss of productivities [8]. Unlike Web, enterprise data is more heterogeneous and integrated, and contain both unstructured and structured information.

In this paper, we study the problem of result diversification for enterprise search with the focus on extracting high-quality query subtopics from the integrated enterprise data. We first describe how to extract query subtopics from either the structured databases or the unstructured documents. We find that the subtopics extracted from *structured* data contain high-quality terms but have vocabulary gap with the retrieved documents while the subtopics extracted from *unstructured* data can better represent the document content but may contain a lot of noisy terms. Thus, we propose a method to integrate the subtopics extracted from *structured* data with the ones from *unstructured* data.

To evaluate the proposed query subtopic identification methods, we use the results as query subtopics and then diversify search results using a state of the art subtopic-based diversification method [13]. Experimental results over a real-world enterprise data collection demonstrate that the integrated information of enterprise data can provide high-quality query subtopics which lead to better diversification results.

2. RELATED WORK

Search result diversification has attracted a lot of attention recently [4]. Subtopic-based diversification methods aim to maximize the coverage of query subtopics in the retrieved documents [1, 13], and they are shown to perform better than redundancy-based methods on TREC collections [4].

One of the key challenges of the subtopic-based diversification methods is to extract the subtopics for a given query. Some studies tried to extract query subtopics from the retrieved documents using either document clustering [2] or topic modeling methods such as PLSA [12]. However, this type of methods cannot perform well because the generated subtopics often contain lots of noisy terms. To generate subtopics of better quality, a few studies utilized taxonomy such as Open Directory Project (ODP [1]) to extract subtopics [10, 14]. Although taxonomy-based methods provide high-quality subtopics, there is often a vocabulary gap between the taxonomy and the retrieved documents which could limit the effectiveness of the subtopics.

CIKM'11, October 24–28, 2011, Glasgow, Scotland, UK.
Copyright 2011 ACM 978-1-4503-0717-8/11/10 ...$10.00.

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

Another way of generating subtopics of better quality is to use query suggestions from Web search engines [13]. However, query suggestion methods cannot generate collection-dependent subtopics which may hinder the diversification performance. Dou et al. [6] extracted subtopics from different sources and diversify documents based on their coverage on different categories of subtopics. However, their method does not generate more meaningful subtopics using the unique features of different categories of subtopics.

Unlike previous studies on result diversification, this work focuses on solving the problem in the context of enterprise search, which has received much less attention despite its importance [8]. Our main contributions include: (1) we propose a novel query subtopic identification method for search result diversification, which can generate collection-dependent and high-quality subtopics; and (2) this is the first study that looked into how to better leverage the integrated information in enterprise data to improve the performance of search result diversification system.

3. SUBTOPIC EXTRACTION AND INTEGRATION

The enterprise data contain not only *unstructured* documents but also *structured* databases which often contain high-quality and domain-dependent information. In this section, we first describe how to extract subtopics from the documents and databases. We then propose the method to integrate these subtopics and generate the high-quality subtopics that are effective in result diversification.

3.1 Subtopic Extraction from Structured Data

We now study the problem of extracting subtopics from the structured data, i.e., the relational database. A unique characteristic of structured data such as the databases is the use of schema, which provides the meaning of the data as well as the domain knowledge. The schema information such as relations among different tables makes it possible to construct a multi-level concept hierarchy of the data in the databases. The goal is to select K subtopics that cover different relevant information of the query. Some existing studies [3, 5] diversify results using the information in the database schema, but they ignore the concept hierarchy indicated by the relations among schemas. Intuitively, a node contains more relevant information if not only the node itself is relevant to the query but also most of its descendants are relevant [9]. We therefore compute the relevance score of a node s_i based on the average similarity between the nodes in the sub-tree rooted at s_i and the query.

$$rel(s_i, q) = \frac{\sum_{s \in T_{s_i}} sim(s, q)}{|T_{s_i}|} \tag{1}$$

where s_i is the ith node in the database structure, i.e., a subtopic candidate, T_{s_i} is the sub-tree rooted at s_i and q is the query. $sim(s, q)$ is the semantic similarity between s and the query. We compute it based on the term co-occurrence information.

$$sim(s, q) = \frac{\sum_{t \in s} sim(t, q)}{|s|} \tag{2}$$

where t is a term in s and $sim(t, q)$ is the semantic similarity between the term and the query based on term co-occurrence information in the document set [7].

