Query Term Ranking based on Search Results Overlap

Wei Song, Yu Zhang, Yubin Xie, Ting Liu, Sheng Li
Computer Science and Technology Department
Harbin Institute of Technology, China
{wsong, yzhang, ybxie, tliu, lisheng}@ir.hit.edu.cn

ABSTRACT

In this paper, we propose a method to rank and assign weights to query terms according to their impact on the topic of the query. We use *Search Result Overlap Ratio (SROR)* to quantify the overlap of the search results of the full query and a shorten query after removing one term. Intuitively, if the overlap is small, it indicates a big topic shift and the removed term should be discriminative and important. The *SROR* could be used for measuring query term importance with a search engine automatically. By this way, learning based models could be trained based on a large number of automatically labeled instances and make predictions for future queries efficiently.

Categories and Subject Descriptors

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

General Terms

Algorithm, Experimentation, Performance

Keywords

Query Term Ranking, Query Reformulation, Search Results Overlap

1. INTRODUCTION

Query term ranking aims to assess query terms' importance and effectiveness. It has many potential applications. For example, it helps for formulating better queries by tuning term weights in retrieval models or removing ineffective terms. The important query terms or concepts also help for understanding a query's topic.

Recently, machine learning techniques have been adopted for similar tasks such as classification for key concepts identification [1], regression models for term ranking [3] and learning to rank for concept importance weighting [2]. Most of these approaches need manually labeled training data, either labeled key concepts or document relevance.

This paper proposes a simple approach based on *Search Result Overlap Ratio (SROR)*. The basic idea is to quantify a term's impact by considering the overlap of the search results of the full query and the shorten query leaving out the term. We assume the search results of a query represent its topic. If the overlap is small, it indicates the query's topic shifts a lot after removing the term. So the removed term plays an important role for determining the query's topic and should be discriminative. The experimental results on manually labeled queries showed that *SROR* based term ranking method had acceptable consistence with human judgement. This approach requires searching multiple times for a given query but the whole process can be done automatically with a search engine and does not need any human assistance. As a result, it could be used for automatically computing term weights for a large number of queries with a search engine beforehand. Learning based models could be trained using the statistical features extracted from these "labeled" instances. The learned models can make predictions for future queries efficiently. The experimental results on real web queries indicated the reformulated queries with *SROR* based weighting schemes improved the retrieval performance.

SIGIR'11, July 24–28, 2011, Beijing, China.
ACM 978-1-4503-0757-4/11/07.

2. SEARCH RESULTS OVERLAP RATIO

For a given query $q=\{t_i\}$, we define the *Search Result Overlap Ratio (SROR)* for term $t_i \in q$ as

$$SROR(t_i) = \frac{\#(\varphi_N(q) \bigcap \varphi_N(q_{\neg t_i}))}{\#\varphi_N(q)} \quad (1)$$

where $\varphi_N(q)$ denotes the top N search results of query q returned by a search engine. $q_{\neg t_i}$ refers to the query after removing term t_i from query q. The sign # is used to count the size of a set. Heuristically, if $SROR(t_i)$ is large, it means t_i is not so important, because this term has little impact on the topic of query q. Otherwise, it indicates this term is discriminative for the full query. Based on $SROR(t_i)$, we assign weights to terms according to

$$Weight(t_i) = 1 - SROR(t_i) \quad (2)$$

$Weight$ can be used for both ranking and assigning weights to query terms. We demonstrate a query in Table 1. We can see the query term importance presented by normalized $Weight$ ($Weight(t_i)/\sum_{t_j \in q} Weight(t_j)$) is consistent with human intuition. Similar idea has been used for predicting query performance [5] by measuring the overlap of search results between using the full query and the individual query term. However, individual term may be not enough to indicate the query topic sometime. Instead, we borrow the idea from [3] they assume leaving out an important term results in decrease retrieval effectiveness. In our case, we assume removing an important term may lead to topic shift.

Table 1: Terms' *SRORs* and normalized *Weights* of the query "movie free download website".

Query Terms	movie	free	download	website
SROR	0.0	0.8	0.3	0.5
Weight(Norm)	0.417	0.083	0.292	0.208

3. EXPERIMENTS

The experiments were conducted on real web queries and a web corpus. We used one month query log collected by Sogou[1] during June 2008 consisting of 5604251 unique queries and associated clicks. For 99.67% queries, the query length $l(q) \leq 10$. The web corpus containing 5 TB web pages was also crawled by Sogou during 2008. We used Indri [4] for indexing and searching this corpus. The top 30 search results of a input query returned by a search engine were used for computing *SROR*.

