Microsoft Research at TREC 2010 Web Track

Nick Craswell, Dennis Fetterly, Marc Najork

Microsoft

Abstract

This paper describes our entry into the TREC 2010 Web track. We extracted and ranked results for both last year's and this year's topics from the ClueWeb09 corpus using a parallel processing pipeline that avoids the generation of an inverted file. We describe the components of the parallel architecture and the pipeline and how we ran the TREC experiments, and we present effectiveness results.

1 Introduction

This paper describes our entry into the TREC 2010 Web track. Our team comprised members from Bing, Microsoft's search engine, and Microsoft Research, Silicon Valley. We used the DryadLINQ data-parallel processing system [6] on a cluster of about 220 machines to process the half-billion page English-language subset of the ClueWeb09 collection. We also used the Scalable Hyperlink Store [4] running on a cluster of 12 machines to compute SALSA scores for each topic. We computed five features (described below) for each candidate result and used a weighted linear combination to blend these features. We performed three runs, and submitted each run to both the ad-hoc task and the diversity task.

2 Processing Pipeline

The processing pipeline we used to prepare this year's entry consisted of the following steps:

- **Parsing:** We used DryadLINQ to parse the HTML web pages, tokenizing them into individual words and hyperlinks. We associated each title and content word occurrence with the document containing it; in addition, we associated each anchor word occurrence with the document referenced by the anchor's hyperlink.

- **Document Frequencies:** We computed document frequencies for the union of all terms of the 100 (training and test) queries we were considering.

- **BM25F score:** We computed a BM25F score for each query and each document in the ClueWeb09 collection, retaining the top 5000 results per query.

- **Span score:** Still in DryadLINQ, we computed the size s (in words) of the minimum window in the document that would cover all of the terms in the query. For this feature, we did not consider anchor text terms to be part of the document. The "span score" is $1/(s-q+1)$, where q is the length (in words) of the query. If any of the terms occurred only in anchors but not the document itself, the span score defaults to 1.

- **SALSA score:** Furthermore, we used SHS to run SALSA-SETR [5] on the top 5000 results of each query. SALSA-SETR is a variant of Lempel & Moran's SALSA algorithm [3].

- **Matching anchor count (MAC):** We resolved all the symbolic hostnames in ClueWeb09 page and link URLs into IP addresses. Then, using DryadLINQ, we identified unique $\langle s, t, a\rangle$ triples, where s is the first three octets of the IP address of a document, t is the target URL of a link within that document and a is the anchor text of that link. Then we built the MAC ranking feature for each query-target pair $\langle q, t\rangle$, counting the number of source IPs that link to target t with anchor $a = q$.

- **Extraction:** For each query we identified a pool of documents and collated our ranking features. The resulting "extraction" file could be used for training or for generating submitted runs.
- **Training:** We combined evidence from the various features by applying ad-hoc transformations to each feature (identity or log) and then performing a weighted linear combination of the transformed features. We trained the weights using an extraction of the fifty 2009 topics and their official judgments.
- **Runs:** Using an extraction of the fifty 2010 topics, we ran our trained model to produce our submitted runs.

This pipeline differed from the approach we took last year [2] in the following ways:

- **Query semantics:** In our 2009 entry, we assumed a disjunctive semantics for multi-term queries, i.e. we would consider documents containing any of the query terms as result candidates. This year, we assumed a conjunctive semantics, i.e. we considered only documents containing all of the query terms as result candidates. Note that both years we considered the anchor text of hyperlinks pointing to a document to be part of the document.
- **Omitted and added features:** Our 2009 entry employed two query-independent link-based features, namely inter-domain in-degree and PageRank; our 2010 entry does not. Conversely, our 2010 entry employed two new features: The "spam score" provided by the University of Waterloo[1], and a "span score" feature quantifying the length (in words) of the shortest window in the result document containing all the query terms.
- **Training:** Both our 2009 and 2010 entries combined individual features into an overall score using weighted linear combination. However, in 2009 we trained the weights based on our own judgments of 118 results from a search engine log; whereas in 2010 we trained based on the TREC 2009 diversity judgments. We chose the diversity judgments because they contain navigational subtopics, and this year both diversity and adhoc reward good navigational performance.

