Applying the Query Change Retrieval Model on Session Search -

Georgetown at TREC 2013 Session Track

Sicong Zhang and Hui Yang

Department of Computer Science, Georgetown University 37th and O Street NW, Washington DC, USA sz303@georgetown.edu, huiyang@cs.georgetown.edu

Abstract In TREC 2013 Session Track, we experiment the Query Change Model (QCM) for session search on the new document collection ClueWeb12 CatB. We use structured query formulation and pseudo-relevance feedback for RL1. The QCM approach is used in RL2 and it studies query change between adjacent queries and models session search as a Markov Decision Process. We further add information from other sessions by a majority vote of cross-session clicked data to the model in RL3. Comparing the retrieval accuracy in RL1 with that in RL2 and RL3, we obtain 46.1% and 46.6% improvements, respectively. We present an analysis and discussion on the dataset difference between ClueWeb12 Cat A and Cat B, the difference between TREC2012 and TREC2013 session, and their impact on the retrieval accuracy.

1. Introduction

Session search is the Information Retrieval task that retrieves documents for the entire session. TREC 2013 Session Track aims to retrieve relevant documents for the last query q_n in a session. A session contains a sequence of queries $q_1, q_2 \dots q_{n-1}, q_n$, the ranked list of URLs for each past query, the clicked urls, and the time spent on those webpages. In TREC 2013 Session Track, there are three subtasks: RL1, a subtask that only allows using the last query q_n and ignoring all other information in the query log; RL2, a subtask that only allows using information from the prior history in the same session; RL3, a subtask that allows using any information available in the query log, including those from other sessions.

For RL1, we use structured query formulation in combination with Lemur's language modeling approach.¹ For RL2, we employ an approach based on the Query Change Model [1], which models session search as a Markov Decision Process. For RL3, we use a majority vote of cross-session clicked data for further improvement. In the following sections, we present our methods and describe and analyze the submitted runs.

2. RL1 - The Basic Retrieval Model

Our approach for RL1 is based on the Multinomial language models with Dirichlet smoothing [2]. We calculate the term weight P(t|d) as:

$$P(t|d) = \frac{TF(t,d) + \mu P(t|C)}{length(d) + \mu}$$

where t is the term and d is the document under evaluation, length(d) is the length of the document, P(t|C) calculates the probability that t appears in corpus C based on the Maximum-Likelihood Estimation (MLE), and μ is the Dirichlet smoothing parameter.

We use two different methods to formulate structured queries. Both are based on query term groupings provided by quotation marks in the original query. The Strict Quotation Method always considers a quoted phrase as a single term. For example, "federal government" is grouped as one term and we write the Lemur query as #1(federal government). The Relax Quotation Method considers the quoted phrase as one single term only when the quoted parts frequently appear in the query log. In RL1, we employ the Strict Quotation Method before sending queries to Lemur. Pseudo relevance feedback is also used in RL1.

¹ http://lemurproject.org/

² http://durum0.uwaterloo.ca/clueweb09spam/

3. RL2 - The Query Change Retrieval Model

During a session, users usually continuously modify their queries to change expressions to better describe their information needs. In our recent work [1], we consider query change as a form of relevance feedbacks, and propose to model session search as a Markov Decision Process (MDP). Our RL2/RL3 approaches for TREC 2013 Session Track are based on this QCM model. Here we introduce how we apply this model. First, we define the query change between adjacent queries as the syntactic editing differences of q_i and q_{i-1} : $\Delta q_i = q_i - q_{i-1}$. Then, we decompose each adjacent query pair into three parts: the added terms $(+\Delta q_i)$ the removed terms $(-\Delta q_i)$ and the theme terms (q_{theme}) [1].

Session	Queries	Query Change	Q _{theme}
Session 4	q1 = heart attack details	$+\Delta q_2 = \emptyset$	heart attack
	q2 = heart attack	$-\Delta q_2 = details$	
	q3 = heart attack statistics	$+\Delta q_3 = statistics$	heart attack
		$-\Delta q_3 = \emptyset$	

Table 1. A Query Change Example (TREC 2013 Session 4).

Table 1 shows an example of TREC 2013 Session queries and the corresponding query changes. The QCM model evaluates the relevance between a document d and the current query q_i as in the following equation:

$$Score(q_{i}, d) = P(q_{i}|d) + \gamma \sum_{a} P(q_{i}|q_{i-1}, D_{i-1}, a) \max_{D_{i-1}} P(q_{i-1}|D_{i-1})$$

where γ is the discount factor. In an Markov Decision Process, this discount factor γ is usually used for future states; however, we use it in a reversed order to discount the prior queries. When $\gamma = 1$, there is no discount for the past queries.

