# Webis at TREC 2013 – Session and Web Track

Matthias Hagen, Michael Völske, Jakob Gomoll, Marie Bornemann, Lene Ganschow, Florian Kneist, Abdul Hamid Sabri, and Benno Stein

Bauhaus-Universität Weimar
99421 Weimar, Germany
<first name>.<last name>@uni-weimar.de

**Abstract** In this paper we give a brief overview of the Webis group's participation in the TREC 2013 Session and Web tracks. All our runs are on the full ClueWeb12 and use the online Indri retrieval system hosted at CMU.
As for the session track, our runs implement three main ideas that were slightly improved compared to our participation in 2012: (1) distinguishing low risk sessions where we want to involve session knowledge in the form of a conservative query expansion strategy (only few expansion terms based on keywords from previous queries and seen/clicked documents/titles/snippets) from those where we don't, (2) conservative query expansion based on similar sessions from other users, (3) result list postprocessing to boost clicked documents of other users in similar sessions. As these techniques leave a lot of queries unchanged when not enough session knowledge is available, we do not expect large gains over all the sessions.
As for the Web track, our runs exploit different strategies of segmenting the queries (i.e., identifying and highlighting concepts within the query as phrases to be contained in the results). Additionally to algorithmic segmentations based on our WWW 2011 and CIKM 2012 ideas, we had one run where we chose the segmentation according to a majority vote amongst five humans. In a last run, the results are constructed so as to be disjunct from the track's baseline's and our other runs' results. Instead, we populate the result list with documents that different segmentations of the query would return top-ranked or that are deeper in the ranking for segmentations already chosen in previous runs. The underlying idea was to obtain at least some judgments for the top documents that other segmentations would bring up in their rankings. As most of the queries are rather short, we expect only slight improvements or no effect at all from the different segmentation strategies that are tailored to longer and more verbose queries.

## 1 Retrieval system

All our runs for the Session track and the Web track are on the full ClueWeb12 corpus (category A) and use the language modeling based Indri search engine provided by the Carnegie Mellon University[1].

We did not further tune any parameter settings apart from term weighting as described in the later sections. Our main research questions are whether our ideas for improving the retrieval via session knowledge or query segmentation have any positive effect with the above standard retrieval model.

[1] `http://lemurproject.org/clueweb12/services.php`

## 2 Session track

The TREC 2013 Session track in its fourth year again focused on techniques for user experience improvement during a web *search session*—the set of consecutive queries submitted for the same information need. The main assumption for the track is the following interaction scheme during such sessions. The user comes up with a set of (in her opinion) appropriate keywords—or keyphrases—for a given information need. She submits a query containing some of these keywords and gets back a ranked result list. If the user does not find a match for her information need among the first results, if some "sub"-information need remains open, or even if some new need evolved during studying the first results, she will hardly browse all the items in the ranked list of the very first query but instead submit different queries until she is satisfied or decides to give up. The idea is to use the observable interaction scheme (e.g., clicked documents and dwell times) to gain knowledge about what the user is looking for and to apply this knowledge to help improve the retrieval for the final query. The task design had three steps which increased the available knowledge of the previous interactions: (1) only the last query string given, (2) additionally given the strings of the previous queries from the session, the top-10 results with snippets for the previous queries, and clicked results and dwell times given for the previous queries of the same session (this is equivalent to RL4 from the previous years), (3) sessions from other users are given with similar data as in (2) .

### 2.1 Our approach – The high level view

With the above described increased knowledge, our framework also evolves in three steps: (1) query used as is, (2) keywords from previous queries, keyphrases from the shown snippets, titles, and the clicked documents as potential further query expansion candidates, (3) queries and clicked results from other users as further expansion candidates. The applied strategy basically is the same as in our last year's approach [HPB+12]—that used our approaches from 2010 and 2011 [HSV10,HGMS11] in a conservative way so as to not change the system's behavior when too few session knowledge is available. For example, when only two queries are available and the last one is a specialization or generalization of the previous one, we believe that not much can be learnt from such few interactions. In such cases, a low risk strategy (that we want to develop) would not change the query but probably only not show seen/clicked results again. Our goal is to develop a strategy that only interferes a user's querying process by applying session knowledge in low risk situations when the chance of harming the user's search experience is low. In high risk cases (e.g., when not much interaction information is available), the idea is to mainly trust the underlying retrieval system.

### 2.2 Research questions

Our underlying research questions are threefold. First, we want to examine the effect of distinguishing between sessions where session knowledge in form of query expansion should be applied (low risk) from those where this is not the case (high risk). Second, we want to expand with very few terms only; avoiding overlong (and thus time-consuming)

queries. Third, we want to examine whether similar sessions from other users are a better source for query expansion terms than the previous queries of the same user.