We iteratively select K nodes having the highest scores as the subtopics. In order to avoid different subtopics covering redundant information, we only select the node that is neither the ancestor nor the descendant of any previously selected nodes. These subtopics selected from the database often contain high-quality and domain-dependent information about the query. For example, the subtopics of the query "*printer*" cover the relevant information of printer types, accessories and customer service.

3.2 Subtopic Extraction from Unstructured Data

We can apply PLSA [11] method to mine the subtopic information from the unstructured data, i.e., original retrieved documents of the query. In order to avoid the overlapping information in different subtopics, the system assigns each term to the cluster where it has the highest score in the PLSA result.

$$s'(t) = \arg\max_{s' \in S'} score(t, s') \tag{3}$$

where S' is the set of subtopics, i.e., PLSA clusters, t is a term and $s'(t)$ is the subtopic that t is assigned to.

Each PLSA cluster is a subtopic. These subtopics are extracted from the clusters of documents and are therefore effective in distinguish documents covering different information of the query.

3.3 Subtopic Integration

The subtopics extracted from the databases contain high-quality information of the query and the subtopics extracted from retrieved documents contain the terms that can distinguish documents covering different subtopics of the query. Therefore, we propose to integrate the subtopics extracted from these two sources.

The task of subtopic integration in this section is that, given the K subtopics extracted from the databases and K subtopics extracted from documents, we combine them into K integrated subtopics where each subtopic contains M terms. Since the final goal of search result diversification is to diversify documents, we propose to use the subtopics extracted from database, containing high-quality information, to guide selection of integrated subtopic terms from subtopics extracted from documents. Specifically, in each query, we first propose to connect each subtopic of databases with a subtopic of documents based on their semantic similarity:

$$s_i = \arg\max_{s \in S} sim(s, s'_i) \tag{4}$$

where S is the set of subtopics extracted from the database, s'_i is the ith subtopic extracted from the documents, $sim(s, s'_i)$ is the semantic similarity between the subtopic of database and the subtopic of documents, and s_i is the subtopic of database assigned to s'_i. We assume the connection between these subtopics is 1 to 1, in order to simplify the problem. We leave other methods of connection for our future work.

For each subtopic extracted from the documents, we then select terms based on their semantic similarity to the connected subtopic of database [7]. The selected terms from each subtopic would form a new integrated subtopic that utilizes the information from both databases and documents. These integrated subtopics are generated by the guidance of subtopics extracted from database, so they often contain the information of higher quality. Moreover, their terms are extracted from the clusters of documents, so they could solve the problem of vocabulary mismatch and are more effective in diversifying documents.

Algorithm 1 Subtopic Integration

Input: a query q, a set of K document subtopics S', a set of K database subtopics S, semantic similarity between subtopics $sim(s, s')$, semantic similarity between the term and subtopic $sim(t, s)$, number of subtopic terms M
Output: integrated subtopic set S^* where $|S^*| = K$

```
 1: /*Assigning database subtopics to document subtopics*/
 2: for $s'_i \in S'$ do
 3:     $s_i = \arg\max_{s \in S} sim(s, s'_i)$
 4:     $S = S \setminus \{s_i\}$
 5: end for
 6: /*Selecting terms from each document subtopic*/
 7: $S^* = \emptyset$
 8: for $s'_i \in S'$ do
 9:     $s^* = \emptyset$        /*select M terms from each subtopic*/
10:     while $|s^*| < M$ and $|s^*| < |s'_i|$ do
11:         $t^* = \arg\max_{t \in s'_i \setminus s^*} sim(t, s_i)$
12:         $s^* = s^* \cup t^*$
13:     end while
14:     $S^* = S^* \cup \{s^*\}$
15: end for
16: return $S^*$
```

We now describe the details of our method in Algorithm 1. The first step is to connect a subtopic extracted from databases with a subtopic extracted from retrieved documents. For each subtopic extracted from documents, we find the most similar subtopic extracted from databases based on the term semantic similarity [7], i.e., lines 2-5 in Algorithm 1. The second step is to select M terms from each subtopic of documents as the terms in an intergraded subtopic. The term selection criteria is that we select the terms that have the highest semantic similarity with the connected database subtopic, as described in lines 7-15.

4. EXPERIMENTS

4.1 Enterprise Data Set

We evaluate the proposed subtopic extraction and integration methods over the collection of HP company. The data set includes both *unstructured data*, which consists of 477,800 web pages crawled from the web site of a computer company, and *structured data*, which includes 25 relational databases of the company.

