Inf Retrieval (2013) 16:484–503
DOI 10.1007/s10791-013-9221-8

SEARCH INTENTS AND DIVERSIFICATION

Mining subtopics from text fragments for a web query

Qinglei Wang · Yanan Qian · Ruihua Song · Zhicheng Dou · Fan Zhang · Tetsuya Sakai · Qinghua Zheng

Received: 22 May 2012 / Accepted: 11 February 2013 / Published online: 27 February 2013

Abstract Web search queries are often ambiguous or faceted, and the task of identifying the major underlying senses and facets of queries has received much attention in recent years. We refer to this task as query subtopic mining. In this paper, we propose to use surrounding text of query terms in top retrieved documents to mine subtopics and rank them. We first extract text fragments containing query terms from different parts of documents. Then we group similar text fragments into clusters and generate a readable subtopic for each cluster. Based on the cluster and the language model trained from a query log, we calculate three features and combine them into a relevance score for each subtopic. Subtopics are finally ranked by balancing relevance and novelty. Our evaluation experiments with the NTCIR-9 INTENT Chinese Subtopic Mining test collection show that our method significantly outperforms a query log based method proposed by Radlinski et al. (2010) and a search result clustering based method proposed by Zeng et al. (2004) in terms

Q. Wang (✉) · Y. Qian · Q. Zheng
SPKLSTN Lab, Department of Computer Science and Technology,
Xi'an Jiaotong University, Xi'an 710049, People's Republic of China
e-mail: wangqinglei0116@gmail.com

Y. Qian
e-mail: yanan.qian@stu.xjtu.edu.cn

Q. Zheng
e-mail: qhzheng@mail.xjtu.edu.cn

R. Song · Z. Dou · T. Sakai
Microsoft Research Asia, No. 5 Danling Street, Haidian District, Beijing 100080,
People's Republic of China
e-mail: Song.Ruihua@microsoft.com

Z. Dou
e-mail: zhichdou@microsoft.com

T. Sakai
e-mail: tetsuyasakai@acm.org

F. Zhang
Nankai-Baidu Joint Lab, Nankai University, Tianjin 300071, People's Republic of China
e-mail: zhangfan555@gmail.com

of precision, I-rec, D-nDCG and D#-nDCG, the official evaluation metrics used at the NTCIR-9 INTENT task. Moreover, our generated subtopics are significantly more *readable* than those generated by the search result clustering method.

keywords Query intent · Intent mining · Intents ranking

1 Introduction

Understanding the intent behind a query has long been recognized as an essential part of an information retrieval system. With the widespread use of web search engines, this research problem is as important but challenging as ever, as web search queries are often *ambiguous* or *faceted* (Clarke et al. 2009), mostly because they tend to be very short.

Figure 1 shows a query length distribution based on *SogouQ*, a Chinese query log released at the NTCIR-9 INTENT task (Song et al. 2011). This shows that 80 % of queries by volume are composed of no more than four Chinese characters (which is roughly equivalent to two English words). This is what makes the task of finding possible *senses* for ambiguous queries and *facets* for underspecified queries so important. For example, "Farewell My Concubine" is an ambiguous query. It may refer to a well-known movie, a Chinese cuisine, or a computer game. "Mozart" is a faceted query which may cover various information needs such as biography of Mozart, music by Mozart and movies about Mozart. Note that a query can be ambiguous *and* faceted at the same time.

This paper addresses the problem of *query subtopic mining*, which we define as: given a query, list up its possible *subtopics* (i.e. senses or facets) as exhaustively as possible, and rank them by importance. This is the same task setting as the NTCIR-9 INTENT Subtopic Mining subtask (Song et al. 2011). Applications of query subtopic mining include query suggestion and *search result diversification*, which aims to satisfy diverse user intents with a single search result page (Rafiei et al. 2010; Zheng and Fang 2011; Santos et al. 2010a).

There are two major existing approaches to tackling the problem of query subtopic mining. The first class of approaches leverage *query log*, including click-throughs and sessions, to mine subtopics. The mined subtopics are real queries and reflect users' interests. User behaviors are also useful in estimating the popularity of a subtopic. However, the drawback is that these kinds of approaches work reliably only for the head queries with rich log data. They cannot well handle either the tail queries that lack log data, or new queries that have never been seen yet. The second class mines subtopics based on *search*

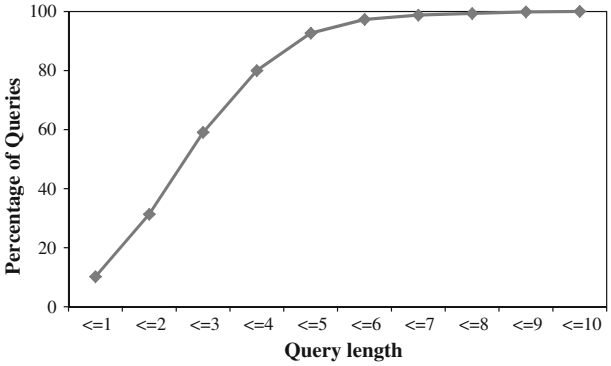

Fig. 1 Query length distribution of SogouQ

result clustering. Usually these kinds of approaches cluster top retrieved documents based on corresponding titles and snippets returned on search result pages. Then they extract salient keywords from each cluster. Assuming that each cluster represents an aspect of a query, we can regard the extracted keywords together with the query as a subtopic. However, such keywords tend to be poor as description of the query's aspects in terms of readability. Readability is important for some applications, such as query suggestion and search result diversification based on explicit subtopics.

We propose a new approach, which is free from the limitations of the aforementioned existing approaches, to address the problem. Our idea is to utilize the *context* surrounding query terms, which we call *text fragments*, from top retrieved documents. For example, for the query "Mozart", our method can extract text fragments such as "Mozart's opera: Magic Flute", "Mozart Serenade", "Mozart effect—the more the cleverer" and "Mozart listen and download." It is possible to generate good descriptions of different aspects of the query based on those text fragments.

Given a query, we retrieve hundreds of Web documents from a corpus and then generate subtopics as follows: First, we extract text fragments from different parts of the top retrieved documents. Second, we cluster the extracted fragments based on text similarity, so that each cluster can be regarded as a potential intent. Third, we propose to generate a subtopic by finding a minimal span that covers query terms and a core phrase for each cluster. Such a subtopic is generally readable and summarizes one aspect of the query. Finally, the relevance of each subtopic is estimated according to a set of features, and the subtopics are ranked by taking into account of the relevance and the redundancy to subtopics ranked higher.