### 2.3 Our three runs

We briefly describe the ideas underlying our three runs. As for RL1, all runs simply use the query as is but for RL2 and RL3 we employ different query expansion and result list processing strategies. However, all runs also share the following query preprocessing and result list postprocessing schemes.

**Query preprocessing** For all runs, we preprocess the query strings by lowercasing all keywords, removing punctuation and double white spaces, by removing stopwords, and by replacing the keywords `wiki` or `facts` with `wikipedia` (but this probably does not have much effect this year, as only less than 20,000 Wikipedia documents are contained in the ClueWeb12). The queries are submitted segmented according to our query segmentation approaches [HPSB10,HPSB11,HPBS12].

**Result list postprocessing** Whenever a query does not return any results, we remove keywords until the result set is not empty anymore. The removal is done in a PROMISING QUERY framework style [SH11,HS11] that was also employed in our three previous Session track approaches [HSV10,HGMS11,HPB$^+$12].

Furthermore, we remove documents of more than 7,000 words length (in a study over TREC data of the past years these were mostly judged as spam), we remove duplicate documents from lower ranks if at some higher rank a document has the cosine similarity ($tf$-weights) of 1- and 5-grams is above 0.98 (facing the risk that duplicates of relevant results are not shown but which would not be desirable in the real world), and by removing spam documents with a threshold of 30 according to the Waterloo spam ranking[2] (that threshold has worked quite well in the previous years).

**Run webisS1** In this run, we use a hybrid Indri-optimized query segmentation [HPBS12]. As for RL2, we use a query type classification scheme. In case that the last query is a new query or a generalization of a previous query, we extract one keyphrase from the set of the previous queries, one from the clicked documents, one from their titles, and one from the shown snippets. These are added to the current query. The keyphrase extraction is done as described in our last year's approach [HPB$^+$12]. Terms from the current query (we use segmentation again) get a weight of 2.0, terms from previous queries a weight of 0.6, terms from documents get a weight of 0.2, and terms from snippets or titles get a weight of 0.1. In case of repeated queries (even with changed word order), we do not show the documents that were shown for previous queries and we remove results when they contain more than two of the top-10 extracted keyphrases from previously shown documents.

As for RL3, we run our session detection scheme [HGBS13] and use for the above described expansion only sessions from other users with a cosine similarity ($tf$-weights)

[2] `http://www.mansci.uwaterloo.ca/~msmucker/cw12spam/`

of the query strings of more than 0.35. Furthermore, at most three clicked documents from these sessions are included at ranks 1–3 (ranked by the number of clicks they received) if they got at least two clicks.

**Run webisS2** In this run, we use the hybrid query segmentation strategy that is optimized for accuracy against human segmentations [HPBS12]. The keyphrase extraction for RL2 and RL3 is similar to the webisS1 run except that only the last clicked documents of each fitting previous query are used (i.e., assuming another click-relevance model). As for result list postprocessing in case of RL3 (adding at most three clicked documents from other sessions), we also only consider the last clicks of queries from previous interactions.

**Run webisS3** In this run, we use the naive query segmentation strategy [HPSB10]. As for RL2, previously clicked documents from the same session are used to populate the top of the ranking. The underlying idea is to obtain relevance judgments for clicked documents. If there are more than 10, the rest is used in RL3 to populate its top results. Furthermore, in RL3 we then also use clicked documents from related sessions if not enough clicks from the same session can be found.

## 3 Web track

The TREC 2013 Web track had risk-aware ranking improvement as its underlying topic. Given a baseline ranking, the task was to improve it to achieve some gain while decreased performance will be punished more than an improvement of the same scale. Hence, decreased performance bears a high risk of low performance.

### 3.1 Our approach – A high level view

We simply apply several of our different query segmentation strategies [HPSB10,HPSB11,HPBS12] but nothing else. Our main goal is to be able to compare different segmentation strategies based on the TREC judgments. However, we plan to do this after TREC and thus chose the ranked lists so as to maximize the number of distinct judged documents. Hence, if a top-10 document from some ranked list already was present in the ranked list of some previous ranked list, we do not list the result again and instead remove it from the ranked list. The underlying idea is to obtain relevance judgments for more documents that can then be used after TREC for deeper comparisons. But this comes at the expense of being really comparable in the original TREC track as for instance all the results of the baseline system are assumed to already be in the judgment pool. Hence, our submitted result lists do not include these results again, even if a segmented query would have produced them in a reordered ranking.

### 3.2 Our runs

Each of our runs employs a different segmentation strategy. But note again that the results submitted for judgment may not have been the real results the retrieval system

would have produced on the segmented query as we avoid double submission of the same documents.

**Run webishybrid** This run uses hybrid Indri-optimized query segmentation [HPBS12]. Top-10 results from the official TREC Web track's baseline ranked list are removed from the ranked list.

