Exploiting Concept Hierarchy for Result Diversification

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

Conglei Yao
Tencent
Beijing, China
ycl.pku@gmail.com

ABSTRACT

The goal of result diversification is to maximize the coverage of query subtopics while minimizing the redundancy in the search results. Intuitively, it is more desirable for a diversification system to cover independent subtopics since it would retrieve sets of non-overlapped relevant documents, which leads to less redundancy in the search results. Unfortunately, existing diversification methods assume that query subtopics are independent and ignore their relations in the diversification process. To overcome this limitation, we propose to exploit concept hierarchies to extract query subtopics and infer their relations. We then apply axiomatic approaches to derive a structural diversification method that can leverage the subtopic relations in result diversification. Experimental results over an enterprise collection show that the relations among query subtopics are useful to improve the diversification performance.

Categories and Subject Descriptors

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

General Terms

Algorithm

Keywords

structural diversification, axiomatic approaches, enterprise search, concept hierarchy

1. INTRODUCTION

The goal of search result diversification is to cover all subtopics of the query while minimizing the redundancy in the top-ranked documents to satisfy different information needs of all users [3]. The basic idea of the existing subtopic-based methods [1, 2, 12, 10] is to first identify a set of subtopics for a given query, and then iteratively select documents to maximize the coverage of query subtopics in the search results.

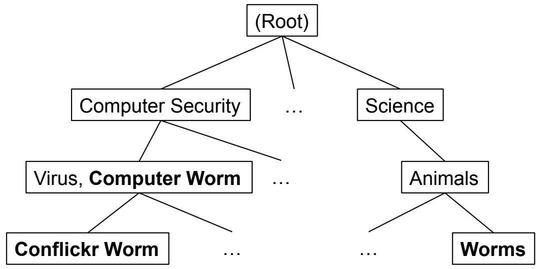

Figure 1: An example of the concept hierarchy.

One major limitation of these existing methods is that they assume query subtopics are independent and ignore the relations among query subtopics. However, the assumption does not hold, and most query subtopics are related to each other. Let us consider query "worm", which is used in TREC 2010 Web track. According to the full query file created by NIST assessors, the query has three subtopics: "computer worms", "worms in nature" and "Conficker worm". It is clear that the subtopic "computer worms" is more related to "Conficker worm" since Conficker worm is a specific computer worm as shown in Figure 1.

Intuitively, subtopic relation is an important factor that needs to be considered in result diversification. When two subtopics are more related, they have a larger set of overlapped relevant documents. Thus, the top-ranked documents covering two independent subtopics are more desirable than those covering two related subtopics since they will provide better coverage and less redundancy in the search results. For example, the top-ranked documents covering subtopics "computer worms" and "worms in nature" are more desirable than those covering "computer worms" and "Conficker worm" since the former can satisfy a wider range of information needs. Unfortunately, existing methods ignore the subtopic relations and would mistakenly think that these two lists are equally diversified, which leads to non-optimal diversification performance. Clearly, it is necessary to study how to incorporate relations among query subtopics to improve diversification performance.

In this paper, we propose a novel *structural* diversification framework that incorporates relations among query subtopics to diversify results based on concept hierarchies, as shown in Figure 1. We first describe how to select nodes from the hierarchy as subtopics for a query. We then use an axiomatic approach to derive a structural similarity function for subtopics based on their structural relations on the con-

CIKM'12, October 29–November 2, 2012, Maui, HI, USA.
Copyright 2012 ACM 978-1-4503-1156-4/12/10 ...$15.00.

cept hierarchy. Finally, we extend an existing diversification method to incorporate the structural similarity function into the diversification process. The proposed structural diversification method iteratively selects documents covering important subtopics that are less structurally similar to the subtopics covered by the selected documents.

The proposed structural diversification method is expected to perform well on the domains with high-quality concept hierarchies. Thus, we conduct experiments in the Enterprise search domain, and find that the proposed method outperforms the state of the art diversification methods.

2. RELATED WORK

Diversification aims to rank documents based on both their relevance and diversity. Previous studies have shown that subtopic-based diversification methods often outperform other methods [10, 15]. In particular, they first identified subtopics, and then used different strategies to diversify documents based on the identified subtopics [1, 10, 17]. Most of the diversification methods assume that query subtopics are independent. However, the assumption does not hold. Most query subtopics are independent no matter whether they are identified using existing methods such as query suggestions [10] or by human assessors for TREC diversity collections [3, 4, 5]. The goal of this paper is to re-examine the assumption and study how to incorporate the relations among query subtopics into the diversification process.

