Search Result Diversification Based on Hierarchical Intents

Sha Hu[1,3], Zhicheng Dou[2,3,*], Xiaojie Wang[2,3], Tetsuya Sakai[4], and Ji-Rong Wen[1,3]
[1]Beijing Key Laboratory of Big Data Management and Analysis Methods, China
[2]Key Laboratory of Data Engineering and Knowledge Engineering, MOE, China
[3]School of Information, Renmin University of China
[4]Waseda University
[1]{sallyshahu,jirong.wen}@gmail.com, [2]{dou,wangxiaojie}@ruc.edu.cn,
[4]tetsuyasakai@acm.org

ABSTRACT

A large percentage of queries issued to search engines are broad or ambiguous. Search result diversification aims to solve this problem, by returning diverse results that can fulfill as many different information needs as possible. Most existing intent-aware search result diversification algorithms formulate user intents for a query as a flat list of subtopics. In this paper, we introduce a new hierarchical structure to represent user intents and propose two general hierarchical diversification models to leverage hierarchical intents. Experimental results show that our hierarchical diversification models outperform state-of-the-art diversification methods that use traditional flat subtopics.

Keywords

Search result diversification; hierarchical diversification; hierarchical intents

1. INTRODUCTION

In web search, the majority of short queries are ambiguous or broad when it comes to specifying a user's information need [16, 18, 25, 33, 34]. For example, by issuing an ambiguous query `[apple]`, one user might be searching for information about the IT company Apple, whereas another user might be looking for information about the fruit. By issuing a broad query `[harry potter]`, a user may want to seek contents covering various aspects, such as `[harry potter movie]`, `[harry potter book]`, or `[harry potter characters]` within this broad topic. Traditional search may fail to cover these different intents in the top ranks.

As an effective way to solve this problem, search result diversification, which aims to return diverse search results that cover as many user intents as possible, has received a lot of attention in recent years. Many search result diversification algorithms [1, 2, 5, 14, 23, 25, 26, 31, 32, 36, 37, 42] and evaluation metrics [1, 4, 8, 9, 28, 39] have been developed to improve and evaluate search result diversity. Some public search result diversification evaluation tasks, including the TREC Web Track Diversity task [11] and the NTCIR Intent/IMine task [22, 27], have been organized to evaluate diversification approaches via public test collections.

In diversification, most existing algorithms generalize user intents in a flat list of independent subtopics, such as topical categories [1, 35], query reformulations by search engine [12, 13, 30], words or phrases extracted from top retrieved documents [3, 13], or combined subtopics from multiple external resources [14, 17]. For instance, Santos *et al.* [30] represented different query intents as a set of Google suggestions or related queries. Dang and Croft [13] extracted words or phrases from top retrieved documents. Dou *et al.* [14] mined subtopics from different types of resources.

In typical search result diversification tasks (such as TREC and NTCIR), the intents of a query are predefined by human labellers. To achieve good performance, it is critical to automatically mine subtopics for a query and the mined subtopics should properly match the predefined intents. However, as the intents and subtopics are independently obtained, it is not easy to match them, especially when a flat intent list is used.

Let us take the query "defender" (topic number 20) in TREC 2009 [6] as an example. Table 1 shows six manually defined intents for it, including "Windows Defender Homepage", "Land Rover Defender", "Defender Marine Supply", "Defender Arcade Game Online", "Windows Defender Reports," and "Chicago Defender Newspaper". To mine subtopics for the query, we use the approach proposed in [12, 13, 30]. We send the query "defender" to a commercial search engine and get back query suggestions "defender windows", "defender arcade game," and "defender land rover" (The first-level subtopics in Figure 1). When we take them as subtopics for the query "defender", we find that they are too coarse to distinguish the user intents. For instance, the subtopic "defender windows" (t_1) covers both the user intents "Windows Defender homepage" (s_1) and "Windows Defender Reports" (s_5). So the diversity algorithm has the risk to only select documents from subtopic t_1 with respect to the intent s_1 without covering subtopic s_5, or vice versa.

An easy remedy to the above problem is to use fine-grained intents in diversification. By further sending the three subtopics as queries to the search engine, we can get their query suggestions as fine-grained subtopics (the second-level subtopics in Figure 1). With the fine-grained subtopics, we can do a better job to distinguish user intents. For example, "defender windows home" ($t_{1,1}$) and "defender windows prob-

[*] Corresponding Author

CIKM'15, October 19–23, 2015, Melbourne, VIC, Australia.
© 2015 ACM. ISBN 978-1-4503-3794-6/15/10 ...$15.00.
DOI: http://dx.doi.org/10.1145/2806416.2806455.

Table 1: Subtopics of query "defender" in TREC 2009 Web Track.

no.	subtopic description
s_1	I'm looking for the homepage of Windows Defender, an anti-spyware program. **(Windows Defender Homepage)**
s_2	Find information on the Land Rover Defender sport-utility vehicle. **(Land Rover Defender)**
s_3	I want to go to the homepage for Defender Marine Supplies. **(Defender Marine Supply)**
s_4	I'm looking for information on Defender, an arcade game by Williams. Is it possible to play it online? **(Defender Arcade Game Online)**
s_5	I'd like to find user reports about Windows Defender, particularly problems with the software. **(Windows Defender Reports)**
s_6	Take me to the homepage for the Chicago Defender newspaper. **(Chicago Defender Newspaper)**

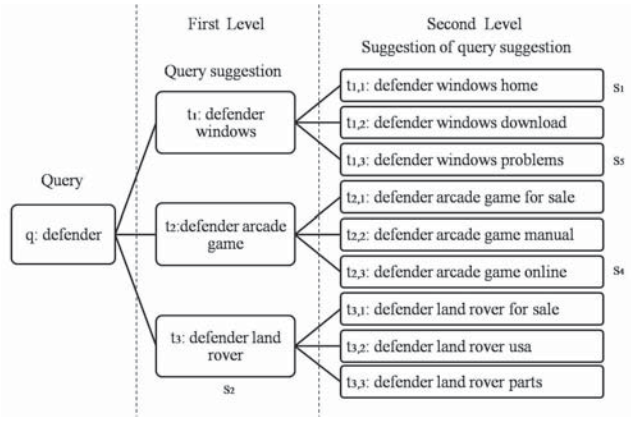
First Level
Second Level
Suggestion of query suggestion
Query suggestion
Query
q: defender
t1: defender windows
t2:defender arcade game
t3: defender land rover
S2
t1,1: defender windows home
S1
t1,2: defender windows download
t1,3: defender windows problems
S5
t2,1: defender arcade game for sale
t2,2: defender arcade game manual
t2,3: defender arcade game online
S4
t3,1: defender land rover for sale
t3,2: defender land rover usa
t3,3: defender land rover parts

Figure 1: Two-level hierarchical subtopics of query "defender" from query suggestions of a commercial search engine.

lems" ($t_{1,3}$) can be easily matched to user intents "Windows Defender homepage" (s_1) and "Windows Defender Reports" (s_5). However, the fine-grained subtopics bring in a new problem that multiple subtopics are matched to the same user intent. For example, "defender land rover for sale" ($t_{3,1}$), "defender land rover usa" ($t_{3,2}$), and "defender land rover parts" ($t_{3,3}$) are all related to the user intent "Land Rover Defender" (s_2). When simply combining the fine-grained subtopics in a flat list, the diversity algorithm may first select one document from subtopic $t_{3,1}$ ("defender land rover for sale"), and then select another document from subtopic $t_{3,2}$ ("defender land rover usa"), unaware of the fact that the selected documents match the same user intent s_2 ("Land Rover Defender").

To solve the above problems, we explore a new way of organizing subtopics in a hierarchical structure. Figure 1 shows the two-level hierarchical subtopic structure for the query "defender". We can see that the six user intents in Table 1 are mapped to the subtopics in both two levels. This hierarchical structure maintains user intents with different granularity levels and the relationships between different levels. It provides the flexibility of applying and balancing different levels of intent granularity in diversification, which ultimately increases the probability of correctly matching more diverse intents.

