University of Essex at the TREC 2010 Session Track

M-Dyaa Albakour, Udo Kruschwitz, Jinzhong Niu, Maria Fasli
School of Computer Science and Electronic Engineering
University of Essex
Wivenhoe Park, Colchester, CO4 3SQ, UK
{malbak,udo,jniu,mfasli}@essex.ac.uk

Abstract

This paper provides an overview of the experiments we carried out at the TREC 2010 Session Track. We propose an approach for interpreting reformulated queries by using query expansions derived from anchor logs which we envisage to be a potential alternative to query logs. We show that expansion with terms or phrases extracted from anchor logs improves the retrieval performance over a search session. We provide a detailed discussions of our runs which were among the top performing systems of the track.

1 Introduction

The Session Track was introduced at the Text REtrieval Conference (TREC) 2010. The Session Track aims to evaluate the ability of search engines to utilise previous user interactions in order to provide better results for subsequent queries in a user session and therefore 'point way' to what the user is actually looking for.

Our contribution to the Session Track is based on the idea that related queries can be derived from queries submitted within the same session. The wider context is the AutoAdapt project[1] which looks at automatically building and adapting domain models from the users' search and browsing behaviour (using query logs). These domain models are used to assist users to find information by suggesting query modification or browsing suggestions in their search. We have shown that learning query modification suggestions based on log data using adaptive algorithms (such as an ant colony optimization approach) can be effective [5]. We envisage that these adaptive domain models can be particularly useful for the problem introduced in the Session Track. Our approach is to use these models to apply query expansion based on query relations extracted from large query logs.

[1] http://autoadaptproject.org

Due to the lack of availability of query logs suitable for this year's task, we used anchor logs instead. Anchor text has shown to be effective for a variety of information retrieval tasks. This includes ad-hoc search and the diversity task [11], [3]. Anchor text can be considered as a replacement to user queries as often web authors use similar labels to describe web pages to those used by searchers to find them [6]. Moreover, Dang and Croft have recently shown how anchor text can be used to simulate user sessions. They have considered all the anchor text pointing to the same document as queries in the same user session [4]. In this work we adopted this technique to simulate query sessions.

Our objective of taking part in this year's Session Track is to see whether adaptive models created from simulated query logs can be actually utilised in interpreting query reformulations. We show that expanding the reformulated query using query terms and phrase derived from these models can improve the performance over a baseline system. We also discuss the results in the light of what has been submitted by all participants.

The rest of the paper is structured as follows. In section 2 we give a brief description of the task introduced this year. We describe the dataset and the resources used in our runs in section 3. We explain the experiments and the runs submitted to TREC in section 4. The results of those runs are then discussed in section 5. Finally, we give a brief conclusion in section 6.

2 The Task

The Session Track tries to evaluate the effectiveness of search engines in interpreting query reformulations. The goals set for the Session Track are: **(G1)** to test whether systems can improve their performance for a given query by using a previous query, and **(G2)** to evaluate system performance over an entire query session instead of a single query [10], [1]. Participants are given a set of 150 query pairs, each query pair (original query, query reformulation) represents a user session. These pairs were simulated from the TREC 2009 Web track 2009 diversity topics [10]. The participants are asked to submit three ranked lists of documents from the ClueWeb09 dataset:

- One for the original query ($RL1$).
- One for the query reformulation ignoring the original query ($RL2$).
- One for the query reformulation taking the original query into consideration ($RL3$).

Based on previous work in analysing query logs, the Session Track identifies three different types of query reformulations. Each query pair provided to participants is considered to belong to one of these types. The session types as explained in [1], [10] are:

1. **Generalisation**: In this case, the user starts with a query and gets back some results which may be too narrow or they realise that they wanted a

broader spectrum of results, so they reformulate to a more general query. e.g.‘low carb high fat diet’ → ‘types of diets’.

2. **Specification**: In this case, the user starts with a query and gets back some results which may be too broad or they realise that they wanted results within a specific category or subtopic, so they reformulate to a more specific query. e.g. ‘us map’ → ‘us map states and capitals’

3. **Drifting/Parallel Reformulation**: This type represents the case of starting with a query and then reformulating with another query at the same of level of specification but with a different aspect of information need. e.g. ‘music man performances’ → ‘music man script’.

