Simulation of Within-Session Query Variations using a Text Segmentation Approach

Debasis Ganguly, Johannes Leveling, and Gareth J. F. Jones

CNGL, School of Computing, Dublin City University, Dublin-9, Ireland
{dganguly, jleveling, gjones}@computing.dcu.ie

Abstract. We propose a generative model for automatic query reformulations from an initial query using the underlying subtopic structure of top ranked retrieved documents. We address three types of query reformulations a) *specialization*; b) *generalization*; and c) *drift*. To test our model we generate the three reformulation variants starting with selected fields from the TREC-8 topics as the initial queries. We use manual judgments from multiple assessors to calculate the accuracy of the reformulated query variants and observe accuracies of 65%, 82% and 69% respectively for specialization, generalization and drift reformulations.

1 Introduction

Laboratory information retrieval (IR) evaluation has generally focused on single query search where a query is applied to an IR system and the effectiveness of retrieving relevant documents is evaluated. However, it is commonly observed that multiple versions of a query are iteratively modified and applied to retrieval systems [1]. Three patterns of query reformulation as observed in real-life search behaviour [2] are the following: a) specialization, where the reformulated query expresses a more specialized information need as compared to the initial query; b) generalization, where the refined information need is more general and covers a broader scope in comparison to the initial query; c) drift (parallel reformulation), where the reformulated query drifts away to another aspect of the initial information need instead of moving to more general or more specific needs. The topic collections of standard ad hoc tracks at IR evaluation workshops provide a set of unrelated topics and hence fail to evaluate the performance of a retrieval system over a session of related queries. The Session Track of TREC [3] aims to evaluate retrieval systems over an entire session of user queries rather than on separate independent topics. The topic creation phase of the Session Track 2010 involved starting with Web-track diversity topics, comprising of separate sub-topics representing different facets of information need, sampled from the query logs of a commercial search engine. Specific variants of the initial topic were created by manually extracting keywords from the different subtopics. The general variants were formed manually in two ways: a) by constructing an over-specified query from one of the subtopics and removing words, and b) by adding selected related words from a different subtopic. For the drifting reformulations,

manually extracted keywords from one of the subtopics was considered as the initial query and those from some other subtopic as the reformulated one.

This paper describes a model to simulate user interactions over a browsing session by automatically generating the above mentioned three types of reformulated queries. We also looks at ways of characterizing the reformulations based on the differences in retrieved sets of documents between the initial and the reformulated queries. The reformulations are qualitatively evaluated by manual judgments. Related work on automatic query reformulation for web-search to better answer the initial original information need itself, includes that of Dang and Croft [4] which uses anchor text to reformulate a query by substituting some of the original terms. Wang and Zhai [5] exploited co-occurrence of terms in search-engine query logs to add terms to correct the mis-specification and under-specification of a query. Our work is different in the sense that we do not intend to improve the initial query but seek to move the query towards a more specific subtopic or a broader topic which is thus associated with a change of the information need. We also aim to develop topic variants on a large scale for ad hoc retrieval collections which do not possess meta-information as query logs or anchor texts.

The rest of the paper is organized as follows: Section 2 discusses the working methodology for generating query sessions, Section 3 hypothesizes the expected characteristics of the retrieved set of documents for the three reformulated versions, Section 4 describes the experiments performed, followed by Section 5 which concludes the paper with directions for future work.

2 Automatically generating query reformulations

A real user would typically start with a general query at the beginning of a session, since initially he might not be aware of the more specific aspects of the information need [6]. Then as he views retrieved documents his knowledge about the topic increases and he may get interested in more specific details about the topic. His reformulation of the initial information need is thus based on the contents of the more focused subtopics. If the user is interested in a more specific reformulation, he is likely to choose terms which occur frequently in one or a few subtopics. Whereas if he is interested in a more general formulation, it is more likely that he would choose terms which are not concentrated in one of the subtopics but occur abundantly throughout the entire document.

Our earlier work [7] shows that a text segmentation based approach can be used to simulate the reformulation patterns of real users and automatically generate the query variants. The differences with our earlier work are as follows: i) in contrast to an additive model for generating general reformulations, we employ removal of words; ii) we propose a method for generating the parallel reformulations; iii) the algorithm for generating the specific reformulations is slightly different in the sense that term selection scores are accumulated over top ranked documents. We now look at each of the reformulation types in more detail.