Our evaluation experiments with the NTCIR-9 INTENT Chinese Subtopic Mining test collection shows that our method is effective in discovering different aspects of a query, generating readable subtopics and ranking them. The results indicate that our method significantly outperforms a query log based method proposed by Radlinski et al. (2010) and a search result clustering based method proposed by Zeng et al. (2004) in terms of precision, I-rec, D-nDCG and D#-nDCG, the official evaluation metrics used at the NTCIR-9 INTENT task. Moreover, our generated subtopics are significantly more *readable* than the method based on search result clustering.

The rest of the paper is organized as follows. Section 2 discusses prior art related to our work. Section 3 proposes our subtopic mining method, and Sect. 4 reports our experimental results. Finally, Sect. 5 concludes this paper.

2 Related work

It has long been recognized that queries often have multiple senses or facets, and that relevance alone cannot provide satisfactory information retrieval (Goffman 1964). Hence researchers have tackled various problems related to handling intents behind the query. Below, we discuss three classes of existing studies that are relevant to the problem of subtopic mining.

2.1 Query log mining

Several existing methods leverage query logs, including clickthroughs and sessions, to mine search intents or subtopics. Beeferman and Berger (2000) proposed a clustering method based on a query-URL bipartite graph, where the query clusters thus obtained can

be regarded as different intents or subtopics. Strohmaier et al. (2009) first obtained similar queries from search sessions, filtered out noisy queries using clickthrough data, and then grouped the remaining queries based on random walk similarity. They also estimated the popularity of each intent based on the number of observations in the query logs. Li et al. (2008) proposed a clickthrough based method for classifying queries into predefined classes. Radlinski et al. (2010) mined query suggestions by using both session logs and Web pages. Wang and Zhai (2007) first used search session logs to find similar queries, and then performed star clustering on related clicked documents. Finally, a user query is selected as the description of each cluster.

Query log based approaches suffer the data sparsity issue and many queries have no enough log to mine subtopics. In contrast, our method is more general and able to mine subtopics for any query if some relevant documents can be retrieved.

2.2 Search result clustering

A search result page contains titles and snippets of retrieved documents, which may describe various aspects of a given query. Search result clustering uses such kind of valuable information and cluster documents. It shows a user the clusters so that the user can choose a particular cluster and then dig into it. There is a lot of existing work on this topic (Cutting et al. 1993; Hearst and Pedersen 1996; Leouski 2005; Leuski and Allan 2000). Zamir and Etzioni (1998, 1999) proposed to group web search results using suffix tree clustering. Hearst and Pedersen (1996) and Hearst et al. (1995) applied the scatter/gather techniques to let the user organize and browse documents interactively. Chen and Dumais (2000) used a set of predefined categories and applied text classification to search results rather than clustering. Zeng et al. (2004) first extracted salient phrases from search result snippets, and then group the search results by means of ranking the salient phrases. Vivisimo (Koshman et al. 2006) provided a Web service using a similar efficient technique. Wang and Zhai (2007) describes an extension of this approach that utilizes not only the search results of the original query but also those of similar queries.

There are also some existing studies on *snippet* clustering. Ferragina and Gulli (2008) clustered snippets hierarchically to organize Web pages according to the themes. Geraci et al. (2006) augmented the furthest-point-first algorithm for k-center clustering Osinski and Weiss (2005) proposed the Lingo algorithm which combines common phrase discovery and latent semantic indexing to separate search results into meaningful groups. Sahami and Heilman (2006) proposed a similarity kernel function to measure the similarity between short text snippets by leveraging web search results.

Our work also applies clustering techniques but we cluster text fragments from anchor, title and body of retrieved HTML documents. Our work is unique in generating readable subtopics from clusters, rather than selecting several salient terms to present a cluster. Experimental results show our generated subtopics are better than those in Zeng et al. (2004) by 50 % in terms of readability.

2.3 Search result diversification

To better fulfill information needs behind ambiguous or underspecified queries, researchers have tried to diversify search results, so that a single search result page can serve as an entry point for different user intents (Gollapudi and Sharma 2009; Rafiei et al. 2010; Santos et al. 2010b; Clough et al. 2009). Recently, Zheng and Fang (2011) compared different diversification strategies, such as Maximal Marginal Relevance (Carbonell and

Goldstein 1998), which balances the relevance and the redundancy among the returned documents, and eXplicit Query Aspect Diversification (xQuAD) (Santos et al. 2010a) which uses an explicit list of query subtopics for diversification. Their results showed that explicit modeling of intents outperforms implicit modeling. We therefore discuss some explicit modeling approaches below.

Agrawal et al. (2009) proposed a search result diversification algorithm, which requires a pre-defined taxonomy of intents. Chandar and Carterette (2010) modeled search result documents as a hyperlinked graph, and then treated dense clusters on the graphs as different subtopics. Santos et al. (2010a) proposed the aforementioned xQuAD framework, in which they harvest query suggestions from three commercial search engines to form query subtopics. They also used web search hit counts to estimate subtopic popularity. Zheng and Fang (2010) discovered subtopics from search result pages, where each subtopic is a set of terms that frequently co-occurred with the query. They ranked the candidate subtopics based on statistics such as TF-IDF and the number of co-occurrences. Zheng et al. (2011) extended this approach, by utilizing the structured ODP data.

Search result diversification is an application of our mined subtopics. Our work can generate subtopics for search result diversification based explicit modeling intents. In our experiments, we aim to rank subtopics, instead of documents, but we also diversify subtopics by applying the MMR method proposed by Carbonell and Goldstein (1998).

3 Our approach

In this section, we propose an approach to address the problem of subtopic mining problem. Given a query, our approach takes top retrieved documents as input and outputs a list of ranked subtopics.

3.1 System overview

We observe that the surrounding text of query terms in relevant documents tend to differentiate various aspects of a given query. And it is possible to discover good phrases from the surrounding text as description of the aspects. Together with the query, such kind of phrase can compose a subtopic of high quality. Therefore, we propose mining subtopics from text fragments of top retrieved documents. Figure 2 provides the flowchart of our proposed approach. We use the Chinese query "farewell my concubine" a an running example. Short translations in English are provided in italic.