In this paper, we propose a diversification framework to explicitly leverage the hierarchical intents. In particular, we extend the state-of-the-art diversification algorithms xQuAD and PM2, and propose a Hierarchical xQuAD model (HxQuAD) and a Hierarchical PM2 model (HPM2). These hierarchical models select documents that maximize diversity in the hierarchical structure. For the example in Figure 1, if the previously selected documents have already cover the subtopic "defender windows home", the next selected document for the intent "defender windows" should better come from the subtopic "defender windows download" or "defender windows problems". Without the hierarchical structure, it is difficult for traditional diversification methods to distinguish and balance these second-level subtopics. Moreover, after selecting documents for the second-level subtopics "defender windows home" and "defender windows problems", which belong to the same first-level subtopic "defender windows," the next document is better to be related to another first-level subtopic such as "defender arcade game" or "defender land rover". This situation is not considered in traditional diversification methods when only the flat intent list is used.

We argue that real user intents are in a hierarchical structure sometimes. Recent work [15, 19] understands queries by query facets and represents query facets by multiple groups of facet items, where these query facets and their facet items can be viewed as a two-level hierarchical explanation of the query. In Table 1, real user intents of the query also lie in a two-level hierarchy. It contains five first-level subtopics "Windows Defender", "Land Rover Defender" (s_2), "Defender Marine" (s_3), "Defender Arcade Game" (s_4), and "Chicago Defender Newspaper" (s_6). The subtopic "Windows Defender" contains two second-level subtopics "Windows Defender Homepage" (s_1) and "Windows Defender Reports". To measure the real user satisfaction on the query, it would be ideal to evaluate diversity based on hierarchical intents. Unfortunately we do not have such diversity judgment data and evaluation metrics. In this paper, we still use existing metrics to evaluate our algorithms.

Our experiments are conducted on two-level hierarchical subtopics which are automatically extracted from a commercial search engine. The algorithms are evaluated on the public TREC [7] dataset and the NTCIR [22] dataset. Experimental results show that by using hierarchical subtopics, our hierarchical algorithms (HxQuAD and HPM2) outperform most state-of-the-art models including xQuAD [30], PM2 [12], TxQuAD, and TPM2 [13], in terms of ERR-IA [4], α-NDCG [8], NRBP [9], and D♯-nDCG [28]. Even when only using single-level subtopics in the hierarchical structure, our hierarchical algorithms still outperform their corresponding algorithms which represent the same subtopics in a flat list, in terms of ERR-IA, α-NDCG, and NRBP. The results show that exploiting the hierarchical intent structure can benefit search result diversification.

2. RELATED WORK

In early diversification algorithms, query intents were *implicitly* considered to promote diversity by selecting documents with little content similarity. Maximal Marginal Relevance (MMR) [2] measured documents with cosine similarities in vocabulary, and attempted to reduce redundancy while maintaining query relevance in re-ranking. Some work estimated document similarity in different ways, such as using Kullback-Leibler divergence [38], ranking sentences by random walks in an absorbing Markov chain [42], and modeling directed graphs based on the document link structure [40]. These approaches assume that similar documents will cover similar query intents, without considering exactly which query intents are being covered.

Many researchers noticed the above problem and *explicitly* considered query intents for diversification. IA-Select [1] developed an intent-aware diversifying method by classified topical categories for queries and documents based on ODP taxonomy. xQuAD [30] used a greedy algorithm to maximize the coverage of query aspects. RxQuAD [35] explicitly provided a relevance formulation for query aspects. PM2 [12] divided diversification into two processes: finding the best unsatisfied topic-by-topic proportionality, and then choosing the best document based on the selected subtopic. Intrinsic diversity [26] predicted the successor queries for initiator query to seek which content to cover. Yu and Ren [36] formulated diversification as a 0-1 multiple subtopic knapsack problem. Fusion diversification [21] inferred latent subtopics based on topic modeling. Some work promoted diversity by leveraging subtopics from multiple external resources, such as involving user clicks to help query aspects [24], combining subtopics from different data types [14, 17]. Although these approaches generate query intents from various sources, combinations or models, they commonly represent intents in a flat list. In contrast, our work utilizes hierarchical intents and provides hierarchical frameworks to promote diversity of search results.

Some researchers use machine learning techniques to diversify search results, such as Structural SVMs [37] and R-LTR [43]. In this paper, we focus on the unsupervised intent-aware diversification. More specifically, we extend xQuAD to hierarchical novelty-based model, and adapt PM2 for hierarchical proportionality-based model.

There are a few prior art that are somewhat similar to ours. Term level diversification [13] represented a subtopic by a set of key terms, which is similar to our idea of denoting a subtopic with a group of child subtopics in hierarchical subtopic structure. Unlike considering hierarchical information as we do, term level diversification integrated all terms to build subtopics. We implement their basic models as our baselines in our experiments. Concept hierarchy based diversification [41] considered subtopic relations in result diversification. Instead of providing hierarchical frameworks to handle hierarchical subtopics, it exploited concept hierarchies to extract query subtopics in a flat list, utilized hierarchical relations of subtopics to propose a structural similarity function for subtopics, and incorporated this function into the traditional xQuAD framework. This function iteratively selected documents covering important subtopics that are less structurally similar to the subtopics covered by the selected documents. This work was done in enterprise search domain, as it is hard to build high-quality concept hierarchies in web search. We implement this method by adopting our hierarchical subtopics used in our hierarchical models as their input concept hierarchy.

There have been some approaches [15, 19, 20, 29] that mine hierarchical information for a query. In the present study, we use Google suggestions as the source of hierarchical subtopics, following previous work [12, 13, 30]. We will explore our own subtopic mining methods in future work.

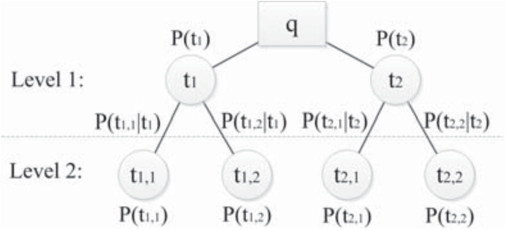

Figure 2: An example of hierarchical subtopic tree.

3. HIERARCHICAL DIVERSIFICATION

3.1 Hierarchical Intents

As mentioned earlier, most intent-aware diversification algorithms model user intents as a group of subtopics and boost diversity based on them. In this paper, we propose to present subtopics in a hierarchical structure.

Formally, we use $T_q = \{t_1, t_2, ...\}$ to indicate a set of first-level subtopics $\{t_{i_1}\}$ for query q where $i_1 = 1, 2...$ is the position of subtopic t_{i_1} in T_q. For subtopic t_{i_1}, we use $T_{i_1} = \{t_{i_1,1}, t_{i_1,2}, ...\}$ to denote a set of its child subtopics $\{t_{i_1,i_2}\}$. For each subtopic $t_{i_1,i_2} \in T_{i_1}$, i_2 is the relative index of subtopic t_{i_1,i_2} within all child subtopics of t_{i_1}. To generalize, we use $t_{i_1,.,i_j}$ to denote a subtopic at level j and use $T_{i_1,.,i_j}$ to denote a set of its child subtopics. $t_{i_1,.,i_{j+1}} \in T_{i_1,.,i_j}$ and $t_{i_1,.,i_{j+1}}$ is a specific child subtopic of $t_{i_1,.,i_j}$ at level $j + 1$.

Figure 2 shows an example of the hierarchical subtopics in two levels. The first level contains two subtopics t_1, t_2 and the second level has four subtopics $t_{1,1}, t_{1,2}, t_{2,1}, t_{2,2}$. $t_{1,1}$ and $t_{1,2}$ are child subtopics of t_1, and $t_{2,1}$ and $t_{2,2}$ are child subtopics of t_2. Here exists $T_1 = \{t_{1,1}, t_{1,2}\}$ and $T_2 = \{t_{2,1}, t_{2,2}\}$.

Let us reuse the example topic "defender" in Figure 1 to illustrate the process. The subtopic "defender windows" contains three child subtopics: "defender windows home" shows users want the homepage of the software, "defender windows download" indicates the download requirement of the software, and "defender windows problems" denotes users are interested in the problems of the software. Similarly, the first-level subtopics "defender arcade game" and "defender land rover" have their own second-level child subtopics.

It is worth noting that term level diversification [13] also indicates a subtopic by a set of terms $t_i = \{t_i^1, t_i^2, ., t_i^j, ...\}$. They treat each t_i^j as an independent subtopic, and integrate them together to build a larger flat term level subtopics $\{t_1^1, t_1^2, ..., t_1^{|t_1|}, ..., t_n^1, t_n^2, ..., t_n^{|t_n|}\}$. Its diversification algorithm is not aware of the relationship between t_i^j and t_i, and the relationship between two terms. In contrast, we maintain the subtopics in hierarchy structure and diversify search results based on these hierarchical subtopics.