Our work is also related to previous studies on using structural relationships among concepts for retrieval [9, 6, 13]. These studies focused on finding semantically similar terms for query expansion. However, our goal is to study how much new information a subtopic can provide given an existing subtopic which requires a more in-depth analysis of their positions in the concept hierarchy.

3. STRUCTURAL DIVERSIFICATION

A concept hierarchy, such as an ontology, encodes domain knowledge as a hierarchically organized collection of nodes. Each node corresponds to a concept, and the links between nodes indicate semantic relationships between the concepts. The nodes at a higher level contain more general information while nodes at a lower level contain more details. Therefore, the positions of subtopics on the hierarchy can reveal whether they have any overlapped information and can be used to compute the structural similarity between subtopics.

Our basic idea is to exploit the concept hierarchy to identify subtopics and then leverage the structural relations among subtopics in the concept hierarchy to diversify documents. There are two challenges that need to be solved: (1) how to discover query subtopics and infer their structural similarities based on a concept hierarchy; and (2) how to leverage their structural similarities to diversify results.

3.1 Concept Hierarchy based Subtopic Identification

Given a query, we propose to use top ranked documents to find the most relevant nodes from the concept hierarchy as query subtopics. In particular, we assign every top ranked document to its most similar node, and all these selected nodes are regarded as query subtopics. The similarity between document d and node n is computed based on not only the content of n itself but also all of its descendants [16], i.e.,

$$sim(d,n) = \beta \cdot R(d,n) + (1-\beta) \cdot \frac{\sum_{n_j \in desc(n)} R(d,n_j)}{|desc(n)|}, \quad (1)$$

where $|desc(n)|$ is the number of descendants of n, $R(d,n)$ is the relevance score between d and the description of n, and $\beta \in [0,1]$ is a parameter that balances the contribution of relevance score of the node itself and average relevance score of its descendants. $R(d,n)$ can be computed using any existing retrieval functions.

With the identified query subtopics, we can infer their structural similarity based on their positions on the concept hierarchy. Formally, let $\varphi(s_j|s_i)$ denote the structural similarity of subtopic s_j to subtopic s_i, which measures the proportion of information relevant to s_j that is overlapped with those relevant to s_i.

Note that the similarity could be asymmetric. For example, $\varphi(``computer\_worms''|''Conficker\_worm'')$ might not be the same as $\varphi(``Conficker\_worm''|''computer\_worms'')$ because it is used to measure how much information about s_j has been covered by s_i.

Now the challenge is to find an appropriate implementation for $\varphi(s_j|s_i)$. We propose to apply an axiomatic approach to solve the challenge, and the details are discussed in Section 4.

3.2 Concept Hierarchy based Diversification

With the identified query subtopics we now discuss how to extend existing diversification methods to derive structural diversification functions.

First, let us start with one of the state of the art diversification functions, i.e., $xQuAD$ [10]. Given query q and previously retrieved documents D, we select a document that can maximize the ranking score $Score(q,d,D)$ shown as follows:

$$\begin{aligned} Score(q,d,D) \;=\; & (1-\lambda) \cdot \sum_{s \in S(q)} [P(s|q) \cdot P(d|s) \\ & \cdot \prod_{d' \in D} (1 - SubCov(d',s))] + \lambda \cdot P(d|q) \quad (2) \end{aligned}$$

where $S(q)$ is the set of subtopics for query q, $P(d|q)$ is the relevance score of d with respect to q, and $P(s|q)$ is the importance of subtopic s in query q. λ is a parameter balancing relevance and diversity which is set to 0.6 in the experiment. $\prod_{d' \in D}(1 - SubCov(d',s))$ measures the novelty of the subtopic given previously selected documents, where

$$SubCov(d',s) = P(d'|s).$$

Note that the subtopic coverage $SubCov(d',s)$ is to measure how much information from subtopic s that has been covered by the previously retrieved document d'.

We propose to modify the way of computing subtopic coverage by incorporating the structural similarity function between subtopics as follows:

$$SubCov_{struc}(d',s) = \sum_{s' \in S(d')} P(d'|s')P(s'|s), \quad (3)$$

where $S(d')$ is a set of query subtopics that are relevant to a document as described in Section 3.1. $P(s'|s)$ is the likelihood that subtopic s' can be inferred from s. It is estimated by the normalized structural similarities between subtopics:

$$P(s'|s) = \frac{\varphi(s'|s)}{\sum_{s_i \in S} \varphi(s_i|s)}. \quad (4)$$

where $\varphi(s_j|s_i)$ is the structural similarity of s_j to s_i, which will be discussed in the next section.