As Fig. 2 shows, given a query q, we first retrieve documents and fetch the full documents to form a document set R. We take R as input and then go through the following four modules to generate a list of ordered subtopics:

1. **Fragment extraction** For each document $d \in R$, we divide it into four fields: title, link text, bold text and other plain text. The last three are from the body of d. In our paper, a fragment is defined as a sentence or text in a HTML node that contain all key terms in q .We extract all fragments from the four fields and thus each fragment f is of a type, i.e. title or link text or bold text or plain text. For our example query "farewell my concubine," we show some fragments extracted from d_1 and d_2 in the text box entitled *Fragment Extraction*. Some fragments are bullets and some are sentences. Each of them contain all query terms. Some short fragments are close to be good subtopics, such as the third fragment in d_1: "Movie Farewell My Concubine".

Fig. 2 The flowchart of our approach (example query: "farewell my concubine")

2. **Fragment clustering** Some fragments are similar to each other and talk about the same aspect of the query. We cluster all extracted fragments based on terms and thus we collapse similar subtopics of q. For example, in the text box entitled *fragment clustering*, some fragments about the opera performed by Mr. Mei, who is a master of Beijing opera, are grouped into the cluster C_1; and some fragments about the movie are in the cluster C_2. In general, major intents or subtopics correspond to larger cluster and have more supported documents.
3. **Subtopic generation** For each cluster, we generate a short but meaningful descriptive string on which aspect of q the cluster talks about. We regard it as a subtopic. For our example query, in the text box entitled *subtopic generation*, we show five subtopics generated for the five clusters $C_1, C_2, \ldots, C_5$. They are "Mei Lanfang's work farewell my concubine," "movie farewell my concubine," "Leslie Cheung farewell my concubine," "song farewell my concubine" and "review of the movie farewell my concubine." Our mined subtopics look like suggested queries but they are automatically generated from text fragments.
4. **Subtopic ranking** We combine three features to measure relevance of mined subtopics and balance relevance and diversity in ranking. Our goal is (1) major intents are ranked higher and (2) top subtopics can cover as many different intents as possible. For our example, the subtopic about the movie is ranked the highest, followed by those on "Mei Lanfang's work," "Leslie Cheung," "download" and so on, as shown in the

text box entitled *subtopic ranking*. According to our estimated relevance score, the movie is a more wanted subtopic than the opera performed by Mr. Mei.

Next, we give more details about the four modules.

3.2 Fragment extraction

From each document d in the top retrieved document set R, we extract fragments that are pieces of text separated from other text by some punctuations or line breaks or HTML nodes and contain all query terms in q. In this paper, we have four types of fragments:

- Title fragment: a fragment in the title of d;
- Link text fragment: a fragment in link text in HTML body of d;
- Bold text fragment: a fragment in inner text of bold family HTML tags, such as $<B>$ and $<H1>$, in HTML body;
- Plain text fragment: a fragment in other text of HTML body.

The first three types of fragments are often short and readable summary of a query's aspect and thus more important in generating and ranking subtopics. For example, for the query "farewell my concubine," we can obtain "Mei Lanfang's work farewell my concubine" as a bold text fragment, which is exactly a good subtopic.

Next, we remove the fragments that are exactly the query q as they are useless to mine subtopics. We also collapse duplicated fragments from a document.

After removing stop words and query terms, we represent each fragment by a term vector as follows:

$$\overrightarrow{f} = (w_{1,f}, w_{2,f}, \ldots, w_{n,f}) \tag{1}$$

where $w_{i,f}$ is the weight of term i contained in the fragment f, given by:

$$w_{i,f} = tf_{i,f} \cdot weight_i. \tag{2}$$

Here, $tf_{i,f}$ is the term frequency of i in fragment f, and $weight_i$ is the IDF-like weight of i and calculated by the formula proposed by Robertson and Jones (1976):

$$weight_i = \log \frac{N - n_i + 0.5}{n_i + 0.5} \tag{3}$$

where N is the total number of documents in the NTCIR-9 INTENT Chinese document collection and n_i the number of documents within the collection that contain i.

3.3 Fragment clustering

We cluster extracted fragments so that similar fragments can gather to form an aspect of a query. Furthermore, we can measure how a subtopic is relevant and important to the query based on clusters. For example, the subtopic describing major intent usually corresponds to the largest cluster; whereas, some clusters with only one or two fragments, which are often noisy fragments such as "farewell my concubine > texts", can be identified and filtered.

A challenge of fragment clustering is that many fragments are so short that the fragments talking about the same aspect of a query do not necessarily share common terms (we have removed query terms when representing a fragment in vector space model.) To handle this issue, we needs a clustering algorithm that accommodates the "chaining effect" in

similarity computation. The single-link hierarchical agglomerative clustering (HAC) holds such a property, but its complexity is $O(n^2)$ (Sibson 1973). To ensure the quality of subtopics, we keep only a set of clusters, in which the maximum similarity between any two member fragments is larger than a similarity threshold θ. Given such criteria, we can modify single-link HAC as shown in Algorithm 1 to reduce complexity. The best complexity of Algorithm 1 is $O(n)$ when all fragments belong to one cluster; the worst is $O(n^2)$ when each fragment belong to a single cluster.

Based on the fragment vectors described in Sect. 3.2, we measure the similarity between two fragments by cosine function:

$$Sim(\overrightarrow{f_i}, \overrightarrow{f_j}) = \frac{\overrightarrow{f_i} \cdot \overrightarrow{f_j}}{|\overrightarrow{f_i}||\overrightarrow{f_j}|} \tag{4}$$

Algorithm 1 Fragment Clustering Algorithm

Input: a set of fragments $F = \{f_1, f_2, \cdots, f_n\}$, each f_i is represented by a vector $\overrightarrow{f_i} = (w_{1,f_i}, w_{2,f_i}, \ldots, w_{m,f_i})$
Output: a set of fragment clusters $\mathcal{C}$, initiated as $\mathcal{C} = \emptyset$

```
Initiate two queues: the unprocessed queue: U = {f1, f2, ..., fn}, note the fragments in F can be randomly selected to form U; The processed queue: Q = ∅
Pop one fi from U, push it in Q
i=0
while i < |Q| do
    for j = 0; j < len(U); j++ do
        if Sim(Q[i], U[j]) > θ then
            Delete U[j] from U
            Push U[j] in Q
        end if
    end for
    i++
end while
C = C ⋃ {Q}, Q = ∅
if U ≠ ∅ then
    Go to step 2
end if
```

3.4 Subtopic generation

Based on the assumption that each cluster of fragments represents an aspect of a query, we further generate a subtopic from each cluster.

More formally, we define the subtopic generation problem as follows: given a cluster C of fragments where $C = \{f_1, f_2, \cdots, f_n\}$, find a substring $s \subseteq f_i$ that represents the subtopic corresponding to C and meets the following requirements: (1) express the main topic of the cluster; (2) be human-readable; and (3) be short like a suggestion to the query.