To evaluate the performance of result diversification methods, we construct a query set and corresponding judgments following the procedure used in the diversity task of TREC Web track [4]. We construct the query set by selecting 50 popular queries from a query log during July 1st, 2010 and July 7th, 2010 of the same company. The average number of terms per query for the query set is 2.2. The 50 queries are selected from the query log in the following steps: (1) rank all the terms based on their frequencies in the query log; (2) manually select 100 meaningful terms from the top-ranked terms; (3) rank all the queries in the query log based on the number of selected terms contained in the queries; and (4) manually select 50 queries from the top-ranked queries to cover different product information in the database. The judgments are created by seven human judgers. The judgment file includes subtopics in each query and the documents relevant to each subtopic. The average number of subtopics per query is 4.12, which is close to the number from TREC diversity collections, i.e., 4.61.

We now discuss how to construct the concept hierarchy based on the structured information, i.e., relational databases. These tables are related to each other based on the foreign/primary keys. We only use the tables related to the products because most of the queries in the query log are related to the products. However, the proposed methods can be applied on any other enterprise databases and collections. We use the data in five tables containing the most important information of the products to build the concept hierarchy. These tables and their data form a concept hierarchy based on the relations among their foreign and primary keys. The levels in the hierarchy, from up to down, in the hierarchy are *product type* containing 8 nodes, *product marketing category* with 54 nodes, *product marketing subcategory* with 134 nodes, *product series* with 983 nodes and *product name* with 3,238 nodes. Each node in a level is the information of the product related to that level. One node in a lower level belongs to a ancestor node in the higher level and one node in the higher level has one or more descendants in the lower level.

4.2 Experiment Design

We first use the Dirichlet smoothing language model retrieval function [15] to retrieve documents for each query from the document collection, and then extract the subtopics using different methods. With these extracted subtopics, the retrieved results are then diversified using a state of the art diversification method, i.e., *xQuAD* [13]. xQuAD [13] computes the score of the document based on the similarity between the document and the query, the importance of the subtopics and the similarity between the document and subtopics. It iteratively selects the documents that are not only similar to the query but also similar to the subtopics that have not been well covered by previously selected documents.

The diversification results are evaluated using α-nDCG on three depths, 5, 10 and 20 top-ranked documents, which are the official evaluation measures applied in the TREC diversity task [4]. α is the parameter to adjust the contribution of the document covering a relevant while novel subtopic. We set it to be 0.5 which is the default value used in TREC.

We implemented three baseline systems: (1) $NoDiverse$ which is the original retrieval results of Dirichlet function without diversification; (2) $FixedLevel$ which manually sets the levels of subtopics in the hierarchy of the database and selects nodes in those levels that are relevant to the query as subtopics [10]. We use the highest two levels in the hierarchy of enterprise database since the number of nodes on the highest level is too small, i.e., 8; and (3) $QuerySugg$ that submits each query to a web search engine and uses the suggested queries from the web search engine as subtopics. We use DB, $PLSA$ and $Combine$ to denote the methods of extracting subtopics from the database, the documents and integrated information, respectively, as described in Section 3. Each subtopic in $PLSA$ is the cluster with the top-ranked M terms in that cluster.

Table 1: Optimal Performance Comparison of Subtopic Extraction/Integration Methods

Methods	α-nDCG@5	α-nDCG@10	α-nDCG@20
$Nodiverse$	0.279	0.330	0.374
$FixedLevel$	0.256	0.305	0.340
$QuerySugg$	0.280	0.322	0.374
DB	0.220	0.274	0.310
$PLSA$	0.298	0.352	0.404
$Combine$	**0.331**	**0.375**	**0.418**

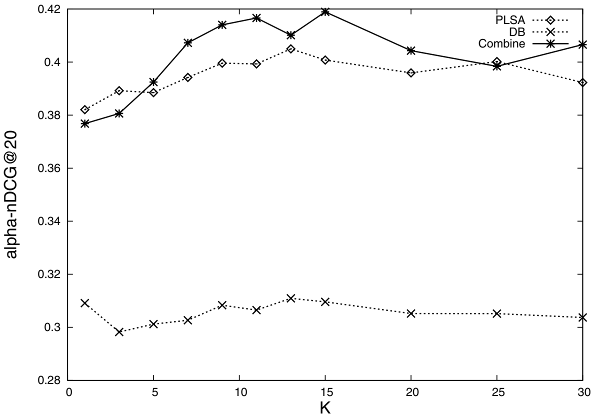

Figure 1: Impact of Number of Subtopics K on the Diversification Performances

4.3 Effectiveness of Subtopic Extraction and Integration Methods

We compare the effectiveness of different subtopic extraction methods and report the results in Table 1. We can make the following interesting observations.