For a given query q, we use $R = \{d_1, d_2, ..., d_m\}$ to denote its initial ranked documents set. For traditional diversification algorithms that use a flat list of subtopics, we use $T = \{t_1, t_2, ..., t_n\}$ to denote subtopics of the query. Let $P(d|q)$ be the probability that document d is relevant to query q, $P(d|t)$ indicate the probability that d satisfies subtopic t, and $P(t|q)$ denote the importance of subtopic t for query q. Existing diversification algorithms use T, $P(d|q)$, $P(d|t)$, and $P(t|q)$ to select a list of diversified documents D out of R. Similarly, for hierarchical diversification, we define the following probabilities:

$\mathbf{P(t_{i_1,..,i_{j+1}}|t_{i_1,..,i_j})}$. $P(t_{i_1,..,i_{j+1}}|t_{i_1,..,i_j})$ is the importance of a subtopic $t_{i_1,..,i_{j+1}}$ with respect to its parent subtopic $t_{i_1,..,i_j}$. We assume that $t_{i_1,..,i_j}$ is fully covered by its child subtopic set $T_{i_1,..,i_j}$ and each of the child subtopic is independent to each other. Hence we have:

$$\sum_{t_{i_1,..,i_{j+1}} \in T_{i_1,..,i_j}} P(t_{i_1,..,i_{j+1}}|t_{i_1,..,i_j}) = 1$$

In Figure 2, we have $P(t_{1,1}|t_1)$+$P(t_{1,2}|t_1)$=1 and $P(t_{2,1}|t_2)$+ $P(t_{2,2}|t_2)$=1.

$\mathbf{P(t_{i_1,..,i_j}|q)}$. $P(t_{i_1,..,i_n}|q)$ is the importance of subtopic $t_{i_1,..,i_j}$ with respect to the query q. The way of calculating $P(t_{i_1,..,i_j}|q)$ may vary in different applications. If the importance of leaf subtopics is known (for example, when parent subtopics are generated by clustering child subtopics), we can update the importance of their ancestors by iteratively summing up the importance of child subtopics, i.e., we have:

$$P(t_{i_1,..,i_j}|q) = \sum_{t_{i_1,..,i_{j+1}} \in T_{i_1,..,i_j}} P(t_{i_1,..,i_{j+1}}|q) \tag{1}$$

In some other cases, we may just know the importance of the first-level subtopics. For example, when building the hierarchy subtopics based on Google suggestion, we need to first retrieve first-level subtopics, and then retrieve second-level subtopics by issuing the first-level subtopics as queries. In this case, we may calculate the weight of each child subtopic by using Bayes's formula:

$$P(t_{i_1,..,i_{j+1}}|q) = P(t_{i_1,..,i_j}|q) \cdot P(t_{i_1,..,i_{j+1}}|t_{i_1,..,i_j})$$

In Figure 2, if $P(t_1|q)$ is given, $P(t_{1,1}|q)$=$P(t_1|q) \cdot P(t_{1,1}|t_1)$, $P(t_{1,2}|q)$=$P(t_1|q) \cdot P(t_{1,2}|t_1)$. If $P(t_{1,1}|q)$ and $P(t_{1,2}|q)$ are known, $P(t_1|q)$=$P(t_{1,1}|q) + P(t_{1,2}|q)$.

$\mathbf{P(d|t_{i_1,..,i_j})}$. $P(d|t_{i_1,..,i_j})$ is the probability that document d satisfies $t_{i_1,..,i_j}$. $P(d|t_{i_1,..,i_j}) = P(d|t_{i_1,..,i_j}, q)$ because we assume $P(q|t_{i_1,..,i_j}) = 1$. We assume that leaf subtopics $t_{i_1,..,i_n}$ are usually represented as words or phrases, and we can directly calculate $P(d|t_{i_1,..,i_n})$ on leaf subtopics using language model or other retrieval model. For non-leaf subtopics, instead of words or phrases, they may be organized as groups of their child subtopics (e.g., when second-level subtopics are Google suggestions, and the first-level subtopics are clusters of these suggestions). In this case, we use a bottom-up method to recursively calculate $P(d|t_{i_1,..,i_j})$ for a subtopic $t_{i_1,..,i_j}$ based on its child subtopics $T_{i_1,..,i_j}$ as follows:

$$P(d|t_{i_1,..,i_j}) = 1 - \prod_{t_{i_1,..,i_{j+1}} \in T_{i_1,..,i_j}} (1 - P(d|t_{i_1,..,i_{j+1}})) \tag{2}$$

where $(1 - P(d|t_{i_1,..,i_{j+1}}))$ is the probability that d dose not satisfy $t_{i_1,..,i_{j+1}}$. The product denotes the probability that d fails to satisfy every child subtopic for $t_{i_1,..,i_j}$. One minus that product equals the probability that d will satisfy at least one child subtopic $t_{i_1,..,i_{j+1}}$ within $T_{i_1,..,i_j}$.

3.2 Hierarchical Diversification Algorithms

3.2.1 Topic novelty model

The topic novelty model, inspired by MMR [2], is a widely used framework in diversification. It considers both the relevance between the document and the query, and the topic diversity of the document among the selected documents. It iteratively selects a next best document d that is relevant to query q and can maximize the diversity of selected documents D. The formulation of the model is as below.

$$d^* = \arg\max_{d \in R \setminus D} (1 - \lambda) \cdot P(d|q) + \lambda \cdot \Phi(d, D) \tag{3}$$

Different probabilistic models are proposed to measure the diversity $\Phi(d, D)$ in previous work [1, 3, 30]. We select xQuAD, one of the state-of-the-art diversification methods, to adapt a diversity model to use hierarchical structured subtopics. xQuAD explicitly estimates the diversity of a document by calculating the coverage of matched subtopics.

$$\Phi(d, D) = \sum_{t \in T_q} [P(d|t) \cdot P(t|q) \cdot \prod_{d' \in D} (1 - P(d'|t))] \tag{4}$$

In the above equation, $(1 - P(d'|t))$ indicates the probability that an existing document d' does not satisfy t. The product shows the probability that all the selected documents D fail to satisfies t. Summing up over all subtopics, weighted by $P(t|q)$, the diversity is the probability that d covers the subtopics while the existing document list D fail to satisfy.

However, Equation (4) is designed for the subtopics formed as a flat list, which may fail when real user intents are hierarchical. We take the query intents shown in Figure 2 as an example. Assume there are four documents: d_1 and d_2 relevant to $t_{1,1}$, d_3 relevant to $t_{1,2}$, and d_4 relevant to $t_{2,1}$. One of the ideal rank lists is ($d_1 \rightarrow d_4 \rightarrow d_3 \rightarrow d_2$) and the diversity is maximized within the top three results. If we just use the first-level subtopics in xQuAD, the returned diversified rank list might be ($d_1 \rightarrow d_4 \rightarrow d_2 \rightarrow d_3$). In the third iteration, it fails to distinguish the difference between d_2 and d_3 because both are relevant to t_1. If we just use the second-level subtopics in xQuAD, the resulting list might be ($d_1 \rightarrow d_3 \rightarrow d_4 \rightarrow d_2$). In this case, it assumes that d_3 and d_4 are equally important because both can offer a new subtopic, but in fact, d_4 is better because both d_3 and d_1 belong to subtopic t_1.

To solve the above problem, we adapt xQuAD so that it can handle hierarchical subtopics. We propose **HxQuAD**, a hierarchical xQuAD model, to explicitly model result diversity based on the hierarchical subtopics. Specifically, at each level j of the hierarchical subtopic tree, HxQuAD estimates result diversity by:

$$\Phi(d, D, j) = \sum_{|i_1,.,i_j|=j} [P(d|t_{i_1,..,i_j}) \cdot P(t_{i_1,..,i_j}|q) \cdot \prod_{d' \in D} (1 - P(d'|t_{i_1,..,i_j}))] \tag{5}$$

Here $|i_1,.,i_j| = j$ means $t_{i_1,..,i_j}$ is a subtopic at level j. $P(d|t_{i_1,..,i_j})$ and $P(t_{i_1,..,i_j}|q)$ estimate the probabilities that d satisfies $t_{i_1,..,i_j}$ and $t_{i_1,..,i_j}$ satisfies q, and we have

$$P(d|t_{i_1,..,i_j}) = 1 - \prod_{t_{i_1,..,i_{j+1}} \in T_{i_1,..,i_j}} (1 - P(d|t_{i_1,..,i_{j+1}}))$$

which is the same as Equation (2). This model evaluates the importance of document d based on whether it can improve overall diversity in terms of the subtopics at level j.