The problem is similar to the frequent subsequence mining problem (Ji and Bailey 2007), but the differences are: (1) we only need to mine the most frequent subsequence; (2) the subsequence would be a good summary of the cluster of fragments; and (3) the subsequence would be relevant to the original query.

To solve this problem, we propose a linear-time algorithm that is composed of three steps: *finding a core term*, *expanding to a core phrase*, and *generating a subtopic*.

1. **Finding a core term** Intuitively, the most frequent term in a cluster is a good candidate as a core term that represents the main topic of C; however, if the term also frequently occurs in other clusters in $\mathcal{C}$, which is the set of clusters for the query q, the term is less useful to distinguish this cluster's main topic from others. Thus we use a TF-IDF like formula to measure how relevant a term t in C is to the cluster's main topic:

$$tf_C(t)\frac{1}{\sum_{C_i\in\mathcal{C}} tf_{C_i}(t)+1} \tag{5}$$

   where $tf_C(t)$ is the frequency of term t in cluster C.
   We refer to the term getting the highest score as the *core term*, which we denote by t*.
2. **Expanding to a core phrase** Although t^* is most likely to be the important part of main topic of C, sometimes only the term is not complete and thus cannot be understood. Therefore, we propose to *expand* the core term to an n-gram phrase that has a complete semantic meaning as follows: first, we add the term that lies to the *right* of t^* to obtain t^*t_1, and test whether its co-occurrence probability $\frac{tf_C(t^*t_1)}{tf_C(t^*)}$ is greater than a threshold τ, If this condition is satisfied, we further expand the string to the *left* to obtain $t_2t^*t_1$, and again test whether $\frac{tf_C(t_2t^*t_1)}{tf_C(t^*t_1)} > \tau^2$. By repeating this process until no further expansion is possible, we obtain $t_{n-1}\cdots t_4t_2t^*t_1t_3\cdots t_{n-2}$ which satisfies

$$\frac{tf_C(t_{n-1}\cdots t_4t_2t^*t_1t_3\cdots t_{n-1})}{tf_C(t^*)} > \tau^{n-1}. \tag{6}$$

   We refer to the expanded string as the *core phrase*, which we denoted by p^*.
3. **Generating a subtopic** Having obtained the n-gram that is expanded from the core term, the final step is to compose a subtopic by connecting the query q with the core phrase p^*. In this way, we can generate a readable subtopic that describe an aspect of the query, like a suggestion.

We first extract a minimal span or substring that covers both query terms and the n-gram from each fragment. The fragment is called supporting fragment to a span if the span is extracted from this fragment. Then we count the number of supporting fragments for each extracted span as span frequency. Spans with the highest frequency are subtopic candidates. Finally, we select the shortest candidate as the subtopic for cluster C.

3.5 Subtopic ranking

The above steps can generate subtopics for discovered clusters. In this section, we rank the subtopics to optimize both relevance and diversity, which is the goal of Subtopic Mining subtask in NTCIR-9 INTENT.

3.5.1 Relevance score of a subtopic

We estimate a relevance score *Rel*(.) for a subtopic by considering the following aspects: (1) the relevance of subtopics to the given query; (2) the importance of subtopics, which partially reflects popularity; and (3) the readability of mined subtopics.

Accordingly we propose some features and linearly combine them into a relevance score.

1. **Document ranking score (DR)** We measure both relevance and importance of a subtopic s from a cluster C in terms of the rank of supporting documents and the type of fragments and calculate a Document ranking score as:

$$DR(s) = \sum_{d_i \cap C \neq \emptyset} \frac{\max_{f_j \in (d_i \cap C)} typeweight(f_j)}{\sqrt{rank(d_i)}} \tag{7}$$

   Here, a document d_i supports C iff. any fragment in C is extracted from d_i. $rank(d_i)$ is the rank of d_i in the list returned by a document retrieval system. We convert the rank into a normalized relevance score and sum the scores from all supporting documents. Intuitively, a subtopic s is more relevant to the query q if more relevant documents support it.
   In addition, the maximum type weight of fragments in a document is also taken into account because we find fragments of different types are not equally important to contribute good subtopics. Fragments from link text are more important than those from title or bold text. Fragments from plain text are the least important. Thus we empirically set weights for different types as shown in Table 1.

2. **Inverted average length (IAL)** We find that some subtopics from shorter fragments are major intents of a query. It is because shorter fragments are often titles, subtitles, or link text. They well describe aspects of the query because they are manually created. In addition, the subtopics that are generated from these fragments tend to be shorter and more like query suggestions. Therefore, we use the reciprocal of average length of fragments in a cluster as one feature and called it inverted average length (IAL):

$$IAL(s) = \frac{|C|}{\sum_{f_i \in C} |f_i|}. \tag{8}$$

3. **Generation probability (GP)** Again a good subtopic is supposed to be like real Web queries. To measure how good the generated subtopic is, we build a language model from query log and estimate the probability of generating a subtopic s given the language model. We call such a feature generation probability.
   In this paper, we learn the query language model from SogouQ (Song et al. 2011) and apply good-turning frequency estimation for smoothing (Gale and Sampson 1995). For a subtopic $s = w_1 w_2 \cdots w_n$, we compute $\mathrm{P}(\mathrm{w_1 w_2 \cdots w_n})$, the generation probability from query language model, as:

$$GP(s) = \mathrm{P}(\mathrm{w_1 w_2 \cdots w_n}) = \prod_{\mathrm{i}=1}^{\mathrm{n}} \mathrm{P}(\mathrm{w_i} | \mathrm{w_{i-3} w_{i-2} w_{i-1}})). \tag{9}$$

As the average query length of the SogouQ dataset is about four Chinese characters, we apply the 4-gram query language model.

Table 1 Weights of different types of fragment

Fragment type	Weight	Fragment type	Weight
Link text	1.0	Title	0.75
Bold text	0.75	Plain text	0.5

Finally, we linearly combine the above three features and get a relevance score for a subtopic s:

$$Rel(s) = a_1 DR(s) + a_2 IAL(s) + a_3 GP(s) \tag{10}$$

where a_i are coefficients and tuned on training data as discussed in Sect. 4.1.4

3.5.2 Subtopic diversification

We utilize the maximum marginal relevance (MMR) framework Carbonell and Goldstein (1998) to further evaluate the diversity of mined subtopics. The MMR model regards the ranking problem as a procedure of successively selecting the "best" unranked object (usually a document but in our scenario, a subtopic) and arranging it at the tail of the rank list. When looking for the next best subtopic, the MMR model chooses not the most relevant one but the one that best balances the relevance and novelty. Novelty means that a subtopic is new compared to those already chosen and ranked. Therefore, the MMR model can further improve the ranking of subtopics in diversity, which is one of the main goals of the NTCIR-9 INTENT task.