Considering different query changes as user's feedback, we utilize four kinds of term weight adjustments based on the types of their query changes. W_{Theme} , $W_{Add,In}$, $W_{Add,Out}$, W_{Remove} are the term weighting adjustments [1]. We then aggregate the relevance score between the query q_i and a document d. Thus, the query to document relevance score can be calculated as:

$$Score(q_i, d) = \log P(q_i|d) + \alpha W_{Theme} - \beta W_{Add,In} + \varepsilon W_{Add,Out} - \delta W_{Remove}$$

where $\log P(q_i|d)$ is the query-document relevance scoring function in the log form. Parameters $\alpha, \beta, \varepsilon$ and δ are the linear weighting coefficients for each type of query changes. We empirically set the parameters as $\alpha = 2.2$, $\beta = 1.8$, $\varepsilon = 0.07$ and $\delta = 0.4$ in the submissions. The final relevance score between the last query q_n and the document is calculated as:

$$Score_{session}(q_{n,d}) = \sum_{i=1}^{n} \gamma^{n-i} Score(q_{i,d})$$

where *Score* $(q_1, d) = \log P(q_1|d)$.

4. RL3 - Utilizing Cross-Session Information

The RL3 subtrack in this year is the first time that participants are allowed to use prior information from other sessions. Since sessions about different topics may barely contribute to each other, we only apply the cross-session information from sessions on the same topics. The topic to session mapping is provided by TREC.

Considering the SAT Clicked Documents (clicked documents that have more than 30 seconds dwell time) may show higher relevance to the information need, we take all SAT Clicks from sessions of the same topic and perform a majority vote. In each session s_i , the document $d_{relevant}(s_i)$ that appears most as SAT Clicks in sessions sharing the same topic is considered as "relevant" and is ranked as the top on the result list:

$$d_{relevant}(s_i) = \underset{d}{\operatorname{argmax}} \sum_{s} f(d, s, s_i)$$

where d represents a document under comparison, s represents a session in the entire TREC 2013 Session query log. The voting function $f(\cdot)$ is defined as the following:

$$f(d, s, s_i) = \begin{cases} 1, & d \in SAT(s), topic(s) = topic(s_i) \\ 0, & otherwise. \end{cases}$$

where SAT(s) is the set of SAT Clicked documents in session s, and topic(s) represents for the topic id of session s. Document d gets a vote from session s if d is an member of SAT(s) while both session s and s_i share the same topic. The total votes for d is the sum of votes from each session, and the selected $d_{relevant}(s_i)$ is the document that receives the most votes.

We further classify the sessions into two types: comparison sessions and top-down sessions. We notice that 19 sessions in TREC 2013 compare two themes or concerning both sides of the same theme. On the other hand, other sessions' topics follow a top-down hierarchical structure and all queries are related to a single theme. We classify the sessions into these two session types and treat them differently during parameter tuning.

5. Experiments

5.1 Dataset and Metrics

Our system is built on top of the Lemur search engine. We use ClueWeb12 Category B (CatB) for TREC 2013 submissions. nDCG@10 is the main evaluation metric. All parameter tuning are based on TREC 2012 and ClueWeb09 CatB. Previous research shows that the document collections like ClueWeb09 and ClueWeb12 contains many spam documents. We filter spam documents according to Waterloo Spam Ranking for both corpora.²³ All documents with spam ranking scores less than 70 are filtered out before the indexing takes place. No topic descriptions are used in the submissions.

Table 2. Methods and parameter settings in TREC 2013 submissions.

μ is the Dirichlet smoothing parameter, k is the number of pseudo relevance feedback,

RUN	RL1	RL2	RL3	
GUrun1	Structured query	Query Change Model	Query Change Model	
	$\mu = 4000, k = 20$	Relax Quotation	SAT Click Majority Vote	
		$\mu = 5000, \gamma = 0.98$	Relax Quotation	
			$\mu = 5000, \gamma = 0.98$	
GUrun2	Structured query	Query Change Model	Query Change Model	
	$\mu = 3500, k = 10$	Relax Quotation	SAT Click Majority Vote	
		$\mu = 5000, \gamma = 1$	Relax Quotation	
			$\mu = 5000, \gamma = 1$	
GUrun3	Urun3 Structured query Query Change M		Query Change Model	
	$\mu = 5000, k = 20$	Strict Quotation	Relax Quotation	
		$\mu = 5000, \gamma = 1$	$\mu = 5000, \gamma = 0.98$	

γ is discount factor in Ouery Change Model.