Text Segmentation or Text-Tiling [8] is the process of decomposing a text into blocks of coherent textual content called segments. Thus each segment content is particularly focused on one subtopic. Our generative model tries to utilize the fact that a term indicative of a more specific aspect of an initial information need, typically is densely distributed in the textual contents of a segment. We use C99 [9], which is shown to perform better than Hearst's original text-tiling algorithm, for segmenting documents. To characterize specific reformulation terms, we assign scores to terms considering the following two factors: a) how frequently a term t occurs in a segment s, denoted by $\texttt{tf}(t, s)$, and how exclusive the occurrence of t in s is, as compared to other segments of the same document, denoted by $\frac{|S|}{\texttt{sf}(t)}$, where $|S|$ is the number of segments in that document and $\texttt{sf}(t)$ is the number of segments in which t occurs; b) how rare the term is in the entire collection, measured by the document frequency ($\texttt{df}$), the assumption being rare terms are more likely to be specific terms. For specific reformulations, we compute the term scores for the most similar segment to the query assuming that this is precisely the section which "catches the eye" of a real-life reader and adding terms from this segment can potentially shift the original query to a more specific information need.

$$\phi(t, s) = a \cdot \texttt{tf}(t, s)\frac{|S|}{\texttt{sf}(t)} + (1 - a) \cdot \log \frac{|D|}{\texttt{df}(t)} \tag{1}$$

$$\psi(t, d) = a \cdot \texttt{tf}(t, d)\frac{\texttt{sf}(t)}{|S|} + (1 - a) \cdot \log \frac{|D|}{\texttt{df}(t)} \tag{2}$$

We use a mixture model to calculate term scores, as shown in Equations 1 and 2. Equation 1 assigns higher values to terms which occur frequently in a segment, occur only in a few segments, and occur infrequently in the collection. The working steps of the proposed method to generate query variants are as follows:

1. For each top ranked R documents do Steps 2-5.
2. Segment a document d into $\{s_1, s_2, \ldots s_n\}$ by executing C99.
3. Let s_{sel} be the segment with maximum and minimum number of matching query terms respectively for specific and parallel reformulations.
4. For general reformulation, score each original query term t by $\psi(t, d)$; otherwise score each term $t \in s_{sel}$ by $\phi(t, s_{sel})$ for specific and parallel reformulations.
5. Average the term scores over documents.
6. Sort each term of the table by its score and add(substitute) top n new terms to the original query for specific(parallel) reformulation type. Retain the top n terms in the query removing the rest for a general reformulation.

When we are done processing a document, we store the term scores from this document and move on to extract terms from the next document, merging the new scores with the previously stored term scores. The state of the stored terms is useful in simulating a user who keeps track of the more specific sub-topical terms as he keeps on reading documents retrieved in response to the initial query. Although he reads each document in turn, his decision of which terms to add for reformulating the query is a global one based on the information gained from

all the top ranked documents read. Merging term scores by averaging out the previous score of a term with the score of that term in the current document simulates this behavior.

In contrast to a more specific term, a more general term is distributed uniformly throughout the entire document text [8]. So an obvious choice is to score a term based on the combination of term frequency in the whole document (instead of frequency in individual segments) and segment frequency (instead of inverse segment frequency) where $\texttt{tf}(t, d)$ is the number of occurrences of t in d (see Equation 2). The model used in generalization removes terms in contrast to the additive model for the specific reformulation type. More precisely, it involves removal of terms of higher $\phi(t, s)$ in the initial query with those having lower ones, thus making general reformulation an inverse to specialization.

To simulate a parallel reformulation, we assume that the user reformulates the initial query by adding specific terms from the least similar (to the initial query) subtopics of the read documents. The differences with the specialization reformulation are that firstly term scores are computed from the least similar segment in contrast to the most similar one and secondly none of the initial query terms are retained in the reformulated query, thus resulting in a substitution based model where we throw away all the initial query terms and add n new terms. Although the reformulated query does not share any common terms, it expresses a parallel information need based on the contents of the documents being read which is different from starting a new session.

3 Reformulation Effect on Retrieval

The original query terms ought to be semantically related to the added specialization terms to make the reformulation precise whereas the reverse is true for generalization reformulations. Although in this paper we have looked at generating the reformulations from an IR perspective, simulating the behaviour of a real-life searcher, an alternative collection independent way of generating reformulations is by using a thesaurus such as the WordNet which has the hierarchic relationships between words encoded in it. Thus, specialization can be achieved by addition of hyponyms or meronyms, whereas generalization can involve addition of hypernyms or holonyms or when viewed as the inverse of specialization - removal of hyponyms or meronyms. It is expected that there will be a smaller number of documents in the collection pertaining to a more specific information need. As the query becomes more specific in nature, the top documents retrieved for the initial query become more general to the new information need and shift down the ranked list. Thus, if we measure overlap of the two ranked lists at specific cut-off points, we would expect a low overlap in the top 10 or top 20, whereas a high degree of overlap beyond that. To measure the shift in the rankings of top ranked documents, we define the *net perturbation* of top m documents for a query q as:

$$p(q, m) = \frac{1}{m} \sum_{k=1}^{m} \delta_k \tag{3}$$

where $\delta_k = |(\texttt{newrank}(d_k) - k|$ if d_k exists in the ranked list of the reformulated query, and $\delta_k = 1000$ otherwise. For the specific reformulations we would expect a lower net perturbation value as compared to generalization and drift. Since generalization involves removal of terms from the original query it opens up a wider range of documents to be retrieved for the reformulated query. Hence, we would expect a lower degree of overlap of the two ranked lists as compared to the specialization case. Since parallel reformulation often involves using substituted words, we expect further lower overlap and higher net perturbation of top ranked documents. We perform experiments to test these assumptions.