**Run webiswikibased** This run uses Wikipedia-based query segmentation [HPSB11]. Top-10 results from the official TREC Web track's baseline ranked list and our webishybrid run are removed from the ranked list.

**Run webisnaive** This run uses naïve query segmentation [HPSB10]. Top-10 results from the official TREC Web track's baseline ranked list and our webishybrid and webiswikibased runs are removed from the ranked list.

**Run webismixed (a potential manual run)** This run uses query segmentations according to the majority vote amongst seven human annotators and may thus be seen as a manual run. Top-10 results from the official TREC Web track's baseline ranked list and our above runs are removed from the ranked list.

**Run webiswtbaseline** This run uses WT-baseline query segmentation [HPBS12]. Top-10 results from the official TREC Web track's baseline ranked list and our above runs are removed from the ranked list.

**Run webisrandom** This run uses all the potentially possible other segmentations of the given query in order to also obtain judgments for the top documents from segmentations not chose in our previous runs. From these other segmentations, the top results are chosen that are not within the top-10 results from the official TREC Web track's baseline ranked list or our above runs. If there are no other segmentations, non-top-10 results from the previous runs are used to populate the top-10.

## 4 Discussion

As can be seen from the evaluation in the tracks' overview papers, our runs usually perform worse than the median of all runs. This is not too surprising as most of our runs tried to maximize the number of judgments and left out potentially relevant results whenever these are already present in a previous run. In a post-TREC evaluation we will use the judgments to test the original unchanged rankings of different segmentations and we will try to evaluate relevance of the clicked documents and the effect of using these documents or keywords from them to improve other users' session experience.

## References


[HGBS13] Matthias Hagen, Jakob Gomoll, Anna Beyer, and Benno Stein. From Search Session Detection to Search Mission Detection. In *10th International Conference Open Research Areas in Information Retrieval (OAIR 13)*, pages 85–92. ACM, May 2013.

[HGMS11] Matthias Hagen, Jan Graßegger, Maximilian Michel, and Benno Stein. Webis at the TREC 2011 Sessions Track. In Ellen M. Voorhees and Lori P. Buckland, editors, *20th International Text Retrieval Conference (TREC 11)*. National Institute of Standards and Technology (NIST), November 2011.

[HPB+12] Matthias Hagen, Martin Potthast, Matthias Busse, Jakob Gomoll, Jannis Harder, and Benno Stein. Webis at the TREC 2012 Sessions Track. In Ellen M. Voorhees and Lori P. Buckland, editors, *21st International Text Retrieval Conference (TREC 12)*. National Institute of Standards and Technology (NIST), November 2012.

[HPBS12] Matthias Hagen, Martin Potthast, Anna Beyer, and Benno Stein. Towards Optimum Query Segmentation: In Doubt Without. In Xuewen Chen, Guy Lebanon, Haixun Wang, and Mohammed J. Zaki, editors, *21st ACM International Conference on Information and Knowledge Management (CIKM 12)*, pages 1015–1024. ACM, October 2012.

[HPSB10] Matthias Hagen, Martin Potthast, Benno Stein, and Christof Bräutigam. The Power of Naïve Query Segmentation. In Fabio Crestani, Stéphane Marchand-Maillet, Hsin-Hsi Chen, Efthimis N. Efthimiadis, and Jacques Savoy, editors, *33rd International ACM Conference on Research and Development in Information Retrieval (SIGIR 10)*, pages 797–798. ACM, July 2010.

[HPSB11] Matthias Hagen, Martin Potthast, Benno Stein, and Christof Bräutigam. Query Segmentation Revisited. In Sadagopan Srinivasan, Krithi Ramamritham, Arun Kumar, M. P. Ravindra, Elisa Bertino, and Ravi Kumar, editors, *20th International Conference on World Wide Web (WWW 11)*, pages 97–106. ACM, March 2011.

[HS11] Matthias Hagen and Benno Stein. Applying the User-over-Ranking Hypothesis to Query Formulation. In *Advances in Information Retrieval Theory. 3rd International Conference on the Theory of Information Retrieval (ICTIR 11)*, volume 6931 of *Lecture Notes in Computer Science*, pages 225–237, Berlin Heidelberg New York, 2011. Springer.

[HSV10] Matthias Hagen, Benno Stein, and Michael Völske. Webis at the TREC 2010 Sessions Track. In Ellen M. Voorhees and Lori P. Buckland, editors, *19th International Text Retrieval Conference (TREC 10)*. National Institute of Standards and Technology (NIST), November 2010.

[SH11] Benno Stein and Matthias Hagen. Introducing the User-over-Ranking Hypothesis. In *Advances in Information Retrieval. 33rd European Conference on IR Resarch (ECIR 11)*, volume 6611 of *Lecture Notes in Computer Science*, pages 503–509, Berlin Heidelberg New York, April 2011. Springer.