Plugging Equations (3) and (4) into Equation (2), we have the following structural diversification function:

$$\begin{aligned} Score_{struc}(q,d,S) &= (1-\lambda)\cdot\sum_{s\in S}[P(s|q)\cdot P(d|s) \\ &\quad \cdot\prod_{d'\in D}(1-\sum_{s'\in S(d')}P(d'|s')\frac{\varphi(s'|s)}{\sum_{s_i\in S}\varphi(s_i|s)})] \\ &\quad +\lambda\cdot P(d|q). \end{aligned} \quad (5)$$

4. STRUCTURAL SIMILARITY FUNCTION FOR QUERY SUBTOPICS

Given two subtopics and their positions on the concept hierarchy, we can know how to traverse from one subtopic to the other. We denote the traverse path from subtopic s_i to subtopic s_j as $path(s_i \rightarrow s_j)$. The path consists of one or multiple segments, where each segment is a directed edge between two nodes that follows the traverse direction. The segments going from a node to its parent node are referred to as *UP* segments, and those going from a node to its child node are referred to as *DOWN* segments. $UP(s_i \rightarrow s_j)$ and $DOWN(s_i \rightarrow s_j)$ denote the set of UP and DOWN segments in $path(s_i \rightarrow s_j)$, respectively.

Recall that $\varphi(s_j|s_i)$ is the structural similarity of s_j to s_i, which measures the proportion of information in s_j that can be covered by s_i. It is asymmetric because our goal is to penalize the novelty of the subtopic given another subtopic.

Since a node often covers more general information than its children nodes, it is a natural choice to use the traverse path to compute the structural similarities. One possible solution is given as follows:

$$\varphi(s_j|s_i) = \alpha\cdot f(|UP(s_i\rightarrow s_j)|)+(1-\alpha)\cdot f(|DOWN(s_i\rightarrow s_j)|) \quad (6)$$

where $|X|$ denotes the number of elements in the set X. For example, $|DOWN(s_i \rightarrow s_j)|$ denotes the number of DOWN segments in the path from subtopic s_i to s_j on the concept hierarchy. Moreover, $f(x)$ is a function that controls how the length of UP or DOWN segments, i.e., x, affect the final structural similarities. Since the subtopics with shorter traverse path are often more similar, the function $f(x)$ should be inversely proportional to the number of segments x. A possible solution is shown as follows:

$$f(x) = \frac{1}{1+x} \quad (7)$$

This function assumes that the similarity is linearly correlated with the number of segments. In our preliminary study, we have also tried other possible functions, such as the sublinear and superlinear ones, and found that their performances are similar.

Note that we treat the UP and DOWN segments separately because they represent different term relations. $\alpha \in [0,1]$ is a parameter in Equation (6) that controls how these two types of segments affect the structural similarities. Unfortunately, it is unclear how to set the parameter value. To solve this problem, we propose to use axiomatic approaches. In particular, we first define three similarity constraints that capture the desirable properties of any reasonable structural similarity functions, and then use the constraint analysis to set the parameter value so that the function would satisfy all the constraints.

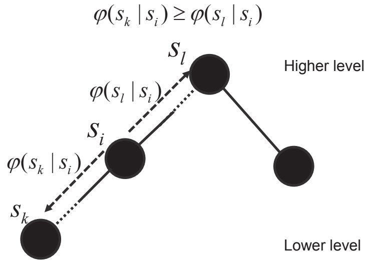

Figure 2: Constraint 3 for structural similarity

4.1 Structural Similarity Constraints

Intuitively, each branch on the concept hierarchy covers a piece of information that is different from the information of other branches. A node at higher levels often contains more branches and therefore is a summary of the information contained in its descendants. Based on this assumption, we define three constraints that measures the structural similarity of two subtopics based on the proportion of their overlapped information on the concept hierarchy.

- **Constraint 1**: *The similarity of a subtopic to itself should not be smaller than that of any other subtopic.*

  Formally, let us assume that s_i and s_j are two subtopics, where $s_i \neq s_j$. Thus, we always have $\varphi(s_i|s_i) \geq \varphi(s_j|s_i)$.