The type of query reformulation is not known to the participants. In our runs we did not attempt to automatically classify a query pair into one of the three categories and therefore we treated all pairs equally.

3 Experimental Setup

The ClueWeb09 dataset[2] is a web crawl of more than a billion pages that has been used in last year’s Web track. The ClueWeb09 category B dataset is a subset of the larger ClueWeb09 crawl and it consists of 50 million English pages. In this year’s task participants were permitted to use either one of the two datasets. An existing Indri[3] index of the ClueWeb09 dataset is already available and searchable via a public web service[4]. The web service would enable us to issue queries and retrieve the top documents returned by the search engine, thus removing the burden of indexing the data internally. The Indri search engine uses language modelling probabilities and supports query expansion.

In our experiments we aim to use anchor logs to simulate query logs. The anchor log for the dataset has been processed and made publicly available by the University of Twente[5]. Each line in the log represents a document in the collection with all the anchor text pointing to the document [8]. We used the anchor log file of the ClueWeb09 Category B dataset. This file contains 43 million lines and thus contains anchor text for about 87% of the documents. Each line is tab separated and consists of the document TREC identifier, its url and all the anchor text pointing to that document.

Figure 1 shows a sample line in the anchor log file. We add quotation marks to group anchor text fields for illustration purposes.

In the next sections describing our runs, we will use the following terminology. For a query q consisting of a number of terms qt_i, our reference search engine (The Indri search engine) would return a ranked list of documents using the query likelihood model from the ClueWeb09 category B dataset:

[2] `http://boston.lti.cs.cmu.edu/Data/clueweb09/`

[3] `http://lemurproject.org/indri.php`

[4] `http://boston.lti.cs.cmu.edu:8085/clueweb09/search/cataenglish/lemur.cgi`

[5] `http://wwwhome.cs.utwente.nl/ hiemstra/2010/anchor-text-for-clueweb09-category-a.html`

```
clueweb09-en0000-23-00060 http://001yourtranslationservice.com/dtp/ 'website design' 'DTP and Web Design'
'Samples' 'programmers' 'desktop publishing' 'DTP pages' 'DTP samples' 'DTP and Web Design Samples'
'DTP and Web Design Samples' 'DTP and Web Design Samples' 'DTP and Webpage Samples' 'DTP'
http://001yourtranslationservice.com/dtp/
```

Figure 1: A sample of the anchor log file

$D_q < d_{q,1}, d_{q,2}, ..., d_{q,n} >$ where $d_{q,i}$ refers to the document ranked i for the query q based on the reference search engine standard ranking function.

4 Runs

Participants in the Session Track are asked to submit up to three runs. This year we are proposing three different runs (systems). Two of those runs are considered baseline systems to which we will compare our proposed method against. In all our runs (essex1, essex2, essex3) we will we use the Indri search engine to provide the first two ranked lists ($RL1$) and ($RL2$) by submitting the original query q and the reformulation r, i.e. the first two ranked lists will be D_q and D_r respectively. The maximum number of returned documents in both lists are limited to 1000.

We also used the Waterloo Spam Rankings[6] for the ClueWeb09 dataset to filter the spam documents from the returned ranked lists. We consider documents with scores of 70% or less as spam which is recommended by the creators of those rankings [2]. Table 1 illustrates the ranked lists matrix of our three runs. In the following subsections we explain how we produced the ranked list $RL3$ for each run.

	RL1	RL2	RL3
essex1	D_q	D_r	(baseline 1)
essex2	D_q	D_r	(baseline 2)
essex3	D_q	D_r	(AutoAdapt Approach)

Table 1: The Runs matrix

4.1 The first Baseline - essex1

This baseline represents the simplest way of using previous user interaction with the search engine to interpret reformulated queries. This is done by submitting a new query $q + r$ to our search engine where the terms in this query is the set $qt \cup rt$. i.e. the system will return the ranked list D_{q+r} as ($RL3$).