Given a relevance function $Rel(.)$ and a similarity function $Sim(., .)$, the MMR model could be set up as follows:

$$s_{i+1} = \text{argmax}_{s \notin S_i} \{\alpha \text{Rel(s)} + (1 - \alpha)\text{Nov(s, S}_i)\}$$

where $\alpha \in [0, 1]$ and it is a combining parameter. Then

$$S_{i+1} = S_i \cup s_{i+1}$$

Here, s_i is the subtopic ranked at the ith position and S_i is the collection containing the top i subtopics. The function $Nov(s, S_i)$ tries to measure the novelty of s given S_i has already been chosen and ranked. In our approach, we implement the novelty function as follows:

$$Nov(s, S_i) = -\max s' \in S_i Sim(s, s')$$

We apply Jaccard Similarity between two sets to calculate the similarity between s and s'. First, a subtopic can be represented by a set of terms. Then we apply Jaccard Similarity to calculate the similarity between s and s':

$$Sim(s, s') = \frac{s \cap s'}{s \cup s'}$$

Finally, we find the maximum among the similarity values between s and all s' in S_i and take its opposite number as the novelty score. The more similar is s to some previous selected query, the less novel it is.

4 Experiments

In this section, we evaluate our proposed methods with different settings using the NTCIR-9 INTENT Chinese Subtopic Mining test collection (Song et al. 2011) and compare our method to some previous work.

4.1 Experiment setup

4.1.1 Data

The NTCIR-9 INTENT Chinese Subtopic Mining test collection (Song et al. 2011) consists of the following:

- A set of 100 Chinese topics selected from "torso" queries in a real query log;
- A set of possible intents for each topic, obtained by pooling and manually clustering the subtopics submitted to the Chinese Subtopic Mining task;
- An estimated probability of each intent, obtained through assessor voting;
- A set of subtopics for each topic, obtained by pooling the submitted runs and mapped to one relevant intent ID (>0) or 0 if it is irrelevant to the topic.

Figure 3 shows an example of a topic and its intents from the INTENT Chinese Subtopic Mining test collection. The topic (i.e. the original query) is "Mozart," and there are seven intents for this topic, each with its estimated probability. The DESCRIPTION fields represent the intent labels manually assigned by assessors who clustered the submitted subtopics, and the EXAMPLES fields show some of the raw subtopics belonging to a particular intent.

Fig. 3 An example of labeled intents, with rough English translations for each intent

The INTENT task also provides a Chinese document collection called SogouT, which consists of approximately 138 million web pages crawled in 2008. This is used as the corpus to retrieve relevant documents. In addition, the task provides a Chinese query log called SogouQ, which we utilize for training the query language model as described in Sect. 3.5, and also for one of our baselines, i.e. the log based method proposed in (Radlinski et al. 2010). For more details on the INTENT data, we refer the reader to Song et al. (2011).

4.1.2 Evaluation metrics

In this paper, we adopted precision, I-rec (Zhai et al. 2003), D-nDCG and D#-nDCG (Sakai and Song 2011) to evaluate the mined and ranked subtopics; those metrics are used as official by the NTCIR-9 INTENT Subtopic Mining task. Precision measures the percentage of a query's subtopics being relevant. I-rec measures the diversity of the returned subtopics; it shows how many percentages of intents can be found. D-nDCG measures the overall relevance across all intents considering the subtopic ranking. D#-nDCG is a combination of I-rec and D-nDCG. It is used as the primary evaluation metric by the INTENT task organisers. The advantages of D#-nDCG over other diversity metrics such as α-nDCG (Clarke et al. 2011) and Intent-Aware metrics (Agrawal et al. 2009) are discussed in Sakai and Song (2011). In our experiments, we use NTCIREVAL (Sakai 2011), the tool provided by the NTCIR-9 organisers, to compute the above four metrics, in which the default setting is used, i.e. D#-nDCG is a simple average of I-rec and D-nDCG.

The gold standard data in the NTCIR-9 test collection contain of a set of possible intents for each topic, where each intent is associated with a set of subtopics pooled from the NTCIR-9 participants' runs. However, these pooled subtopics often do not match with the subtopics output by our system. Hence, to evaluate our system with the NTCIR-9 test collection, we hired two college students to independently map our subtopics to the gold-standard intents. Whenever the two assessors disagreed, they were required to discuss and reach and agreement. If a new subtopic was relevant to the topic but did not match any of the gold-standard intents, it was classified as "others."

4.1.3 Document retrieval platform

We index all documents in SogouT using WebStudio. WebStudio[1] is an experimental search system developed by Microsoft Research for facilitating large-scale, end-to-end search experiments. We deploy the platform on 20 servers, each of which has 4 CPUs, 16 GB memory, and four 1T IDE disks. Each server indexes about 6.9 M Web pages. When users submit a query, the query request will be sent to all servers via an aggregator. Each server processes the query and retrieves top results separately. The aggregator then aggregates all results and returns to end users.

We apply a ranking function named MSRA2000, which was one of top performers in TREC 2004 (Song et al. 2004). It combines augmented BM25 scores of four different Web page fields including title, body, URL and anchor, and it also considers the proximity between query term occurrences. To make the work reproducible, we have shared intermediate data (top 1,000 retrieved documents for each topic) to research community[2].

[1] http://research.microsoft.com/en-us/projects/WebStudio.

[2] http://labs.xjtudlc.com/labs/data.html.

Table 2 Evaluation of the proposed method

Cut-off	Precision	I-rec	D-nDCG	D#-nDCG
@10	**0.8230**	0.5006	**0.6912**	0.5959
@20	0.7750	0.6118	0.6506	**0.6312**
@30	0.6283	**0.6455**	0.5369	0.5913

4.1.4 Parameters tuning

Our method has a few parameters: the number of Web pages $|R|$ from which fragments are extracted, the clustering threshold θ for Algorithm 1, the parameter τ in finding a core phrase, the three coefficients in our ranking function and α in the MMR framework. We use the ten example topics released by the NTCIR-9 INTENT task organisers[3] as a tuning set and tune the parameters in terms of D#-nDCG. Finally, the optimal parameters are used in our experiments. $|R| = 200$, $\theta = 0.5$, $\tau = 0.8$, $a_1 = 0.415$, $a_2 = 0.166$, $a_3 = 0.385$, $\alpha = 0.8$. In addition, we discuss the effect of $|R|$ in Sect. 4.2.4.