Term Ranking. We computed *SROR* using Sogou and Baidu[2] search engines respectively. To evaluate the ability of ranking terms using *SROR*, we accessed the agreement between the automatically generated rankings and human labelers. We sampled 120 queries, $3 \leq l(q) \leq 10$. Two labelers were asked to rank terms in each query according to the term importance for the full query. We computed Kendall's W coefficient[3], which ranges from 0 (no agreement) to 1 (complete agreement), between different rankings for each query. The average score over all queries is reported.

The agreement between using two search engines is consistent (0.77). It indicates *SROR* is stable across different search engines. The agreement between two labelers is 0.75. The average agreement between one *SROR* based ranking (using Sogou or Baidu) and one labeled ranking (by labeler 1 or labeler 2) is 0.64. The average agreement among one *SROR* based ranking and two labeled rankings is 0.57. Due to the difficulty of this task, there is a positive correlation.

Retrieval Effectiveness. For evaluating the retrieval effectiveness of the *SROR* based weighting schemes, we sampled test queries from the query log. Each query must have at least 5 associated clicked pages in the indexed corpus and $3 \leq l(q) \leq 10$. Finally, we collected 1538 test queries, 113 of which contained at least 7 terms. We viewed user clicked pages as relevant. The average MAP and p@10 over all queries were computed.

We could directly apply $Weight$ for assigning weights to terms, noted as **WT**. This strategy has to search many times for a given query. For efficiency, we used $Weight$ to automatically weight query terms for a large number of queries beforehand with a search engine. Then we trained a regression model based on statistical features extracted from the labeled instances. The learned model could predict weights for future queries efficiently at querying time. We trained this regression model based on 2000 queries automatically labeled by $Weight$ using Baidu search engine, noted as **REG**. These queries contain 4 to 6 terms each. We do not use long queries for training, because the current search engines do not perform so well on such queries compared to shorter queries. The extracted features include *term index within a query, times the term occurs in query log as a whole query, average rank of associated clicks, corpus tf, corpus idf and mutual information among query terms.*

[1]http://www.sogou.com/
[2]http://www.baidu.com/
[3]http://en.wikipedia.org/wiki/Kendall's

Table 2: The retrieval effectiveness comparison of different weighting strategies on queries with different length.

%	MAP		p@10	
Length	[3,6]	[7,10]	[3,6]	[7,10]
UNF	9.84	4.51	8.95	1.96
WT	**10.23**	4.81	**9.23**	**2.23**
REG	10.02	4.86	9.02	2.06
IDF	9.37	**5.00**	7.92	2.05

Table 2 shows the retrieval effectiveness using different weighting methods on queries with different length. **UNF** denotes giving terms uniform weights. **IDF** denotes the corpus based IDF approach which solely depends on the term distribution within the corpus. We can see **WT** which directly uses $Weight$ for term weighting outperforms other methods in most cases. It means *SROR* is a useful signal for query term ranking. The regression model **REG** performs better than **UNF** as well. It indicates the potential to use *SROR* for automatically labeling. **IDF** achieves good results when dealing with long queries but performs poorly on queries with moderate length in our experiments.

4. CONCLUSIONS

In this paper, we have proposed a simple approach to measure query term importance by examining the search results overlap ratio. The experiments showed *SROR* based weighting schemes improved the retrieval performance. We believe our method could be improved further if noun phrases or multi-term concepts are identified and used, because they explain the query topics better than individual terms. We also plan to incorporate more features for learning based on automatically labeled instances using *SROR*.

Acknowledgments

This research is supported by NSFC under Grant No. 60736044, 61073126 and 61073129, by HIT.KLOF.2009020.

5. REFERENCES

[1] M. Bendersky and W. B. Croft. Discovering key concepts in verbose queries. *In Proc. SIGIR 2008*, pages 491–498, 2008.

[2] M. Bendersky and W. B. Croft. Learning concept importance using a weighted dependence model. *In Proc. WSDM 2010*, pages 133–142, 2010.

[3] S.-H. K. Chia-Jung Lee, Ruey-Cheng Chen and P.-J. Cheng. A term dependency-based approach for query terms ranking. *In Proc. CIKM 2009*, pages 745–770, 2009.

[4] T. Strohman, D. Metzler, H. Turtle, and W. B. Croft. Indri: A language-model based search engine for complex queries (extended version). IR 407, University of Massachusetts, 2005.

[5] E. Yom-Tov, S. Fine, D. Carmel, and A. Darlow. Learning to estimate query difficulty: including applications to missing content detection and distributed information retrieval. In *In Proc. SIGIR 2005*, pages 512–519. ACM, 2005.