[6] http://durum0.uwaterloo.ca/clueweb09spam/

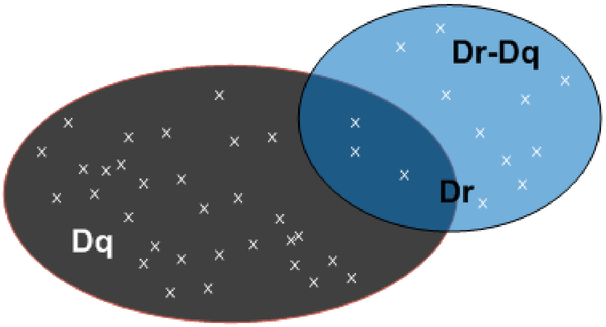

Figure 2: Illustration of the filtering process

4.2 The second Baseline - essex2

This baseline reflects on the assumption that the users are not satisfied with the first set of results and that is why they reformulated their original query. Therefore one possible naive way to utilise the previous query is to filter the results for the next query by eliminating whatever appears in the result set returned for the original (first) query. In this baseline, for the ranked list ($RL3$) we return the ranked list: $D_r - D_q = \{d; d \in D_r, d \notin D_q\}$
The documents in $D_r - D_q$ are ordered using their ranking in D_r.
Figure 2 illustrates this filtering process. Note that we are filtering the top 1000 documents returned not the entire result set.

4.3 The AutoAdapt Approach - essex3:

In this run we developed a method for extracting useful terms and phrases to expand the reformulated query in the session. Our method stems from previous work in using query logs to extract related queries and our work in the Autoadapt project to learn domain models from query logs [5], [12]. As described in the previous sections we used an anchor log constructed from the same dataset (the ClueWeb09 category B dataset) to simulate query logs. We consider all the anchor text pointing to one document as a set of queries in a user session. Following these assumptions we can derive suggestions for a user query using association rules proposed by Fonseca et al. [7]. The intersection of suggestions extracted for both queries in the session can be considered useful for query expansion of the reformulated query as they can provide an approximation of the potential user session route.
The following steps were taken for each query pair to extract the query expansion terms and phrases:

- We remove all common stop words from both queries in the session.
- From the anchor log, we extract all the lines (the sessions) in which the anchor text contains either one of the queries.

- If one of the queries is entirely contained in the other one, i.e the queries looks like XY:X or vice versa, then we treat the pair as the pair X,Y.
- Using the association rules approach for log data proposed by Fonseca et al.[7] we extract all the suggestions for each of the queries X, Y. In this step, we considered some anchor terms as stop words and filtered out these stop words from the anchor text. This is due to the observation made previously by Eiron et al. that links within the site are often navigational links and they results in anchor terms such as 'click', 'next', 'here' [6].
- We intersect the two lists of suggestions in the previous step and consider those as expansions to the reformulated query in addition to the original query.
- To generate the ranked list $RL3$, we submit the following query (using Indri operators to weight the different query terms):

$$\begin{gathered}\#\ \text{combine(}\\ 0.7\ \#\ \text{combine(}\ rt_1\, rt_2 \,..\, rt_n)\\ 0.3\ \#\ \text{combine(}\ q\ e_1\, e_2 \,..\, e_{10})\\ )\end{gathered}$$

  where rt_i are the individual terms in the reformulated query r, q is the query phrase of the original query q and e_i is an expansion term or phrase extracted as explained in the previous step. Note that in the case where no expanded terms or phrases are extracted in the previous step, we are only expanding with the original query.

  Table 2 shows some the extracted expansion terms and phrases for 3 different pairs.

Session	**Expansion terms and phrases**
gps devices → garmin	'gps devices', 'wikipedia','usb', 'gps device', 'gps products', 'garmin nuvi880', 'garmin gps device','visit garmin'
computer worms → malware	'computer worms','computer security', 'category','worm'
us geographic map → us political map	'us political map','article'

Table 2: Example of expansion terms and phrases extracted for three query pairs

5 Results

Table 3 shows the overall retrieval performance of our 3 runs and the summary results of all the participants in the track. Variants on Järvelin et al.'s [9] normalised session DCG and the standard nDCG were estimated. The figures in Column 2 represent the session normalised discounted cumulative gain $nsDCG(RL13)$ and used as the main measure for goal G2 (evaluating the performance over the entire session). Both $nsDCG(RL13)$ and $nsDCG(RL12)$ can be used to compare the performance of our runs for goal G1 (testing whether a system can improve its performance by using information from previous queries)