4.1.5 Baselines

As discussed in Sect. 2, there are two major approaches that can be used to mine subtopics: one is based on query log and the other is based on search result clustering. We implement the query log based method proposed by Radlinski et al. (2010) as one baseline. The method has been used to form subtopics for TREC diversity task. We choose the search result clustering algorithm proposed by Zeng et al. (2004) as another baseline. We appreciate that the authors kindly shared their codes with us.

4.2 Experimental results

4.2.1 Evaluating our method

We evaluate our proposed method in terms of four metrics for three cutoffs and show results in Table 2. The results indicate that as the cutoff value increases, I-rec goes up, while precision and D-nDCG go down. It is natural because when more subtopics are retrieved, most likely more intents would be covered. That is why I-rec increases. On the contrary, the overall relevance would get worse if we measure more returned subtopics, because the subtopics ranked lower are less likely relevant. D#-nDCG is different. As it averages precision and I-rec, D#-nDCG@20 is the best as it balances precision and intent recall.

The evaluation result shows that our method is effective in mining and ranking subtopics. MSINT-S-C-2, one official run basically generated by our method, performs the second best among all submitted runs to the Chinese Subtopic Mining task in terms of D#-nDCG at the cutoff 10, and the difference between MSINT-S-C-2 and the best run is not statistically significant (Song et al. 2011).

[3] http://www.thuir.org/intent/ntcir9/ntcir9intent.topics.full.ec.xml.

Table 3 We compare our method with the log based baseline over the 69 topics that the baseline can return some subtopics

Cut-off@10	Precision	I-rec	D-nDCG	D#-nDCG
Log based baseline	0.7850	0.4702	0.6464	0.5583
Our method	**0.8970** †	**0.5365** †	**0.7279** †	**0.6322** †

We conduct Student's paired t test with a two-tailed distribution. † means that the difference over the log based baseline is significantly (p value <0.01)

4.2.2 Comparison with the log based baseline

We apply the log based method proposed by Radlinski et al. (2010) to mine subtopics from SogouQ for the 100 topics. However, due to the data sparsity issue of log, no any subtopic is returned for 31 topics. To be fair, we compare our method with the baseline on the remaining 69 topics. Results are shown in Table 3. The cutoff is 10.

The results show that our method significantly outperforms the query log based baseline in all metrics (by 12–14 %). It is easy to understand that our method is better in I-rec. Query log may cover only a few intents that users have queried about a topic; whereas, our method summarizes top 200 documents and thus cover more intents. This validates the advantage of methods based on top retrieved documents. It is somehow surprising that our method also wins in terms of precision. By looking closely into the results, we find that the log based method can only return two or three subtopics. When we calculate precision@10, the precision is no larger than 0.2 or 0.3. That indicates that the data sparsity issue also hurts the performance of log based methods in terms of precision at a cutoff.

4.2.3 Comparison with search result clustering

We apply the search result clustering (SRC) method proposed by Zeng et al. (2004) for the 100 topics. The generated cluster labels combined with the raw query are naturally regarded as subtopics, which are ranked in the same order as clusters. As the SRC method extracts fragments from titles and snippets in a search result page, we also run our method from titles and snippets, instead of full documents, to ensure fair comparison. We evaluated the SRC method, the title and snippet only version of our method and the full version of our method. Table 4 shows the results. Our method based on title and snippet only significantly outperforms the SRC method in all metrics (by 10–27%).

By case studies, we find that the readability of SRC generated subtopics is not as good as ours. Sometimes the most salient words selected cannot be easily understood and sometimes augmenting the topic by the salient words cannot form a subtopic. For example,

Table 4 We compare the baseline of SRC, the title and snippet only version of our method, and the full version of our method

Cut-off@10	Precision	I-rec	D-nDCG	D#-nDCG
SRC baseline	0.5750	0.4121	0.4335	0.4228
Our method (title+snippet)	0.6580†	0.4522†	0.5526†	0.5024†
Our method	**0.8230** ‡	**0.5006** ‡	**0.6912**‡	**0.5959** ‡

We conduct Student's paired t test with a two-tailed distribution. † means that the difference over the SRC baseline is significantly (p value <0.01) and ‡ means both the difference over the SRC baseline and the difference over the title and snippet only version of our method are significant (p value <0.01)

the SRC method returns "Mozart Beethoven 2008" and "Mozart Vienna Austria" for the topic "Mozart". Those cannot describe an aspect of the topic and thus they are not good subtopics. For the example query "Farewell My Concubine", the SRC method generates "Farewell My Concubine web design". We cannot understand which subtopic it refers to.

To measure readability, we hired two assessors, who are college students and did not participate in our project to independently provide a binary judgment. A subtopic is readable if: (1) The meaning of the subtopic is clearly understandable; and (2) The subtopic is acceptable as a new search query.

The assessors discuss for every inconsistent assessment until they reach an agreement. Based on the assessments, we calculate the percentage of readable subtopics in top ten results returned by the SRC method and our method. Results show that the percentage of readable subtopics generated by the SRC method is only 62.5 %; whereas, 90.4 % of subtopics generated by our method are readable. This verifies our observation. The most salient words in a cluster are not necessary good in readability. That is one main reason that our method performs better than the SRC method.

In addition, our method based on full documents is significantly better than both methods in all metrics. The improvements over our method based on title and snippet only are from 11 % (in I-rec) to 25 % (in precision and D-nDCG). This indicates that full documents can provide much more information than title and snippet for mining subtopics.

4.2.4 Effect of the number of retrieved documents

In this experiment, we change the number of documents that are retrieved and used to generate subtopics, i.e., $|R|$, and analyze its effect to subtopic mining and ranking. Figure 4 plots the performance of our method in terms of four metrics when $|R|$ changes. We observe that the four curves have the same trend. Our method's performance does not stop increasing until the number of documents goes up to about 200. This indicates that more documents can cover more intents. Performance saturates when $|R| > 200$. It indicates that 200 documents are enough to cover almost all intents. Additional documents cannot discover subtopics of new intents. When too many documents are used, e.g., $|R| = 1{,}000$, the performance even goes down a little bit because relevance of documents gets worse and may bring some noise in mining subtopics.