We then combine these components, and evaluate the overall importance of a document in terms of all levels within the hierarchy. A parameter α is introduced to control the granularity of subtopics that the diversification tends to optimize for. We have:

$$\begin{aligned}&\Phi(d,D) = \alpha \cdot \Phi(d,D,1) + (1-\alpha)\cdot \Phi(d,D,2)+\\ &\frac{(1-\alpha)^2}{\alpha}\cdot \Phi(d,D,3) + ... + \frac{(1-\alpha)^{n-1}}{\alpha^{n-2}}\cdot \Phi(d,D,n)\end{aligned} \tag{6}$$

Here $\alpha \in (0,1]$ and $\alpha = 0.5$ indicates that all the subtopic levels are equally weighted. A value larger than 0.5 means that the algorithm tends to diversify result based on coarse subtopics; whereas a value lower than 0.5 indicates that it provides fine-grained diversify. Specially, if $\alpha = 1$, then it only uses the first-level subtopics; whereas, if α is close to 0, the model tends to take the leaf subtopics. Note that α could be 0 if there are only two levels in hierarchy.

In summary, HxQuAD extends xQuAD to hierarchical subtopics by redefining a multi-level diversity function. Besides balancing the relevance and diversity by parameter λ, we use a parameter α to control the impact of subtopics at different depth.

3.2.2 *Topic proportionality model*

Instead of considering diversity of subtopics and documents at the same time, the topic proportionality based diversification model selects subtopics and documents separately. At each iteration, the algorithm first selects the best subtopic based on the proportionality strategy, and then finds the most relevant document optimized for the selected subtopic.

PM2 [12] is one of the state-of-the-art topic proportionality based diversification algorithms. It considers the diversification problem as assigning seats to members of competing political parties and follows a highest quotient (Sainte-Lague) method to select subtopics as allocating seats. In the beginning, PM2 computes the quotient qt_i for each subtopic t_i by the Sainte-Lague formula[1].

$$qt_i = \frac{w_i}{2s_i + 1} \tag{7}$$

To maintain the proportionality of the subtopic distribution, PM2 assigns the subtopic with the largest quotient as the selected subtopic t_{i^*}. Then, it calculates the diversity function $\Phi(d,D,t^*)$ to find the document d^* that is most relevant to t_{i^*} and relatively relevant to other subtopics.

$$d^* = \arg\max_{d \in R\setminus D} \Phi(d,D,t^*)$$

where

$$\Phi(d,D,t^*) = \lambda \cdot qt_{i^*}\cdot P(d|t_{i^*}) + (1-\lambda)\cdot \sum_{i \neq i^*} qt_i \cdot P(d|t_i) \tag{8}$$

After document d^* is selected, to punish its highly relevant subtopics, PM2 increases the "portion" of occupied seats s_i for each subtopic t_i by its normalized relevance to d^*.

$$s_i = s_i + \frac{P(d^*|t_i)}{\sum_{t_j \in T_q} P(d^*|t_j)} \tag{9}$$

The algorithm repeats the above process to iteratively select next best documents from R to D.

In this paper, we modify the framework of PM2 to adapt hierarchical subtopics and propose the hierarchical PM2 model (**HPM2**). HPM2 maintains the basic idea of finding the best document based on the preselected subtopic by proportionality. Moreover, since each level of hierarchical subtopics may contain different diversity information, HPM2 selects one best subtopic for each level of the subtopic tree, and combine them together to find the best document. Considering Figure 2 as an example, we may select t_1 with max quotient from the first-level and $t_{1,1}$ with max quotient for the second-level, and find the best document based on them. Note that we may choose t_1 and $t_{2,1}$ sometimes, as the best subtopics are selected independently.

First of all, HPM2 computes the quotient values for the subtopics in each level, respectively. For the subtopic $t_{i_1,.,i_j}$ at level j, the quotient is similarly formulated as Equation (7). Note that $P(t_{i_1,.,i_j}|q)$ is the probability that $t_{i_1,.,i_j}$ satisfies q.

$$qt_{i_1,.,i_j} = \frac{P(t_{i_1,.,i_j}|q)}{2s_{i_1,.,i_j} + 1} \tag{10}$$

Comparing all the quotients of the subtopics at level j, HPM2 selects the best subtopic $t^*_{i_1,.,i_j}$ with max quotient value $qt^*_{i_1,.,i_j}$. The best subtopics $t^*_{i_1}$, $t^*_{i_1,i_2}$, ..., $t^*_{i_1,.,i_n}$ are respectively chosen from level $1,2,...,n$ in hierarchy. HPM2 calculates the document diversity for each level of hierarchical subtopics and combines all to select the best document.

For level j in the hierarchical subtopics, since $t^*_{i_1,.,i_j}$ is the preselected subtopic at level j, according to the diversity definition in PM2, a document is more diverse if it is relevant to $t^*_{i_1,.,i_j}$ and fairly related to other subtopics of this level. Therefore, we define $\Phi(d,D,t^*_{i_1,.,i_j})$ as:

$$\begin{aligned}&\Phi(d,D,t^*_{i_1,.,i_j}) = \lambda \cdot qt^*_{i_1,.,i_j}\cdot P(d|t^*_{i_1,.,i_j}) +\\ &\quad (1-\lambda)\cdot \sum_{t_k \neq t^*_{i_1,.,i_j}, |k|=j} qt_k \cdot P(d|t_k)\cdot P(t_k|t^*_{i_1,.,i_j})\end{aligned} \tag{11}$$

Here $P(t_k|t^*_{i_1,.,i_j})$ is used to model the dependency between t_k and the selected subtopic $t^*_{i_1,.,i_j}$. We use it because treating all the unselected subtopics equally is not fair in the hierarchy. A close subtopic t, which may share common parent, grandparent, or ancestor with t^*, is more related to t^* than other subtopics. For instance, if $t_{1,1}$ is the selected subtopic, $t_{1,2}$ is usually more semantically related to $t_{1,1}$ than $t_{2,1}$ and $t_{2,2}$. So we should assign a higher weight to $t_{1,2}$ than $t_{2,1}$ and $t_{2,2}$ in Equation (11). We use the following function to evaluate the weight of a subtopic based on its distance to the selected subtopic.

$$P(t_{i_1,.,i_j}|t^*_{i_1,.,i_j}) = \frac{2j - dis(t_{i_1,.,i_j}, t^*_{i_1,.,i_j}) + 1}{2j} \tag{12}$$

where $dis(t,t^*)$ is the length of the path for moving from t to t^*. Since both subtopics are at level j, the maximal distance between t and t^* is $2j$. It is used to normalize the distance. Considering the example in Figure 2, we have $P(t_2|t_1) = (2\cdot 1-2+1)/(2\cdot 1) = 0.5$, $P(t_{1,1}|t_{1,2}) = (2\cdot 2-2+1)/(2\cdot 2) = 0.75$ and $P(t_{2,1}|t_{1,2}) = (2\cdot 2-4+1)/(2\cdot 2) = 0.25$.

[1] Sainte-Lague: `http://www.elections.org.nz/voting/mmp/sainte-lague.html`

Based on Equation (11), we further combine all levels, and find the best document d^* by the following formula.

$$d^* = \arg\max_{d \in R \setminus D} \alpha \cdot \Phi(d, D, t^*_{i_1}) + (1-\alpha) \cdot \Phi(d, D, t^*_{i_1,i_2}) + \frac{(1-\alpha)^2}{\alpha} \cdot \Phi(d, D, t^*_{i_1,i_2,i_3}) + ... + \frac{(1-\alpha)^{n-1}}{\alpha^{n-2}} \cdot \Phi(d, D, t^*_{i_1,..,i_n}) \tag{13}$$

Similar to Equation (6), parameter $\alpha \in (0, 1]$ is used to control the impact of subtopic granularity.

At last, HPM2 updates the occupied seat $s_{i_1,..,i_j}$ for the subtopic $t_{i_1,..,i_j}$ based on the selected document d^* as follows. Note that the update is respectively done for each level in hierarchy, as the best subtopic is selected at each level.

$$s_{i_1,..,i_j} = s_{i_1,..,i_j} + \frac{P(d^*|t_{i_1,..,i_j})}{\sum_{|k|=j} P(d^*|t_k)} \tag{14}$$

In short, HPM2 selects each best subtopic by proportionality, finds the document based on selected subtopics and updates the occupied seats by the chosen document, following the steps of PM2. In addition, HPM2 performs subtopic selecting and seats updating at each level of the hierarchical subtopics and finds the best document by considering n selected subtopics at the same time. It uses a distance function to control the influence of the unselected subtopics by considering their distances to selected subtopics.