- **Constraint 2**: *The structural similarity of a subtopic's ancestor to the subtopic should not be smaller than that of its ancestor's ancestor subtopic.*

  Formally, we consider three subtopics s_i, s_k and s_l. If we know that s_k is an ancestor of s_i, i.e., $|UP(s_i \rightarrow s_k)| > 0$ and $|DOWN(s_i \rightarrow s_k)| = 0$, and s_l is an ancestor of s_k, i.e,. $|UP(s_k \rightarrow s_l)| > 0$ and and $|DOWN(s_k \rightarrow s_l)| = 0$, then we have $\varphi(s_k|s_i) \geq \varphi(s_l|s_i)$.

  This constraint is motivated by the fact that ancestors further away from a subtopic often have more branches on the hierarchy containing more novel information.

- **Constraint 3**: *The structural similarity of a subtopic's any descendant to the subtopic should not be smaller than that of the subtopic's any ancestor subtopic, as shown in Figure 2.*

  Formally, s_i, s_k and s_l denote three subtopics. If we know that s_l is an ancestor of s_i, i.e., $|UP(s_i \rightarrow s_l)| > 0$, $|DOWN(s_i \rightarrow s_l)| = 0$, and s_k is a descendant of s_i, i.e., $|UP(s_i \rightarrow s_k)| = 0$, $|DOWN(s_i \rightarrow s_k)| > 0$, then we have $\varphi(s_k|s_i) \geq \varphi(s_l|s_i)$.

  The constraint is motivated by the fact that a subtopic's ancestor covers more novel information than the subtopic itself, while the subtopic's descendants cover only more specific information that has been covered by the subtopic itself. For example, a user has seen a document about "computer worm". The system will assign higher novelty to the ancestor "computer security" and select documents covering it than the descendant "Conflickr Worm". This is consistent with the goal of diversification, i.e., maximizing the coverage of the query and minimizing the redundancy.

Table 1: Optimal Performance.

Methods		α-nDCG		
		@5	@10	@20
NoDiverse		0.279	0.331	0.375
xQuAD	*QuerySugg*	0.293	0.327	0.375
	FixedLevel	0.262	0.306	0.341
Structural		**0.381** ♦▲	**0.420** ♦▲	**0.459** ♦▲

The three constraints define a set of basic properties for the structural similarity function so that it can be leveraged in the diversification process. It is unclear whether the constraints form a complete set of all the desirable properties, and we plan to explore more constraints in the future work.

4.2 Constraint Analysis

With the constraints, we analyze the structural similarity function shown in Equation (6) to check whether it satisfies the constraints.

The function always satisfies the first constraint because

$$\varphi(s_i|s_i) - \varphi(s_i|s_j) \quad = \quad 1 - \varphi(s_i|s_j) > 0.$$

Next, we analyze Constraint 2. We denote $|UP(s_i \rightarrow s_k)| = x$ and $|UP(s_i \rightarrow s_l)| = y$. And we know $y > x \geq 1$ given the positions of these three subtopics. Thus,

$$\varphi_(s_k|s_i) - \varphi_(s_l|s_i) = \frac{\alpha}{1+x} - \frac{\alpha}{1+y} > 0$$

It is clear that the function satisfies the Constraint 2.

Finally, we check whether the function satisfies the last constraint. We denote $|DOWN(s_i \rightarrow s_k)| = x$ and $|UP(s_i \rightarrow s_l)| = y$, where $x \geq 1$ and $y \geq 1$. Thus, we have

$$\varphi(s_k|s_i) - \varphi_(s_l|s_i) = \alpha + \frac{(1-\alpha)}{1+x} - \frac{\alpha}{1+y} - (1-\alpha).$$

In order to satisfy the constraint, i.e., $\varphi(s_k|s_i) - \varphi_(s_l|s_i) \geq 0$, we have

$$\alpha \quad \geq \quad \frac{\frac{x}{1+x}}{\frac{x}{1+x} + \frac{y}{1+y}}.$$

Since we have $x \geq 1$ and $y \geq 1$, we can derive the bounds for the parameter is $\alpha \geq \frac{2}{3}$. Thus, we set α to be $\frac{2}{3}$.