Run	$nsDCG.RL12$	$nsDCG.RL13$	$nDCG.RL1$	$nDCG.RL2$	$nDCG.RL3$
essex1	0.2154	0.2231	0.2077	0.2215	0.2348
essex2	0.2154	0.1993	0.2077	0.2215	0.1700
essex3	0.2154	**0.2246**	0.2077	0.2215	**0.2456**
min	0.0666	0.0458	0.0557	0.0900	0.0263
median	0.2044	0.1784	0.1894	0.2144	0.1700
max	0.2488	0.2375	0.2354	0.2658	0.2602

Table 3: The results for our runs and the overall results of the Session Track. The figures in bold are the best achieved scores in our runs for $nsDCG.RL13$ and $nDCG.RL3$.

We summarise the findings of analysing these results as follows:

- Our anchor expansion approach 'essex3' outperforms both baselines 'essex1' and 'essex2' for task2. Both 'essex3' and 'essex1' runs are among the top performing systems in the track for goal G2 as their $nsDCG(RL13)$ score is close to the maximum score reported by NIST and is above the median.

- Each of 'essex1' and 'essex3' systems has achieved a marginal overall improvement of retrieval performance for $RL13$ over $RL12$ i.e. they were both capable of using previous queries to improve retrieval performance. The anchor expansion approach 'essex3' was marginally better than 'essex1'. However both approaches did not achieve a statistically significant improvement when t-test is applied on $nsDCG@10$ for $RL12$ and $RL13$. The second baseline 'essex2' failed to improve retrieval performance when using previous queries history. Table 4 illustrates the specifics for goal G1.

- To analyse the performance with respect to the reformulation type, table 5 illustrates the results for goal G2 when considering each reformulation type. In all runs, the order of performance with regards to reformulation type is drifting, generalisation then specification. This suggests that drifting is the easiest reformulation type and specification is the hardest. In

	nsDCG@10			nDCG@10			
Run	**RL12**	→	**RL13**	**RL1**	**RL2**	→	**RL3**
essex3	0.2154	↑	0.2249	0.2077	0.2215	↑	0.2461
essex1	0.2154	↑	0.2234	0.2077	0.2215	↑	0.2353
essex2	0.2154	⇓	0.1993	0.2077	0.2215	⇓	0.1700

Table 4: The ↑ symbol denotes a measurable but not significant increase in the performance of the retrieval system when utilising the initial query compared with the performance of the system when ignoring the initial query, while the ↓ denotes a drop. The ⇓ symbol denotes a significant drop when t-test is applied

	nsDCG@10.RL13			
Run	**all sessions**	**Specification**	**Generalisation**	**Drifting**
essex3	0.2249	0.1481	0.2531	0.2763
essex1	0.2233	0.1456	0.2538	0.2738
essex2	0.1993	0.1395	0.2190	0.2416

Table 5: Results showing performance over the entire session, RL1 → RL3 for the different reformulation types

	nsDCG@10								
Run	**Specification**			**Generalisation**			**Drifting**		
	RL12	→	**RL13**	**RL12**	→	**RL13**	**RL12**	→	**RL13**
essex3	0.1563	↓	0.1481	0.2381	↑	0.2531	0.2542	↑	0.2763
essex1	0.1563	↓	0.1456	0.2381	↑	0.2538	0.2542	↑	0.2738
essex2	0.1563	⇓	0.1395	0.2381	↓	0.2190	0.2542	⇓	0.2416

Table 6: System performance per reformulation type.

Run	**All sessions**	**Specification**	**Generalisation**	**Drifting**
essex3	19.83	0.20	14.37	44.36
essex1	7.32	-13.61	12.85	23.32
essex2	-6.67	-8.42	-1.67	-9.39

Table 7: % average increase from nsDCG@10.RL12 to nsDCG@10.RL13

particular the scores obtained for specification sessions were significantly lower from the ones obtained for generalisation and drifting.

We also analyse the performance according to the reformulation type for goal G1 in table 6. For all runs, better results were achieved in drifting and generalisation and no improvement was obtained for specification sessions. Both 'essex3' and 'essex1' achieved improvement for $RL13$ over $RL12$ in drifting and generalisation sessions but not in specification. However when taking the average percentage increase into account in table 7 the anchor expansion approach 'essex3' did improve on all reformulation types including specification.