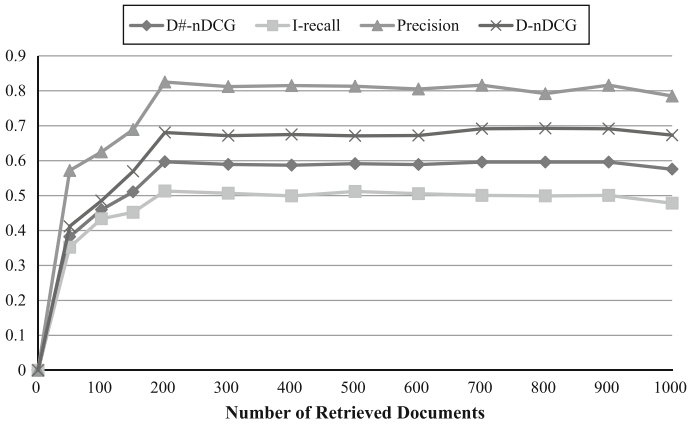

Fig. 4 The performance w.r.t. the number of search results

4.2.5 Evaluating individual ranking features

Our ranking function considers three individual features, namely document ranking score (DR), inverted average length (IAL) and generation probability (GP). In this experiment, we investigate contributions of individual features. We have the following four versions of our method:

- Combined: the full version of our method that linear combines three features in ranking.
- GP: the GP version of our method that uses only GP feature in ranking;
- DR: the DR version of our method that uses only DR feature in ranking;
- IAL+DR: the IAL and DR version of our method that uses IAL feature to rank subtopics, and uses DR feature as the second key to sort subtopics whose IAL features are in a draw.

Results are shown in Table 5. All the methods based on individual features are not as good as our method that combines the three features. This indicates that the three features are somehow complementary in measuring good subtopics. Among the three features, GP is the strongest, DR is the second and IAL is the last.

We also conduct Student's paired t test with a two-tailed distribution of any two of the four methods and show the results in Table 6. For example, $(-, \dagger, \dagger, \dagger)$ is the test result between Combined and GP. The elements corresponding to precision, I-rec, D-nDCG and D#-nDCG. That is to say, the difference between Combined and GP is significant in terms of I-rec, D-nDCG and D#-nDCG, but it is not significant in terms of precision. The Combined method is significantly better than DR and IAL+DR in all metrics. As the table shows, GP is also not significantly better than DR in terms of precision and D-nDCG, although the difference is significant in terms of I-rec and D#-nDCG. It indicates that the advantage of GP over DR in not precision but intent recall.

Table 5 We compare the full version of our method (combined) with the versions based on individual ranking features

Cut-off@10	Precision	I-rec	D-nDCG	D#-nDCG
Combined	**0.8230**	**0.5006**	**0.6912**	**0.5959**
GP	0.7825	0.4813	0.6757	0.5785
DR	0.7425	0.4498	0.6749	0.5624
IAL+DR	0.625	0.3613	0.3343	0.3478

Table 6 We conduct Student's paired t test with a two-tailed distribution over any two methods in terms of all four metrics: (precision, I-rec, D-nDCG, D#-nDCG)

Cut-off@10	GP	DR	IAL+DR
Combined	$(-, \dagger, \dagger, \dagger)$	$(*, \dagger, \dagger, \dagger)$	$(\dagger, \dagger, \dagger, \dagger)$
GP		$(-, \dagger, -, \dagger)$	$(\dagger, \dagger, \dagger, \dagger)$
DR			$(\dagger, \dagger, \dagger, \dagger)$

* Means the difference is significant with p value <0.05 in a particular metric; † means the difference is significant with p value <0.01, – means the difference is not significant, i.e., p value ≥ 0.05

5 Conclusion and future work

In this paper, we propose a method to address the query subtopic mining problem. First, we extract four types of query related text fragments from different parts of top hundreds of retrieved documents. Then we cluster fragments and thus merge similar ones and filter some noise. Next, we generate a subtopic for each cluster by mining a core term, expanding the core term to a core phrase and mining the frequent and short span that covers both the query and the core phrase. At last, we combine three features based on clusters and a query language model to rank and diversify the subtopics.

Experimental results show that our proposed method is effective to mine and rank subtopics and it significantly performs better than two baselines of previous work. Our method is better than the query log based baseline mainly in the coverage of intents and it is also better than the search result clustering method mainly in readability of generated subtopics. By further analysis, we find that top 200 retrieved full documents can well cover possible intents of a query and full documents perform better than titles and snippets only; our proposed ranking features are complementary and combining them obtains the best performance.

As the future work, we will apply mined subtopics in several applications, such as search result diversification, multi-facet search and query suggestion. It is also interesting to design new user interactions and thus the mined subtopics can help users in learn what a topic covers and drill down to their interested subtopic. For example, on small touch screens, users may prefer touching a subtopic to reformulate a query. New devices require new search applications that can understand users intents and provide options to select.

Acknowledgments The research was supported in part by National Science Foundation of China under Grant Nos. 61173112, 61070072, 61103160, 61103239; National High Technology Research and Development Program 863 of China under Grant No. 2012AA011003; Cheung Kong Scholars Program; Ministry of Education of China Humanities and Social Sciences Project under Grant No. 12YJC880117. The fifth author is supported by NSFC of China (60903028, 61070014), Key Projects in the Tianjin Science & Technology Pillar Program (11ZCKFGX01100).

References

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

Beeferman, D., &, Berger, A. (2000). Agglomerative clustering of a search engine query log. In *Proceedings of the sixth ACM SIGKDD international conference on knowledge discovery and data mining*, ACM, pp. 407–416.

Carbonell, J., &, Goldstein, J. (1998). The use of Mmr, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval* (pp. 335–336). SIGIR '98. New York, NY: ACM, ISBN 1-58113-015-5. doi:http://doi.acm.org/10.1145/290941.291025.

Chandar, P., &, Carterette, B. (2010). Diversification of search results using webgraphs. In *Proceeding of the 33rd international ACM SIGIR conference on research and development in information retrieval*, ACM, pp. 869–870.

Chen, H., &, Dumais, S. (2000). Bringing order to the web: Automatically categorizing search results. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, ACM, pp. 145–152.

Clarke, C. L., Craswell, N., &, Soboroff, I. (2009). Overview of the trec 2009 web track, technical report, DTIC document.

Clarke, C. L. A., Craswell, N., Soboroff, I., &, Ashkan, A. (2011). A comparative analysis of cascade measures for novelty and diversity. In *Proceedings of the fourth ACM international conference on Web search and data mining*, ACM, pp. 75–84.

Clough, P., Sanderson, M., Abouammoh, M., Navarro, S., &, Paramita, M. (2009). Multiple approaches to analysing query diversity. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, ACM, pp. 734–735.