In summary, the following structural similarity function that will be used in our structural diversification function (i.e., Equation (5)):

$$\varphi(s_j|s_i) = \frac{\frac{2}{3}}{1 + |UP(s_i \rightarrow s_j)|} + \frac{\frac{1}{3}}{1 + |DOWN(s_i \rightarrow s_j)|}. \qquad (8)$$

5. EXPERIMENTS

5.1 Experiment Design

To evaluate the effectiveness of the proposed structural diversification method, we need to conduct experiments on an important search domain with high-quality concept hierarchies. We choose the enterprise search domain since, with the increasing usage of taxonomies in enterprise search [8], almost every enterprise often has its own concept hierarchies, which are either manually built by the domain expert or automatically inferred from the enterprise data. In particular, we use an enterprise search diversification data set constructed in our previous study [16]. The data set consists of: (1) a document collection with 477,800 Intranet pages; (2) a concept hierarchy related to the enterprise; (3) a query set with 50 queries. For each query, human assessors create a set of subtopics and label the relevance of the document with respect to each subtopic. The average number of subtopics per query is 4.12.

Table 2: Cross Validation (α-$nDCG$@20)

Methods		Train		Test	
		Avg.	Deviation	Avg.	Deviation
xQuAD	*QuerySugg*	0.379	0.017	0.362	0.070
	FixedLevel	0.341	0.020	0.344	0.079
Structural		**0.463** ♦▲	0.027	**0.457** ♦▲	0.103

Methodology: We first retrieve a list of relevant documents and use them to select subtopics from the concept hierarchy as described in Section 3.1. We then apply the structural diversification functions shown in Equation (5).

Baselines: To compare the proposed methods with the state of the art, we implemented the following methods: (1) *NoDiverse*, which ranks search results based on only relevance using Dirichlet Prior retrieval function [14]; (2) Two variants of *xQuAD* [10] based on the subtopic identification strategies: (a) *QuerySugg*, which uses suggested queries of Web search engines as subtopics [11]; and (b) *FixedLevel*, which selects subtopics from the top level of the concept hierarchy [7]. Note that all the diversification methods re-rank the results of *NoDiverse*.

Evaluation Measures: We use one of the official measures used for the diversity task at TREC Web track [3], i.e., α-$nDCG$@20 as the primary measure. α is set to 0.5. We also report the performance measured with α-$nDCG$@5 and α-$nDCG$@10. α-$nDCG$ actually assumes that the subtopics are independent, which makes it difficult for the structural methods to get good performance. Therefore, it can prove that the structural method is more effective in diversifying results if it outperforms the state-of-the-art methods based on α-$nDCG$.

5.2 Effectiveness of Structural Diversification

We first compare the optimal performance of structural diversification method and the baselines. All parameters except λ in different diversification methods are tuned to the optimal values. The results are shown in Table 1. ▲and ♦indicate that the performance improvement of *Structural* over *FixedLevel* and *QuerySugg* are statistically significant at 0.05 level. It is clear that the proposed structural diversification method can statistically significantly outperform all the baseline methods. The better performance suggests that the structural relationships among subtopics are important in the diversification process.

Another interesting observation is that existing diversification methods cannot effectively diversify the results from this enterprise collection. We find that this is caused by the quality of the subtopics. *QuerySugg* method uses query suggestions from Web search engine, which are independent to the collection and thus cannot effectively diversity the documents in the enterprise collection. *FixedLevel* is forced to select subtopics at the top level, and these subtopics may not be the most effective ones to diversify results.

The structural diversification method is mainly different from the *xQuAD* methods in two components. One is the subtopics extracted from the concept hierarchy and the other is the structural diversification function. In order to check

Table 3: Constraint Verification (α-nDCG@20)

	Constraint 1	Constraint 2	Constraint 3
NonConsPerf	0.372	0.337	0.090
ConsPerf	**0.378**	**0.349**	**0.106**

whether both parts contribute to the performance improvement, we also implement a variant of *xQuAD* that uses subtopics extracted from the concept hierarchy, which could enable us to focuses on the effectiveness of individual components. The optimal α-nDCG@20 performance of this variant is 0.435 which is better than other variants of *xQuAD*, i.e., 0.375 of *QuerySugg* and 0.341 of *FixedLevel*, while worse than the performance of structural diversification method, i.e., 0.459. Clearly, both the subtopic extraction and structural diversification components are effective.

Finally, we train the parameters used in the diversification methods by performing a 5-fold cross validation over the 50 queries. The training is optimized for the primary evaluation measure, i.e., α-*nDCG*@20. Table 2 shows the results of cross validation. We can see that, with the trained parameters, the proposed structural diversification method can still perform significantly better than the baseline methods.

5.3 Constraint Verification

We have defined three constraints to derive structural similarity functions for query subtopics. To verify whether the constraints are reasonable, we design the following experiments to test the effectiveness of each constraint.