3 Web track submissions

We submitted three runs to both the ad-hoc task, and the diversity task. In all three runs the ranker is a simple linear combination of features. The following table shows the features, weights and transformations used in each run:

Feature	BM25F	MAC	SpamScore-Fusion	SALSA-SETR	Span score
Transformation	identity	log	log	log	identity
msrsv1	1	15	5	100	16
msrsv2	1	15	5	100	16
msrsv3	1	15	5	100	0

Runs msrsv1 and msrsv2 differed only in the way that MAC scores were computed: For msrsv1, we directly matched the anchor URL strings with page URL strings. For msrsv2 we first converted URL strings into URI objects (thereby normalizing escape sequences) and then matched anchor URIs with page URIs.

The next table shows the performance of our three runs, according to a variety of different performance measures. These statistics are based on 48 topics for which judgments are available. Based on these numbers and the numbers in the Web Track overview, we can draw two observations: First, msrsv3 performed better than msrsv1 and msrsv2. In other words, the "span score" feature we introduced in this year's entry apparently did not improve performance. Second, relative to other groups we performed better on metrics that reward finding a navigational result near the top of the ranking, and worse on metrics that flatten such distinctions such as MAP. Good navigational performance may be consistent with our use of training data with navigational subtopics and our focus on link-based ranking features.

runid	ERR@20	nDCG@20	P@20	MAP
msrsv1	0.160	0.229	0.347	0.081
msrsv2	0.164	0.235	0.352	0.083
msrsv3	0.166	0.237	0.344	0.082

Table 1: Adhoc task results.

runid	ERR-IA@20	alpha-nDCG@20	NRBP	MAP-IA
msrsv1	0.338	0.485	0.292	0.066
msrsv2	0.338	0.483	0.292	0.066
msrsv3	0.347	0.491	0.303	0.068

Table 2: Diversity task results.

4 Conclusions

In preparing our entry this year, we utilized the same computational infrastructure we employed the previous year: the DryadLINQ data-parallel processing platform and the Scalable Hyperlink Store. We moved to a conjunctive query semantics (following the lead of virtually all commercial search engines), eliminated two of the three link-based features we used last year and instead introduced two new features (one of them provided by the University of Waterloo), and used a smaller but standard data set for training. According to our performance results, one of our two new features ("span score") turned out to be counterproductive. Our best-performing run msrsv3 incorporates only four features into the overall score. According to the overview of the TREC 2010 Web Track and the primary effectiveness measures of the TREC 2010 Web Track (ERR@20 for the adhoc task and ERR-IA@20 for the diversity task), msrsv3 outperformed all competing runs.

References

[1] G.V. Cormack, M.D. Smucker, C.L.A. Clarke. Efficient and effective spam filtering and re-ranking for large web datasets. Online at `http://arxiv.org/abs/1004.5168`, 2010.

[2] N. Craswell, D. Fetterly, M. Najork, S. Robertson, E. Yilmaz. Microsoft Research at TREC 2009 – Web and Relevance Feedback Tracks. In *Proc. of the 18th Text Retrieval Conference*, 2009.

[3] R. Lempel and S. Moran. The stochastic approach for link-structure analysis (SALSA) and the TKC effect. *Computer Networks and ISDN Systems*, 33(1–6):387–401, 2000.

[4] M. Najork. The scalable hyperlink store. In *Proc of the 20th ACM Conference on Hypertext and Hypermedia*, pages 89–98, 2009.

[5] M. Najork, S. Gollapudi, and R. Panigrahy. Less is More: Sampling the neighborhood graph makes SALSA better and faster. In *Proc of the 2nd ACM International Conference on Web Search and Data Mining*, pages 242–251, 2009.

[6] Y. Yu, M. Isard, D. Fetterly, M. Budiu, Ú. Erlingsson, P. K. Gunda, J. Currey. DryadLINQ: a system for general-purpose distributed data-parallel computing using a high-level language. In *Proc. of the 8th USENIX Symposium on Operating Systems Design and Implementation*, pages 1–14, 2008.