4. EXPERIMENTAL SETUP

4.1 Datasets

We experiment with the proposed algorithms on four topic sets provided by TREC Web Tracks from 2009 [6] to 2012 [7]. Every topic set contains 50 topics, each of which includes three to eight subtopics. Topics 95 and 100 in the TREC 2010 topic set were removed as they lack diversity relevance judgments. Following the official task definitions, we use the ClueWeb09 [10] document collection for the four topic sets. We merge the four datasets into one and name it as TREC in this paper.

For each topic, we retrieve top 1,000 documents using the batch search service provided by Lemur project[2]. Similar to existing approaches [12, 13], we remove the documents with spam score larger than 70 using the Waterloo Spam Filter[3] for ClueWeb09. We take these filtered documents as our initial non-diversified ranking results, and conduct our experimental results on top 50 documents as existing work has found that both xQuAD and PM2 achieve their best performance when using 50 documents [12]. We estimate the relevance between documents and topics by the language model used in Lemur[2]. And we use the same model to calculate the relevance between documents and subtopics.

In addition, we use the dataset provided by the IMine (Intent Mining) task in NTCIR-11 [22]. We use the Chinese topic set which is comprised of 18 clear queries and 32 ambiguous or broad queries, and name this dataset as NTCIR in this paper. We evaluate the algorithms using the 32 ambiguous or broad queries, as diversity relevance judgments are not provided for the clear queries. We use the official non-diversified baseline retrieval results from the SogouT2008[4] corpus provided by the organizers. Note that, in contrast to the TREC data, IMine task provides two-level human annotated subtopics and we can evaluate diversity using both coarse-grained and fine-grained search intents.

4.2 Evaluation Metrics

We use ERR-IA [4], α-NDCG [8], and NRBP [9], which are official evaluation metrics at TREC Web Track, to evaluate result diversity. They measure the diversity of a result list by explicitly rewarding novelty and penalizing redundancy observed at every rank. We use the same parameters as those used in official TREC tasks, and hence per-subtopic graded relevance assessments are treated as binary. In addition, we use D$\sharp$-measures [28], the primary metric that used in NTCIR IMine task, which actually utilizes graded diversity relevance judgments. All metrics are computed based on the top 20 ranking results, consistent with the official tasks in TREC 2010-2012. Moreover, we use the two-tailed paired t-test for statistically significance testing and report a significant difference if the p-value is lower than 0.05.

4.3 Generation of hierarchical subtopics

Similar to previous work [12, 30], we use query suggestions extracted from Google search engine as subtopics (as shown in Figure 1). To avoid Google's personalized suggestions, we clean the cookies and set the location to United States before query suggestion crawling.

For each topic, we collect its query suggestions from Google as the first-level subtopics. To generate subtopic hierarchy, we further issue the first-level subtopics as queries to Google and retrieve their query suggestions as the second-level subtopics. Sometimes Google fails to provide suggestions for some queries. When a topic has no suggestion, we view this topic as invalid and omit it. If a first-level subtopic from a valid topic has no suggestion, we add itself as its second-level subtopic to ensure the two-level structure. Finally, we collect 1,696 first-level subtopics and 10,527 second-level subtopics for 194 queries. We only consider two-level hierarchical subtopics in this paper, and leave the investigation of using third and deeper levels to future work.

Consistent with existing research [30, 12], we assume a uniform probability distribution for all the first-level subtopics, i.e., $P(t_i|q) = \frac{1}{|T_q|}$ where T_q is the set of the first-level subtopics. We also assume a uniform probability distribution for the second-level subtopics with respect to their parent subtopics. We use Equation (1) to calculate the importance of a second-level subtopic with respect to the query. For example, for the j^{th} second-level subtopic $t_{i,j}$ of the i^{th} first-level subtopic t_i, its importance $P(t_{i,j}|q)$ is $\frac{1}{|T_q| * |T_i|}$ where $|T_i|$ is the count of second-level subtopics of t_i.

4.4 Baseline Models

We compare our proposed models with the following baseline approaches: the non-diversified baseline ranking (Baseline), xQuAD, PM2, TxQuAD, TPM2, and ConceptH. We already introduced **xQuAD** and **PM2** in Section 3.2 and we recall that our hierarchical diversification models are extended from them. Term level diversification models, viz., **TxQuAD** and **TPM2** [13], split the original subtopics (used by xQuAD and PM2) into terms and use these terms as

[2] Batch Service Clueweb09: http://boston.lti.cs.cmu.edu/Services/clueweb09_batch/

[3] Waterloo Spam Filter: http://plg.uwaterloo.ca/~gvcormac/clueweb09spam/

[4] SogouT2008: http://www.sogou.com/labs/dl/t-e.html

Table 2: Performance comparison on TREC 2009-2012. The best result is in bold. Statistically significant differences between the hierarchical methods (HxQuAD and HPM2) and the baseline methods (xQuAD, TxQuAD, PM2, TPM2, and ConceptH) are marked with $*$, $\star$, $\diamond$, $\circ$, $\dagger$, respectively.

	ERR-IA	α-nDCG	NRBP	D♯-nDCG
Baseline	.2630	.3610	.2238	.4124
xQuAD*	.2842	.3822	.2465	.4109
TxQuAD*	.2792	.3835	.2396	.4189
PM2$^\diamond$	.2952	.3990	.2548	.4289
TPM2°	.2805	.3895	.2385	.4256
ConceptH†	.3002	.4064	.2607	.4366
HxQuAD	$.3206^{*\star\dagger}_{\diamond\circ}$	$.4229^{*\star}_{\diamond\circ}$	$.2845^{*\star\dagger}_{\diamond\circ}$	$.4378^{*\star}_{\diamond\circ}$
HPM2	$\mathbf{.3235}^{*\star\dagger}_{\diamond\circ}$	$\mathbf{.4234}^{*\star}_{\diamond\circ}$	$\mathbf{.2880}^{*\star\dagger}_{\diamond\circ}$	$\mathbf{.4381}^{*\star}_{\diamond\circ}$

their subtopics. In order to maintain the consistency of the subtopics, we did not use the DSPApprox method introduced in [13] that extracts terms from search results. **ConceptH** [41] exploits concept hierarchies to find flat query subtopics and infers subtopic relations in traditional diversification by hierarchical similarities of subtopics. We directly take our subtopic hierarchy (See Subsection 4.3) as the concept hierarchy in ConceptH, which means that ConceptH shares the same subtopic hierarchy with HxQuAD and HPM2. All the baseline methods have a parameter λ that requires tuning. Our hierarchical models have two parameters to tune: the traditional parameter λ and the hierarchical weight parameter α. We use a **5-fold cross validation** to tune these parameters in terms of ERR-IA@20 on TREC data and NTCIR data, respectively.

The hierarchical subtopics and all runs can be found on the website: `http://www.playbigdata.com/dou/hdiv` .

5. EXPERIMENTAL RESULTS

5.1 Overall results

We compare our models with the baseline approaches. For the baseline models, we use the first-level subtopics which is a common approach in existing work [12, 13, 30]. The results are shown in Table 2 and Table 3. We find that:

(1) The hierarchical diversification models, i.e., HxQuAD and HPM2, outperform all baseline methods on the TREC 2009-2012 dataset. Table 2 shows that HxQuAD and HPM2 have statistically significant improvements in terms of ERR-IA, α-nDCG, NRBP, and D♯-nDCG ($p<0.05$ with two-tailed paired t-tests). Specifically, HPM2 outperforms xQuAD, TxQuAD, and TPM2 by more than three hundredths; outperforms PM2 by more than two hundredths; outperforms ConceptH by more than one hundredth, in terms of ERR-IA, α-nDCG, and NRBP.

(2) As mentioned before, the diversity on the NTCIR IMine dataset can be evaluated either on the coarse-grained intents or on the fine-grained intents. Table 3 shows that HxQuAD and HPM2 outperform all baseline algorithms in terms of ERR-IA, α-nDCG, and NRBP at either the coarse-grained level or the fine-grained level, though their improvements are not statistically significant. Please note that there are only 32 topics on NTCIR data. The results indicate that the hierarchical models provide more diverse results than the baseline models when users are interested in either coarse-grained subtopics or fine-grained subtopics.