First, we construct document sets from the search results to make sure that, for every set, the associated subtopic structure fits the ones described in the three constraints. For example, d_i, d_k and d_l are three documents in the original result which are related s_i, s_k and s_l in Figure 2, respectively. Therefore, $\{d_i, d_k, d_l\}$ is a combination of documents whose structure fits the constrain structures.

After that, for each document set, we generate two sets of diversification results: *ConsPerf*, which contains the re-rankings the documents based on the corresponding constraint; and *NonConsPerf*, whose results violate the constraint. Let us consider the above example again. The re-ranking $\{d_i, d_l, d_k\}$ satisfy constraint 3 and the re-ranking $\{d_i, d_k, d_l\}$ does not. Table 3 compares the average diversification performance of the two sets The performance of *ConsPerf* is better in every constraint, which indicates that the constraint satisfaction for a structural similarity function is related to the structural diversification performance.

6. CONCLUSIONS AND FUTURE WORK

The paper aims to break the limitation of existing diversification methods, which assume that query subtopics are independent to each other. The contribution of this paper can be summarized as follows: (1) we propose to use the structural relations among subtopics to diversify search results; and (2) we use an axiomatic approach to derive the structural similarity function for subtopics based on their positions on the concept hierarchy, and then derive new diversification methods with these similarity functions.

Enterprise search is a domain that expects to benefit the most from the proposed diversification methods, since the enterprise data often contain concept hierarchies that are more complementary to the information from enterprise document collections. Our experimental results show that the structural diversification method can significantly outperform the state-of-the-art methods.

There are many interesting future directions. First, we will study how to adaptively apply the structural diversification method based on the quality of hierarchy. Second, we will exploit more constraints to derive new structural diversification methods.

7. ACKNOWLEDGMENTS

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.

8. REFERENCES

[1] R. Agrawal, S. Gollapudi, A. Halverson, and S. Ieong. Diversifying Search Results. In *Proceedings of WSDM'09*, pages 5–14, New York, NY, 2009. ACM.

[2] B. Carterette and P. Chandar. Probabilistic Models of Novel Document Rankings for Faceted Topic Retrieval. In *Proceedings of CIKM'09*, pages 1287–1296, New York, NY, 2009. ACM.

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

[4] C. L. A. Clarke, N. Craswell, I. Soboroff, and G. V. Cormack. Overview of the TREC 2010 Web Track. In *Proceedings of TREC'10*, 2009.

[5] C. L. A. Clarke, N. Craswell, I. Soboroff, and E. M. Voorhees. Overview of the TREC 2011 Web Track. In *Proceedings of TREC'11*, 2011.

[6] H. Fang. A Re-examination of Query Expansion Using Lexical Resources. In *Proceedings of ACL'08*, New York, NY, 2008. ACM.

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

[8] D. Hawking. Challenges in Enterprise Search. In *Proceedings of ADC'04*, pages 15–24, Darlinghurst, Australia, 2004. Australian Computer Society, Inc.

[9] M. Lalmas. *XML Retrieval (Synthesis Lectures on Information Concepts, Retrieval, and Services)*. Morgan and Claypool, San Rafael, CA, 2009.

[10] R. L. T. Santos, C. Macdonald, and I. Ounis. Exploiting Query Reformulations for Web Search Result Diversification. In *Proceedings of WWW'10*, pages 881–890, New York, NY, 2010. ACM.

[11] R. L. T. Santos, C. Macdonald, and I. Ounis. Selectively Diversifying Web Search Results. In *Proceedings of CIKM'09*, pages 1179–1188, New York, NY, 2010. ACM.

[12] R. L. T. Santos, J. Peng, C. Macdonald, and I. Ounis. Explicit search result diversification through sub-queries. In *Proceedings of ECIR'10*, pages 87–99, New York, NY, 2010. Springer.

[13] R. Thiagarajan, G. Manjunath, and M. Stumptner. Computing semantic similarity using ontologies. In *HPLabs Tech Report*, 2008.

[14] C. Zhai and J. Lafferty. A study of smoothing methods for language models applied to ad hoc information retrieval. In *Proceedings of SIGIR'01*, pages 334 – 342, New York, NY, 2001. ACM.

[15] W. Zheng and H. Fang. A comparative study of search result diversification methods. In *Proceedings of DDR'11*, 2011.

[16] W. Zheng, H. Fang, C. Yao, and M. Wang. Search result diversification for enterprise search. In *Proceedings of CIKM'11*, New York, NY, 2011. ACM.

[17] W. Zheng, X. Wang, H. Fang, and H. Cheng. Coverage-based search result diversification. *Journal of Information Retrieval*, 2011.