Cutting, D. R., Karger, D. R., &, Pedersen, J. O. (1993). Constant interaction-time scatter/gather browsing of very large document collections. In *Proceedings of the 16th annual international ACM SIGIR conference on research and development in information retrieval*, ACM, pp. 126–134.

Ferragina, P., &, Gulli, A. (2008). A personalized search engine based on web-snippet hierarchical clustering. *Software: Practice and Experience 38*(2), 189–225.

Gale, W.A., &, Sampson, G. (1995). Good-turing frequency estimation without tears*. *Journal of Quantitative Linguistics 2*(3), 217–237.

Geraci, F., Pellegrini, M., Pisati, P., &, Sebastiani, F. (2006). A scalable algorithm for high-quality clustering ofweb snippets. In *Proceedings of the 2006 ACM symposium on applied computing*, ACM, pp. 1058–1062.

Goffman, W. (1964), On relevance as a measure. *Information Storage and Retrieval 2*(3), 201–203.

Gollapudi, S., &, Sharma, A. (2009). An axiomatic approach for result diversification. In *Proceedings of the 18th international conference on world wide web*, ACM, pp. 381–390.

Hearst, M., Pedersen, J., &, Karger, D. (1995). Scatter/gather as a tool for the analysis of retrieval results. In *Working notes of the AAAI fall symposium on AI applications in knowledge navigation.*

Hearst, M. A., &, Pedersen, J. O. (1996). Reexamining the cluster hypothesis: Scatter/gather on retrieval results. In *Proceedings of the 19th annual international ACM SIGIR conference on research and development in information retrieval*, ACM, pp. 76–84.

Ji, X., &, Bailey, J. (2007). An efficient technique for mining approximately frequent substring patterns. In *Data mining workshops, 2007. ICDM workshops 2007. Seventh IEEE international conference on*, IEEE, pp. 325–330.

Koshman, S., Spink, A., &, Jansen, B. J. (2006). Web searching on the vivisimo search engine. *Journal of the American Society for Information Science and Technology 57*(14), 1875–1887.

Leouski, A. V. (2005). An evaluation of techniques for clustering search results, technical report, DTIC document.

Leuski, A., &, Allan, J. (2000). Improving interactive retrieval by combining ranked lists and clustering. In *Proceedings of RIAO*, vol. 2000.

Li, X., Wang, Y. Y., &, Acero, A. (2008). Learning query intent from regularized click graphs. In *Proceedings of the 31st annual international ACM SIGIR conference on research and development in information retrieval*, ACM, pp. 339–346.

Osinski, S., &, Weiss, D. (2005). A concept-driven algorithm for clustering search results. *Intelligent Systems, IEEE 20*(3), 48–54.

Radlinski, F., Szummer, M., &, Craswell, N. (2010). Inferring query intent from reformulations and clicks. In *Proceedings of the 19th international conference on world wide web*, ACM, pp. 1171–1172.

Rafiei, D., Bharat, K., &, Shukla, A. (2010). Diversifying web search results. In *Proceedings of the 19th international conference on world wide web*, ACM, pp. 781–790.

Robertson, S. E., &, Jones, K. S. (1976). Relevance weighting of search terms. *Journal of the American Society for Information Science 27*(3), 129–146.

Sahami, M., &, Heilman, T. D. (2006). A web-based kernel function for measuring the similarity of short text snippets. In *Proceedings of the 15th international conference on world wide web*, ACM, pp. 377–386.

Sakai, T. (2011). NTCIREVAL: A generic toolkit for information access evaluation. In *Proceedings of the forum on information technology 2011*, vol. 2, pp. 23–30

Sakai, T., &, Song, R. (2011). Evaluating diversified search results using per-intent graded relevance. in *Proceedings of the 34th international ACM SIGIR conference on research and development in Information*, ACM, pp. 1043–1052.

Santos, R. L. T., Macdonald, C., &, Ounis, I. (2010a). Exploiting query reformulations for web search result diversification. In *Proceedings of the 19th international conference on world wide web*, ACM, pp. 881–890.

Santos, R. L. T., Macdonald, C., &, Ounis, I. (2010b). Selectively diversifying web search results, in *Proceedings of the 19th ACM international conference on information and knowledge management*, ACM, pp. 1179–1188.

Sibson, R. (1973). Slink: An optimally efficient algorithm for the single-link cluster method. *The Computer Journal 16*(1), 30–34.

Song, R., Wen, J. R., Shi, S., Xin, G., Liu, T. Y., Qin, T., Zheng, X., Zhang, J., Xue, G., &, Ma, W. Y. (2004). Microsoft research Asia at web track and terabyte track of trec 2004. in *Proceedings of the thirteenth text retrieval conference proceedings (TREC-2004).*

Song, R., Zhang, M., Sakai, T., Kato, M. P., Liu, Y., Sugimoto, M., Wang, Q., &, Orii, N. (2011). Overview of the NTCIR-9 intent task. In *Proceedings. of the 9th NTCIR workshop meeting on evaluation of information access technologies*.

Strohmaier, M., Kröll, M., &, Körner, C. (2009). Intentional query suggestion: Making user goals more explicit during search. In *Proceedings of the 2009 workshop on web search click data*, ACM, pp. 68–74.

Wang, X., &, Zhai, C. X. (2007). Learn from web search logs to organize search results. In *Proceedings of the 30th annual international ACM SIGIR conference on research and development in information retrieval*, ACM, pp. 87–94.

Zamir, O., &, Etzioni, O. (1998). Web document clustering: A feasibility demonstration. In *Proceedings of the 21st annual international ACM SIGIR conference on research and development in information retrieval*, ACM, pp. 46–54.

Zamir, O., &, Etzioni, O. (1999), Grouper: A dynamic clustering interface to web search results. *Computer Networks 31*(11–16), 1361–1374.

Zeng, H. J., He, Q. C., Chen, Z., Ma, W. Y., &, Ma, J. (2004). Learning to cluster web search results. In *Proceedings of the 27th annual international ACM SIGIR conference on research and development in information retrieval*, ACM, pp. 210–217.

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

Zheng, W., &, Fang, H. (2010). University of Delaware at diverstiy task of web track 2010. In *Proceedings of TREC*, vol. 10.

Zheng, W., &, Fang, H. (2011). A comparative study of search result diversification methods. In *Proceedings of DDR* 11.

Zheng, W., Wang, X., Fang, H., &, Cheng, H. (2011). An exploration of pattern-based subtopic modeling for search result diversification. in *Proceedings of JCDL*, vol. 11.