Table 3: Performance comparison on NTCIR data.

	ERR-IA	α-nDCG	NRBP	D♯-nDCG
(a) Evaluated by using coarse-grained intents				
Baseline	.3044	.4508	.2644	.3952
xQuAD*	.3146	.4666	.2682	.4079
TxQuAD*	.3282	.4915	.2852	.4266
PM2$^\diamond$	.3200	.4865	.2702	.3596
TPM2°	.3239	.5115	.2699	.4493
ConceptH†	.3203	.4807	.2731	.4185
HxQuAD	.3436	.4901	**.3064**	.4094
HPM2	**.3449**	**.5150**	.2975	**.4507**
(b) Evaluated by using fine-grained intents				
Baseline	.1596	.3497	.1338	.2782
xQuAD*	.1669	.3616	.1396	.2853
TxQuAD*	.1717	.3836	.1458	.2929
PM2$^\diamond$	.1641	.3695	.1348	.2825
TPM2°	.1609	.3929	.1222	**.3135**
ConceptH†	.1684	.3652	.1403	.2837
HxQuAD	**.1831**	.3823	**.1598**	.2881
HPM2	**.1831**	**.3994**	.1543	.3081

(3) Both HxQuAD and HPM2 significantly outperform their corresponding models xQuAD, TxQuAD, PM2, and TPM2 on TREC data, and outperform their corresponding models by more than one to three hundredths on both levels of NTCIR data, in terms of ERR-IA, α-nDCG, and NRBP. This indicates that utilizing hierarchical subtopics, even just containing two levels, could help promote result diversity than traditional subtopics formed as a flat list.

(4) Recall that ConceptH leveraged subtopic hierarchies to calculate subtopic dependencies for traditional flat subtopics. The results show that ConceptH outperforms all the other baseline models on TREC data. However, both HxQuAD and HPM2 outperform ConceptH in terms of all metrics. They significantly outperform ConceptH in terms of ERR-IA and NRBP on TREC data, and outperform ConceptH by more than one to three hundredths on both levels of NTCIR data, in terms of ERR-IA, α-nDCG, and NRBP. Therefore, when using subtopic hierarchies in diversification, proposing hierarchical frameworks for hierarchical subtopics works better than improving traditional frameworks on subtopic relations for flat subtopics.

(5) Consistent with the recent work [12], PM2 performs slightly better than xQuAD in terms of all metrics on the TREC dataset in Table 2. And they perform similarly in the coarse-grained intent of the NTCIR data in Table 3. Term level models TxQuAD and TPM2 perform differently on different datasets. On the TREC data, they underperform their corresponding models xQuAD and PM2 in terms of most metrics. But on NTCIR IMine data, they outperform their corresponding models in terms of most metrics, and are the top performers in some cases. Based on the study made by Dang *et al.* [13], they work very closely to their corresponding models. After analyzing the results, we found that term level models do not work well on topics containing phrases. For instance, considering topic number 165 "blue throated hummingbird" in TREC 2012, it has a subtopic "blue throated hummingbird picture" meaning to find the pictures of the specific bird. Term level models

in our experiments use the split terms "blue", "throated", "hummingbird," and "picture" as the subtopics, which may select some noisy irrelevant documents relevant to "blue" or "picture". Dang et al. [13] claimed that they could identify phrases within the subtopics (for example, split the above example subtopic to "blue throated hummingbird" and "picture"), which may help improve the performance of term-level diversification algorithms. However, we used the Stanford Tokenizor[5] and it failed to detect such kinds of phrases. This might be one of reasons why TxQuAD and TPM2 do not work well in our experiments. For the NTCIR data, we use the ICTCLAS Chinese tokenizer[6]. The tokenizer can recognize named entities and phrases, and may generate better term-level subtopics than on TREC data.

5.2 Impact of using different levels of subtopics

The analysis so far has shown that the proposed hierarchical models using hierarchical subtopics outperform the baseline models with flat-formed subtopics (xQuAD, TxQuAD, PM2, and TPM2). There are at least two possible explanations for it. The first is that our two-level hierarchical approach for diversification is indeed effective. The second is that our approach outperformed the baseline models simply because we used extra subtopics (i.e., second-level subtopics) while the baseline models only used first-level subtopics.

To clarify this problem, we experiment with baseline models using different levels of subtopics. We want to investigate whether our models still outperform baseline models with the same subtopics in a flat list. For fair comparison, we provide baseline models exactly the same subtopics as hierarchical models, by organizing hierarchical subtopics into a flat list, and set a uniform weight for them. We use xQuAD$_{1st}$, xQuAD$_{2nd}$, and xQuAD$_{all}$ to denote using first-level subtopics, second-level subtopics, and all subtopics (merged by the obtained suggestions in two levels) for xQuAD model. Similar symbols are used for other models.

Recall that, in hierarchical models, the impact of first-level subtopics and second-level subtopics are controlled by a parameter α according to Equation (6) and Equation (11). When $\alpha = 1$, only the first-level subtopics are used in hierarchical models. When $\alpha = 0$, hierarchical models only use the second-level subtopics. Due to space limitation of the paper, we only report the results on the TREC dataset. The results are shown in Table 4 and Table 5. The best results are in bold, statistically significant differences between hierarchical methods and their corresponding methods are respectively marked in the upper right corner, and all related parameters are tuned by 5-fold cross validations.

5.2.1 Using first-level subtopics only

Table 4 and Table 5 show that, by utilizing first-level subtopics, hierarchical models outperform all their corresponding models in terms of ERR-IA, α-nDCG, and NRBP. Specifically, HxQuAD$_{1st}$ significantly outperforms xQuAD$_{1st}$ in terms of all metrics.

The difference between hierarchical models and their counterparts is **the relevance estimation $P(d|t_i)$ between the document d and the first-level subtopic t_i**. In traditional diversification models, i.e., xQuAD$_{1st}$ and PM2$_{1st}$, the probability that document d satisfies subtopic t_i, i.e., $P(d|t_i)$, is calculated by language model [30]. For hierarchical models, only the relevances between documents and second-level (leaf) subtopics are directly computed as traditional methods. The relevances between documents and first-level (non-leaf) subtopics are estimated by $P(d|t_i) = 1 - \prod_{t_{i,j} \in T_i}(1 - P(d|t_{i,j}))$ as Equation (2) in Subsection 3.1. This design makes our hierarchical models more flexible to handle different subtopic mining algorithms, which may contain virtual subtopics at the non-leaf levels in hierarchy.

Table 4: Performance comparison of HxQuAD and its corresponding methods using different subtopics.

	ERR-IA	α-nDCG	NRBP	D$\sharp$-nDCG
(a) Using first-level subtopics only				
xQuAD$^{*}_{1st}$	.2842	.3822	.2465	.4109
TxQuAD$^{\diamond}_{1st}$	.2792	.3835	.2396	.4189
HxQuAD$^{\dagger}_{1st}$	**.3054**$^{*\diamond}$	**.4041***	**.2683***	**.4259***
(b) Using second-level subtopics only				
xQuAD$^{\star}_{2nd}$	.2956	.3940	.2609	.4207
TxQuAD$^{\circ}_{2nd}$	.2850	.3903	.2452	.4171
HxQuAD$^{\ddagger}_{2nd}$	**.3145**$^{*\star\divideontimes}_{\diamond\circ}$	**.4160**$^{*\star\divideontimes}_{\diamond\circ\bullet}$	**.2782**$^{*\divideontimes}_{\diamond\circ}$	**.4334**$^{*\divideontimes}_{\diamond\circ\bullet}$
(c) Using all subtopics (both two levels)				
xQuAD$^{\divideontimes}_{all}$	.2948	.3930	.2572	.4193
TxQuAD$^{\bullet}_{all}$	.2898	.3930	.2508	.4185
HxQuAD$_{all}$	**.3206**$^{*\star\divideontimes\dagger}_{\diamond\circ\bullet\ddagger}$	**.4229**$^{*\star\divideontimes\dagger}_{\diamond\circ\bullet\ddagger}$	**.2845**$^{*\star\divideontimes\dagger}_{\diamond\circ\bullet\ddagger}$	**.4378**$^{*\star\divideontimes\dagger}_{\diamond\circ\bullet\ddagger}$

The results indicate that using the relevance estimation between documents and subtopics in hierarchical models, makes the results more diverse than the traditional way. One of the possible reasons is that, by involving the child subtopics into estimation, finding a relevant document for the subtopic becomes finding a relevant document for its child subtopics. The document related to more child second-level subtopics will be viewed as more relevant to the parent first-level subtopic. For example, a first-level subtopic t_1 has two relevant documents d_1, d_2. Traditional models find that the documents' relevances are $P(d_1|t_1) = 0.8$ and $P(d_2|t_1) = 0.7$, and think d_1 is more relevant to t_1. Hierarchical models check the child second-level subtopics of t_1 and find that d_1 is related to one child subtopic $P(d_1|t_{1,1}) = 0.7$ and d_2 is related to two child subtopics $P(d_2|t_{1,1}) = 0.6$, $P(d_2|t_{1,2}) = 0.4$. Then hierarchical models treat d_2 as more relevant to t_1 as $P(d_2|t_1)$=$1 - (1 - 0.6) * (1 - 0.4)$=0.76 and $P(d_1|t_1)$=$1 - (1 - 0.7)$=0.7. In fact, since d_2 covers more child subtopics, it contains more diversity information than d_1 and should be valued more in diversification. By utilizing the child subtopics to find relevant documents for the first-level subtopics, hierarchical models increase the relevances for the documents covering more child subtopics, and provide better result diversity than traditional models.