5.2 TREC 2013 Submissions

We submitted three runs to TREC 2013 Session Track. The run names, methods, and major parameters are listed in Table 2. μ is the Dirichlet smoothing parameter, k is the number of pseudo relevance

² http://durum0.uwaterloo.ca/clueweb09spam/

³ http://www.mansci.uwaterloo.ca/~msmucker/cw12spam/

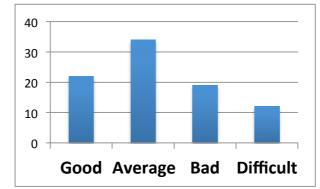


Figure 1. Session difficulty levels. (Y-Axis: # of sessions)

Туре	GUrun2			Median		Max			
-Session	RL1	RL2	RL3	RL1	RL2	RL3	RL1	RL2	RL3
Good-33	0.2727	0.4362	0.4362	0.1005	0.0571	0.0571	0.3511	0.4362	0.4362
Average-21	0.4576	0.2016	0.1593	0.1486	0.2016	0.1593	0.5255	0.4959	0.4542
Bad-83	0.0249	0	0	0	0	0	0.4427	0.2837	0.4328
Difficult-57	0	0	0	0	0	0	0	0	0

feedback, and γ is the discount factor [1]. Four more parameters α , β , ϵ and δ are fixed in RL2 and RL3. Since the sessions are much longer in 2013 than in 2012, for instance, session 87 contains 21 queries, we generated runs with different discount factors. The selected submitted runs used either $\gamma = 0.98$ or $\gamma = 1$.

5.3 Error Analysis

We classify all 87 TREC 2013 sessions into "Good", "Average", "Bad", and "Difficult" based on how our runs perform using the official TREC reported nDCG@10. Low nDCG@10 score means a bad or difficult session while high nDCG@10 score means a good session. Figure 1 shows the distribution of the session difficulty levels. In total, we find 22 sessions as "Good", 34 as "Average", 19 as "Bad", and 12 as "Difficult".

Table 3 shows the nDCG@10 scores for the sessions just mentioned for GURun2, the TREC median run, and the TREC max run. In the "Good" sessions, our runs perform very well and have leading results. These sessions tend to show clear query editing patterns of adding and removing terms, which our model is designed for. For example, our run GURun2 gives the best performance among all groups for session 33. The four queries of session 33 are:

- $q_1 = general schedule pay tables$
- $q_2 = general schedule pay tables 2013$
- $q_3 = general$ schedule pay tables federal wage system
- $q_4 = general schedule pay tables "federal government"$

We get "Average" results in 34 sessions, for instance, session 21, where our runs perform comparable with other groups.

In the 19 "Bad" sessions, we find that one of the major reasons prevents us from getting good results is a bias of relevant documents away from CatB. One typical example is Session 83, which corresponds to topic 65. The ground truth for Session 83 has 12 relevant documents from CatA, while only 2 from CatB. This bias away from CatB makes it difficult for our runs to show good performance. This bias suggests that systems running over CatA and systems running over CatB cannot be directly compared.

Lastly, there are 12 sessions where no submitted runs receive good nDCG@10 score; we classify them as "Difficult". Examples include session 57.

5.4 Results, CatB and CatA

Six groups participated in TREC 2013 Session Track: 2 groups used CatB and 4 groups used CatA. All runs are evaluated together as if they were CatA submissions. We submitted CatB runs. Table 4 presents the nDCG@10 scores for our submitted runs.

	GUrun1	Improvement	GUrun2	Improvement	GUrun3	Improvement
RL1	0.1043	-	0.1013	-	0.105	-
RL2	0.1459	+ 0.0416 (39.9%)	0.148	+ 0.0467 (46.1%)	0.143	+ 0.0380 (36.2%)
RL3	0.146	+ 0.0001 (0.07%)	0.1485	+ 0.0005 (0.34%)	0.1459	+ 0.0029 (2.03%)

Table 4. nDCG@10 scores for TREC2013 runs.