4 Experiments

We start with the titles of the TREC-8 topics as initial queries for generating the specific and parallel reformulation variants. For the general reformulations we use the description field of the TREC-8 topics as the initial query. We use 5 top ranked documents for reformulating each query and we put $a = 0.5$ for computing the $\phi(t, s)$ and $\psi(t, d)$ scores i.e. equal importance is given to the term distribution factor in a document and the *idf* factor of that term. We add at most 3 additional terms for the specific and parallel reformulations, and retain at most 2 terms from the description field to construct the general reformulations. Manual judgments regarding the quality of the generated queries were provided by two assessors through yes/no answers. Table 1 shows the assessor judgments on each reformulation type over 50 TREC-8 topics. From Table 1 we see that two assessors have the highest inter-agreement for the parallel reformulation type. This is expected because parallel reformulation always involves a change in information need and is more obvious to judge than the other two. The lowest inter-agreement is on specialization which can be attributed to the fact that such a reformulation involves addition of new words which ought to be semantically related to the original keywords, and the degree of semantic closeness is often subject to personal judgments. Table 1 also confirms the resultset change hypothesis discussed in Section 3. Specific reformulations show the highest overlap with the initial retrieved set of documents. Drift reformulations exhibit minimum overlap and for general reformulations overlap percentages are somewhere in between the two extremes. We also see that the specific and general reformulations are associated with an increase in overlap percentage with increasing

Table 1: Evaluation of the generated reformulations. O(m) denotes the avg. % overlap and $p(m)$ denotes the avg. *net perturbation* of m top ranked docs.

Reformulation	Manual assessments		Resultset measures				
type	Assessor-1	Assessor-2	O(10)	O(20)	O(50)	O(500)	$p(5)$
Specific	39 (78%)	26 (52%)	39.0	38.1	42.7	44.7	367.9
General	39 (78%)	43 (86%)	22.4	22.5	24.5	32.2	2208.6
Parallel	34 (68%)	35 (70%)	12.0	10.2	8.6	5.89	3853.3

cut-off rank, thus indicating that moving down the ranked list results in finding more documents already retrieved for the original query. However the drift reformulation exhibits a decreases in overlap with increasing cut-off rank, indicating that we find more unseen documents as we walk down the ranked list. Table 1 shows that the average net perturbation as defined in Equation 3, is the least for specific reformulation type. Thus, the top 5 documents of the initial ranked list can be found not too far down the reformulated ranked list.

5 Conclusions

This paper presented a novel approach of generating simulated query sessions without using real-life search logs or external resources, simulating real-life reading behaviour by adding frequent sub-topical terms to form a more specific query, removing frequent non-uniformly distributed terms to form more general queries, and substituting dense sub-topical terms to construct parallel reformulations. Results show that our generative model can be used to produce query reformulations with an average accuracy of 65%, 82% and 69% for the specialization, generalization and drift reformulations respectively. This paper also provides arguments for the expected changes in the reformulated result-set of documents in comparison with the initial retrieval set by introducing measures such as average percentage overlap at fixed number of documents and the average net perturbation. Retrieval results on the reformulated queries confirm the hypothesis put forward. Based on our findings we conclude that our automatic method of query generation can be used to generate topic variants on a large scale and thus create simulated user sessions with no manual intervention.

Acknowledgments This research is supported by the Science Foundation Ireland (Grant 07/CE/I1142) as part of the Centre for Next Generation Localisation (CNGL).

References

1. Bates, J., M.: The Design of Browsing and Berrypicking Techniques for the Online Search Interface. Online Review **13**(5) (1989) 407–424
2. Jansen, B.J., Booth, D.L., Spink, A.: Patterns of query reformulation during web searching. J. Am. Soc. Inf. Sci. Technol. **60** (July 2009) 1358–1371
3. Kanoulas, E., Clough, P., Carterette, B., Sanderson, M.: Session track at TREC 2010. In: SIMINT workshop SIGIR '10, New York, NY, USA, ACM (2010)
4. Dang, V., Croft, B.W.: Query reformulation using anchor text. In: Proceedings of WSDM '10, New York, NY, USA, ACM (2010) 41–50
5. Xuanhui, W., ChengXiang, Z.: Mining term association patterns from search logs for effective query reformulation. In: Proceedings of CIKM'08. (2008) 479–488
6. Leveling, J., Ghorab, M.R., Magdy, W., Jones, G.J.F., Wade, V.: DCU-TCD@logCLEF 2010: Re-ranking document collections and query performance estimation. In: CLEF (Notebook Papers/LABs/Workshops). (2010)
7. Ganguly, D., Leveling, J., Jones, G.: Automatic generation of query sessions using text segmentation. In: SIR workshop at ECIR '11. (2011)

8. Hearst, M.: TextTiling: Segmenting text into multi-paragraph subtopic passages. CL **23**(1) (1997) 33–64
9. Choi, F.Y.Y.: Advances in domain independent linear text segmentation. In: Proceedings of the NAACL 2000. (2000) 26–33