5.2.2 Using second-level subtopics only

When adopting second-level subtopics, hierarchical models still outperform all their corresponding models in terms of ERR-IA, α-nDCG, and NRBP, as shown in Table 4 and Table 5. In particular, HxQuAD$_{2nd}$ significantly outperforms xQuAD$_{2nd}$ in terms of ERR-IA and α-nDCG, and HPM2$_{2nd}$ outperforms PM2$_{2nd}$ by more than one hundredth in terms of ERR-IA, α-nDCG, and NRBP.

The difference between hierarchical models and their counterparts is **the relevance probability $P(t_{i,j}|q)$ between**

[5] Stanford Tokenizor: `http://nlp.stanford.edu/software/tokenizer.shtml`

[6] ICTCLAS: `http://www.ictclas.org/`

Table 5: Performance comparison of HPM2 and its corresponding methods using different subtopics.

	ERR-IA	α-nDCG	NRBP	D♯-nDCG
(a) Using first-level subtopics only				
PM2$^{*}_{1st}$	.2952	.3990	.2548	**.4289**
TPM2$^{\diamond}_{1st}$	.2805	.3895	.2385	.4256
HPM2$^{\dagger}_{1st}$	**.3070**$^{\diamond}$	**.4055**	**.2693**$^{\diamond}$	.4280
(b) Using second-level subtopics only				
PM2$^{\star}_{2nd}$	.3054	.4104	.2670	**.4380**
TPM2$^{\circ}_{2nd}$	.2847	.3925	.2435	.4206
HPM2$^{\ddagger}_{2nd}$	**.3172**$^{*\circledast}_{\diamond\circ\bullet}$	**.4189**$^{*\circledast\dagger}_{\diamond\circ\bullet}$	**.2806**$^{*\circledast}_{\diamond\circ\bullet}$	.4373$^{*\dagger}_{\diamond\circ\bullet}$
(c) Using all subtopics (both two levels)				
PM2$^{\circledast}_{all}$	.3022	.4072	.2620	.4353
TPM2$^{\bullet}_{all}$	.2879	.3948	.2485	.4211
HPM2$_{all}$	**.3225**$^{*\star\circledast\dagger}_{\diamond\circ\bullet\ddagger}$	**.4234**$^{*\star\circledast\dagger}_{\diamond\circ\bullet\ddagger}$	**.2880**$^{*\star\circledast\dagger}_{\diamond\circ\bullet\ddagger}$	**.4381**$^{*\dagger}_{\diamond\circ\bullet}$

the query q and the second-level subtopic $t_{i,j}$. Recall that, in subtopic hierarchy, we set a uniform probability distribution for the first-level subtopics of the query, i.e., $P(t_i|q) = \frac{1}{|T_q|}$ where $|T_q|$ is the count of the first-level subtopics (See Subsection 4.3). And we set a uniform weight for each second-level subtopic with respect to its parent first-level subtopic, i.e., $P(t_{i,j}|t_i) = \frac{1}{|T_i|}$, where $|T_i|$ is the count of the child subtopics in the first-level subtopic t_i. Then we calculate the relevance probabilities between queries and second-level subtopics by Equation (1) in Subsection 3.1, i.e., $P(t_{i,j}|q) = P(t_{i,j}|t_i) * P(t_i|q) = \frac{1}{|T_q|*|T_i|}$. On the other side, in traditional models, since their subtopics are in a flat list, they set a uniform relevance probability distribution for the second-level subtopics of the query, i.e., $P(t_{i,j}|q) = \frac{1}{\sum|T_i|}$, where $\sum|T_i|$ indicates the total number of all the second-level subtopics. For example, consider topic number 3 "getting organized" in TREC 2009, which has 9 first-level subtopics and 52 second-level subtopics in total. In traditional models, all second-level subtopics share the same relevance to the query $P(t_{i,j}|q) = \frac{1}{52} = 0.019$. In hierarchical models, for a first-level subtopic t_1, "getting organized at work", it has 9 child second-level subtopics and their relevance probabilities to the query are $P(t_{1,j}|q) = \frac{1}{9} * \frac{1}{9} = 0.012$; for another first-level subtopic t_2, "getting organized for college", it has 3 child second-level subtopics whose relevance probabilities to the query are $P(t_{2,j}|q) = \frac{1}{9} * \frac{1}{3} = 0.037$.

The results indicate that hierarchical models assign better relevance probabilities for second-level subtopics with respect to queries than traditional models with flat subtopics. One possible reason is that, many second-level subtopics may come from some coarse first-level subtopics, so the documents related to these coarse subtopics may be overvalued in traditional models who treat all second-level subtopics as equally important. On the contrary, by passing the relevance of queries uniformly from first-level subtopics to second-level subtopics, hierarchical models control the total contribution of second-level subtopics from coarse first-level subtopics in diversification. Continue the upper example, assume that document d_1 is highly related to all second-level subtopics from t_1, $P(d_1|t_{1,i})$=1 (i=1,2,..,9), and document d_2 is highly related to all second-level subtopics from t_2, $P(d_2|t_{2,j})$=1 (j=1,2,3). Traditional models think d_1 is more diverse as d_1 is related to more subtopics. $\sum P(d|t_{1,i}) * P(t_{1,i}|q)$=$9 * 1 * 0.019$=0.171 $> \sum P(d|t_{2,j}) * P(t_{2,j}|q)$ =$3 * 1 * 0.019$=0.057. Hierarchical models find that d_2 is also important because d_2 is related to more relevant subtopics. $\sum P(d|t_{1,i}) * P(t_{1,i}|q)$ =$9 * 1 * 0.012$=0.11 $= \sum P(d|t_{2,j}) * P(t_{2,j}|q)$=$3*1*0.037$=0.11. Accurately, d_1 and d_2 should be equally important in diversity since either d_1 or d_2 is only related to one first-level subtopic (t_1 or t_2). Therefore, involving first-level subtopics in assigning relevances for second-level subtopics is a fair choice in hierarchical models. The experimental results show that considering first-level subtopics in estimating the relevances between subtopics and queries is very helpful in result diversification.

5.2.3 *Using all subtopics*

Table 4 and Table 5 show that, by using all the subtopics, our hierarchical models significantly outperform their corresponding models in terms of ERR-IA, α-nDCG, and NRBP. This indicate that, when considering two level subtopics in diversification, our hierarchical models with two-level hierarchy are better than traditional diversification models which merge two-level subtopics in a flat list.

Moreover, for HxQuAD and HPM2, using subtopic hierarchy is always better than using single-level subtopics, and the improvements are significant in terms of ERR-IA, α-nDCG, and NRBP. This means that incorporating the entire hierarchy is better than the sole use of a single-level of subtopics in hierarchical models. Each level plays its own role on diversifying search results.

The results also show that all traditional models with the second-level subtopics outperform their counterparts with the first-level subtopics in terms of most metrics. The improvements are mostly not significant, but they indicate that the traditional models perform better by using the second-level (fine-grained) subtopics than by using the first-level (coarse) subtopics. In a preliminary study in TREC data, we found that there are more fine-grained or specific subtopics than coarse or general subtopics. For example, for the "defender" example shown in Section 1, many predefined subtopics (including s_1, s_4, and s_5) correspond to second-level subtopics we extracted from the commercial search engine. This means that exploiting more fine-grained subtopics could help improve result diversity. xQuAD and PM2 perform worse when using all subtopics than using second-level subtopics. One possible reason is that the merged subtopics include many overlapped subtopics, which may involve redundant documents in search results. So the second-level subtopics is good enough for the flat-subtopic based models, and involving the all subtopics is unnecessary and risky.