(RL2 improvements compared with RL1, RL3 improvements compared with RL2)

Table 5. ClueWeb12 CatB vs. ClueWeb12 CatA						
	ClueWeb12 CatB	ClueWeb12 CatA				
# of Documents	52,343,021	733,019,372				
Corpus Size Uncompressed	1.95TB	27.3TB				
# of Relevant Documents	[683, 1491]	3428				

[19.9%, 43.5%]

100%

After analyzing ClueWeb12 CatB and CatA, we realize that we couldn't directly compare systems on these two collections. Although CatB is a subset from CatA, many relevant documents from CatA are not in CatB therefore a CatB system has no chance to retrieve most relevant documents. A random sample from the ground truth table (the set of all relevant documents corresponding to all sessions) shows that, only 19 out of 60 (31.7%) relevant documents from CatA are included in CatB. Based on an approximation using the hyper-geometric distribution, we get a 95% confidence interval of the actual amount of relevant documents in CatB as [683, 1491], while there are 3428 relevant documents in CatA.

Table 5 shows the statistics about both document collections. We show that these two corpora, CatB and CatA, are quite different from each other. Their major differences include but are not limited to document collection scales and coverage of the relevant documents. If only compare with our own CatB submissions among the 3 subtasks, our approaches effectively increased the retrieval result of the last query by utilizing the session prior information (RL2, by 46.1%) and the cross-session prior inforamtion (RL3, by 46.6%).

5.5 Session Categories

% of Relevant Doc Included

All the session tasks are classified under two dimensions: a session can be either Factual (F) or Intellectual (I), and with a goal that can be either Specific (S) or Amorphous (A) [3]. Therefore, all session can be divided into four categories: F-S, F-A, I-S, and I-A. Table 6 shows our performance improvements from RL1 to RL2 for each of the four categories. Improvements from the TREC median runs are also listed.

Compared with the median run, we can see that our approach (QCM) is significantly more effective in search accuracy for Amorphous (A) sessions. This is because we employ previous results to guide the search engine to improve retrieval accuracy, and the search process of these Amorphous sessions have more uncertainty and rely more on previous search results. They fit well to each other. Therefore our model performs well in Amorphous sessions.

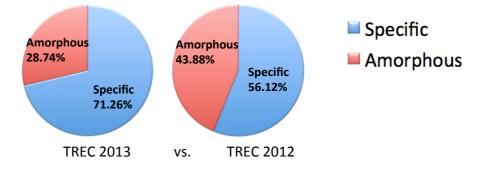
Figure 2 shows the percentage of different session categories for TREC 2012 and TREC 2013. As we can see, TREC 2013 has much less Amorphous sessions (28.74%) than TREC 2012 has (43.88%). Since these Amorphous sessions are where our approach is strong about, the potential of our approach has not been fully explored in TREC 2013. On the TREC 2012 dataset, we can achieve an nDCG@10 of 0.3368, which is statistically significantly better than last year's TREC best runs. [1]

Task: Factual(F)/Intellectual(I), Goal: Specific(S)/Amorphous(A).					
	GUrun2 (RL1 -> RL2)	Median (RL1 -> RL2)			
F-S	+0.0168	+0.0175			
F-A	+0.1463	+0.1028			
I-S	+0.0557	+0.0264			
I-A	+0.0559	-0.0143			

 Table 6. Search accuracy improvement.

 Task: Factual(F)/Intellectual(I), Goal: Specific(S)/Amorphous(A)

Figure 2. Session Category Differences between TREC 2013 and TREC 2012.



6. Conclusions

In this work, we employ the Query Change Retrieve Model on the new document collection ClueWeb12 CatB, and perform a cross-session majority vote approach based on clicked data. The evaluation results show that our method of using prior session history is effective and can bring a 46.1% improvement from RL1 to RL2, and an extra 0.34% improvement from RL2 to RL3. By keeping track of query changes, our model performs well on amorphous sessions, which show more uncertainty and rely more on the previous search results. With less amorphous sessions and more specific sessions, the strength of our model has not been fully demonstrated in TREC 2013. Our analysis on the document collections also shows the difference between the new ClueWeb12 CatB and CatA corpora.

7. Reference

[1] Dongyi Guan, Sicong Zhang, and Hui Yang. 2013. Utilizing query change for session search. In Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval (SIGIR '13). ACM, New York, NY, USA, 453-462.

[2] C. Zhai and J. Lafferty. A study of smoothing methods for language models applied to information retrieval. ACM Trans. Inf. Syst., 22(2):179–214, Apr. 2004.

[3] E. Kanoulas, B. Carterette, M. Hall, P. Clough, and M. Sanderson. Overview of the TREC 2013 Session Track. In the Proceedings of the TREC 2013 Conference.