First, $Combine$ outperforms other methods. $Combine$ can integrate the high-quality information in DB subtopics with the information in $PLSA$ that is effective in distinguishing documents covering different subtopics. Therefore, it can outperform both DB and $PLSA$. However, its improvement over $PLSA$, especially on α-nDCG@20, is not significant. Therefore, we further analyze the results of these two methods in every query. We find that $Combine$ outperforms $PLSA$ in more queries, where $Combine$ performs better in 21 queries and $PLSA$ performs better in 16 queries. What's more, $Combine$ is more effective in queries where the diversity performances of the original retrieval results are better. The average α-nDCG@20 value of original retrieval results in queries where $Combine$ performs better is 0.463 while the value in queries where $PLSA$ performs better is 0.351. It shows that we may apply different methods according to the diversity quality of the original retrieval result of the query. We leave this for our future work.

Second, DB performs worse than $PLSA$. This is due to the vocabulary gap between the database and retrieved web pages. Therefore, the subtopics in DB are not effective in diversifying the documents although they contain high-quality information relevant to the query.

Figure 1 shows the diversification performances with different number of subtopics K. The number of subtopic terms M is set to be the optimum value on each point. The optimum numbers of subtopics in $PLSA$ and $Combine$ are 13 and 15, representatively. $Combine$ outperforms $PLSA$ in most values of K. The performance of DB does not change significantly with the values of K because the vocabulary gap between the subtopics of DB and original retrieved documents makes these subtopics ineffective in diversifying documents.

5. CONCLUSIONS AND FUTURE WORK

In this paper, we propose the method of integrating the structured and unstructured data to extract subtopics for search result diversification. The experimental result shows that the integrated subtopics can cover different information of the query and are effective in diversifying the documents. In the future, we plan to study how to diversify search results by utilizing the hierarchical structure of the databases.

6. ACKNOWLEDGEMENTS

This material is based upon work supported by the National Science Foundation under Grant Number IIS-1017026 and HP Labs Innovation Research Program. We thank the anonymous CIKM reviewers for their useful comments.

7. REFERENCES

[1] R. Agrawal, S. Gollapudi, A. Halverson, and S. Ieong. Diversifying Search Results. In *WSDM*, 2009.

[2] B. Carterette and P. Chandar. Probabilistic Models of Novel Document Rankings for Faceted Topic Retrieval. In *CIKM*, 2009.

[3] Z. Chen and T. Li. Addressing diverse user preferences in sql-query-result navigation. In *Proceedings of SIGMOD'07*, 2007.

[4] C. L. A. Clarke, N. Craswell, and I. Soboroff. Overview of the TREC 2009 Web Track. In *Proceedings of TREC'09*, 2009.

[5] E. Demidova, P. Fankhauser, X. Zhou, and W. Nejdl. Divq: Diversification for keyword search over structured databases. In *Proceedings of SIGIR'10*, 2010.

[6] Z. Dou, S. Hu, K. Chen, R. Song, and J. R. Wen. Multi-dimensional search result diversification. In *Proceedings of WSDM'11*, 2011.

[7] H. Fang and C. Zhai. Semantic Term Matching in Axiomatic Approaches to Information Retrieval. In *SIGIR*, 2006.

[8] S. Feldman and C. Sherman. The High Cost of Not Finding Information. In *Technical Report No. 29127, IDC*, 2003.

[9] S. Geva. Gpx - gardens point xml ir at inex 2006. In *Proceedings of INEX'06*, 2006.

[10] C. Hauff and D. Hiemstra. University of Twente @ TREC 2009: Indexing half a billion web pages. In *Proceedings of TREC'09*, 2009.

[11] T. Hofmann. Probabilistic latent semantic analysis. In *Proceedings of UAI'99*, 1999.

[12] P. Lubell-Doughtie and K. Hofmann. Improving result diversity using probabilistic latent semantic analysis. In *Proceedings of DIR'11*, 2011.

[13] R. L. T. Santos, C. Macdonald, and I. Ounis. Exploiting Query Reformulations for Web Search Result Diversification. In *WWW*, 2010.

[14] R. L. T. Santos, C. Macdonald, and I. Ounis. Selectively Diversifying Web Search Results. In *CIKM*, 2010.

[15] C. Zhai and J. Lafferty. A study of smoothing methods for language models applied to ad hoc information retrieval. In *Proceedings of SIGIR'01*, 2001.

[16] W. Zheng, X. Wang, H. Fang, and H. Cheng. An exploration of pattern-based subtopic modeling for search result diversification. In *Proceedings of JCDL'11*, 2011.