In short, our hierarchical models with subtopic hierarchy significantly outperform all their corresponding models with different levels of subtopics in terms of ERR-IA, α-nDCG, and NRBP. This indicates that our two-level hierarchical models are indeed effective in diversification.

6. CONCLUSIONS

In this paper, we argued that the user intents covered by a query can be hierarchical. We leveraged hierarchical intents and proposed hierarchical diversification models to promote search result diversification. Specifically, hierarchical diversification calculated the document diversity on each level of hierarchical subtopics and combined these diversity scores together to help select the best document. We conducted our experiments with two-level hierarchical subtopics generated automatically from Google suggestions. The experimental results showed that our approaches based on hierarchical subtopics outperformed their counterparts with the traditional subtopics in a flat list. Even when using single-level subtopics, hierarchical diversification also provided reasonable benefits, as these single-level subtopics from hierarchical tree implicitly utilize hierarchical information to help diversify search results, while the traditional diversification algorithms use pure subtopics in a flat list without additional information.

7. ACKNOWLEDGMENT

This work was partially supported by the National Key Basic Research 973 Program of China under grant No. 2014CB340403 and the Fundamental Research Funds for the Central Universities, the Research Funds of Renmin University of China No. 14XNLF05 and No. 15XNLF03.

8. REFERENCES

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

[2] J. Carbonell and J. Goldstein. The use of MMR, diversity-based reranking for reordering documents and producing summaries. In *SIGIR*, 1998.

[3] B. Carterette and P. Chandar. Probabilistic models of ranking novel documents for faceted topic retrieval. In *CIKM*, pages 1287–1296, Hong Kong, China, 2009.

[4] O. Chapelle, D. Metlzer, Y. Zhang, and P. Grinspan. Expected reciprocal rank for graded relevance. In *CIKM*, pages 621–630, Hong Kong, China, 2009.

[5] H. Chen and D. R. Karger. Less is more: Probabilistic models for retrieving fewer relevant documents. In *SIGIR*, pages 429–436, Seattle, Washington, 2006.

[6] C. L. Clarke, N. Craswell, and I. Soboroff. Overview of the TREC 2009 web track. In *Proc. the 18th TREC*, 2009.

[7] C. L. Clarke, N. Craswell, and E. M. Voorhees. Overview of the TREC 2012 web track. In *Proc. the 21st TREC*, 2012.

[8] C. L. Clarke, M. Kolla, G. V. Cormack, O. Vechtomova, A. Ashkan, S. Büttcher, and I. MacKinnon. Novelty and diversity in information retrieval evaluation. In *SIGIR*, pages 659–666, 2008.

[9] C. L. Clarke, M. Kolla, and O. Vechtomova. An effectiveness measure for ambiguous and underspecified queries. In *ICTIR*, 2009.

[10] The ClueWeb09 dataset. `http://boston.lti.cs.cmu.edu/Data/clueweb09/`.

[11] K. Collins-Thompson, P. Bennett, F. Diaz, C. L. A. Clarke, and E. M. Voorhees. TREC 2013 Web track overview. In *TREC*, 2013.

[12] V. Dang and W. B. Croft. Diversity by proportionality: An election-based approach to search result diversification. In *SIGIR*, pages 65–74, 2012.

[13] V. Dang and W. B. Croft. Term level search result diversification. In *SIGIR*, pages 603–612, 2013.

[14] Z. Dou, S. Hu, K. Chen, R. Song, and J.-R. Wen. Multi-dimensional search result diversification. In *WSDM*, pages 475–484, Hong Kong, China, 2011.

[15] Z. Dou, S. Hu, Y. Luo, R. Song, and J.-R. Wen. Finding dimensions for queries. In *CIKM*, 2011.

[16] Z. Dou, R. Song, and J.-R. Wen. A large-scale evaluation and analysis of personalized search strategies. In *WWW*, pages 581–590, 2007.

[17] J. He, V. Hollink, and A. de Vries. Combining implicit and explicit topic representations for result diversification. In *SIGIR*, pages 851–860, 2012.

[18] B. J. Jansen, A. Spink, and T. Saracevic. Real life, real users, and real needs: a study and analysis of user queries on the web. *Inf. Process. Manage.*, 36(2):207–227, 2000.

[19] W. Kong and J. Allan. Extracting query facets from search results. In *SIGIR*, pages 93–102, 2013.

[20] D. Lawrie, W. B. Croft, and A. Rosenberg. Finding topic words for hierarchical summarization. In *SIGIR*, pages 349–357, New Orleans, Louisiana, USA, 2001.

[21] S. Liang, Z. Ren, and M. de Rijke. Fusion helps diversification. In *SIGIR*, pages 303–312, 2014.

[22] Y. Liu, R. Song, M. Zhang, Z. Dou, T. Yamamoto, M. Kato, H. Ohshima, and K. Zhou. Overview of the NTCIR-11 IMine task. In *NTCIR-11*, 2014.

[23] F. Radlinski and S. Dumais. Improving personalized web search using result diversification. In *SIGIR*, 2006.

[24] F. Radlinski, R. Kleinberg, and T. Joachims. Learning diverse rankings with multi-armed bandits. In *ICML*, 2008.

[25] D. Rafiei, K. Bharat, and A. Shukla. Diversifying web search results. In *WWW '10*, pages 781–790, 2010.

[26] K. Raman, P. N. Bennett, and K. Collins-Thompson. Toward whole-session relevance: Exploring intrinsic diversity in web search. In *SIGIR*, 2013.

[27] T. Sakai, Z. Dou, T. Yamamoto, Y. Liu, M. Zhang, and R. Song. Overview of the NTCIR-10 INTENT-2 task. In *NTCIR-10*, 2013.

[28] T. Sakai and R. Song. Evaluating diversified search results using per-intent graded relevance. In *SIGIR*, pages 1043–1052, Beijing, China, 2011.

[29] M. Sanderson and B. Croft. Deriving concept hierarchies from text. In *Research and Development in Information Retrieval*, 1999.

[30] R. L. Santos, C. Macdonald, and I. Ounis. Exploiting query reformulations for web search result diversification. In *WWW*, pages 881–890, 2010.

[31] R. L. Santos, C. Macdonald, and I. Ounis. Selectively diversifying web search results. In *CIKM*, 2010.

[32] R. L. Santos, C. Macdonald, and I. Ounis. Intent-aware search result diversification. In *SIGIR*, pages 595–604, 2011.

[33] C. Silverstein, H. Marais, M. Henzinger, and M. Moricz. Analysis of a very large web search engine query log. *SIGIR Forum*, 33(1):6–12, 1999.

[34] R. Song, Z. Luo, J.-Y. Nie, Y. Yu, and H.-W. Hon. Identification of ambiguous queries in web search. *IPM*, 45(2), 2009.

[35] S. Vargas, P. Castells, and D. Vallet. Explicit relevance models in intent-oriented information retrieval diversification. In *SIGIR*, pages 75–84, 2012.

[36] H.-T. Yu and F. Ren. Search result diversification via filling up multiple knapsacks. In *CIKM*, pages 609–618, 2014.

[37] Y. Yue and T. Joachims. Predicting diverse subsets using structural svms. In *ICML*, 2008.

[38] C. Zhai and J. Lafferty. A risk minimization framework for information retrieval. *IPM*, 42(1):31–55, 2006.

[39] C. X. Zhai, W. W. Cohen, and J. Lafferty. Beyond independent relevance: methods and evaluation metrics for subtopic retrieval. In *SIGIR*, 2003.

[40] B. Zhang, H. Li, Y. Liu, L. Ji, W. Xi, W. Fan, Z. Chen, and W.-Y. Ma. Improving web search results using affinity graph. In *SIGIR*, pages 504–511, 2005.

[41] W. Zheng, H. Fang, and C. Yao. Exploiting concept hierarchy for result diversification. In *CIKM*, 2012.

[42] X. Zhu, A. Goldberg, J. Van Gael, and D. Andrzejewski. Improving diversity in ranking using absorbing random walks. *HLT-NAACL*, 2007.

[43] Y. Zhu, Y. Lan, J. Guo, X. Cheng, and S. Niu. Learning for search result diversification. In *SIGIR*, pages 293–302, 2014.