- Expansion terms could not be derived for all query pairs using our anchor log approach. When looking at individual query pairs where 'essex3' succeeded in extracting query expansions from the anchor logs, the majority of these resulted in a better retrieval performance over 'essex1' for $RL3$. In 'essex3' we successfully obtained expansions for the reformulated query in 52 topics out of the 136 judged by NIST. In 69% of those topics 'essex3' achieved a better performance than 'essex1' with regards to the first task.

6 Conclusion

This year's Session Track provided a platform to evaluate the effectiveness of Information Retrieval systems in interpreting query reformulations. The results of our runs are promising. First, they show that even a very simple way of using previous user interactions can improve retrieval performance for reformulated queries. Second, they provide evidence that using anchor logs to derive query expansions for the reformulated query can improve performance over a baseline system. Our anchor expansion approach was among the top performing systems and it has the best retrieval performance for both goals G1 and G2 when compared to the two baselines submitted. This suggests that using the anchor log as an alternative for query logs to extract query expansions for reformulated can be particularly useful to the problem of searching over sessions.

Acknowledgements

This research is part of the AutoAdapt research project. AutoAdapt is funded by EPSRC grants EP/F035357/1 and EP/F035705/1.

References

[1] TREC 2010 Session Track Guidelines. `http://ir.cis.udel.edu/sessions/guidelines.html`, July 2010.

[2] G. Cormack, M. Smucker, and C. Clarke. Efficient and effective spam filtering and re-ranking for large web datasets. *CoRR*, 2010.

[3] N. Craswell, D. Fetterly, M. Najork, S. Robertson, and E. Yilmaz. Microsoft research at trec 2009: Web and relevance feedback tracks. In *Proceedings of the 18th Text REtrieval Conference (TREC)*. NIST, 2009.

[4] V. Dang and B. W. Croft. Query reformulation using anchor text. In *WSDM '10: Proceedings of the third ACM international conference on Web search and data mining*, pages 41–50, New York, NY, USA, 2010. ACM.

[5] S. Dignum, U. Kruschwitz, M. Fasli, Y. Kim, D. Song, U. Cervino, and A. De Roeck. Incorporating Seasonality into Search Suggestions Derived

from Intranet Query Logs. In *Proceedings of the IEEE/WIC/ACM International Conferences on Web Intelligence (WI'10)*, pages 425–430, Toronto, 2010.

[6] N. Eiron and K. S. McCurley. Analysis of anchor text for web search. In *SIGIR '03: Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, pages 459–460, New York, NY, USA, 2003. ACM.

[7] B. M. Fonseca, P. B. Golgher, E. S. de Moura, and N. Ziviani. Using association rules to discover search engines related queries. In *Proceedings of the First Latin American Web Congress*, pages 66–71, 2003.

[8] D. Hiemstra and C. Hauff. Mirex: Mapreduce information retrieval experiments. Technical Report TR-CTIT-10-15, Centre for Telematics and Information Technology University of Twente, Enschede, April 2010.

[9] K. Järvelin and J. Kekäläinen. Ir evaluation methods for retrieving highly relevant documents. In *SIGIR '02: Proceedings of the 23rd ACM SIGIR Conference on Research and Development of Information Retrieval*, pages 41–48, New York, NY, USA, 2000. ACM.

[10] E. Kanoulas, P. Clough, B. Carterette, and M. Sanderson. Session Track at TREC 2010. In *Proceedings of the Workshop on the Automated Evaluation of Interactive Information Retrieval in conjunction with the 33rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 2010*, Geneva, Switzerland, 2010.

[11] M. Koolen and J. Kamps. The importance of anchor text for ad hoc search revisited. In *SIGIR '10: Proceeding of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, pages 122–129, New York, NY, USA, 2010. ACM.

[12] U. Kruschwitz, M.-D. Albakour, J. Niu, J. Leveling, N. Nanas, Y. Kim, D. Song, M. Fasli, and A. De Roeck. Moving towards Adaptive Search in Digital Libraries. In *Advanced Language Technologies for Digital Libraries*. Springer, 2011. Forthcoming.