A non-functional prototype at TREC 2003

Brian D. Davison, Wei Zhang, and Josh Miller
Department of Computer Science & Engineering
Lehigh University, Bethlehem, PA 18015
{davison,wei,jamr}@lehigh.edu

Overview

As a first attempt at participation in the TREC competition, we built a system which produced some preliminary results, but was unable to generate the quality of results that we expected. While we were able to submit four base-line runs, bugs were discovered in the final hours before the deadline making it impossible to submit results using our intended implementation. We have since found additional coding errors, making our submitted results expectedly poor.

The size of our index dataset was approximately 3.8GB without compression. We did not use term position information nor any kind of phrasal indexing.

Topic distillation task

We submitted two runs for topic distillation. They employed both vector space and simple popularity-based link analysis techniques. Queries were down-cased and stop words were removed before ranking.

Term weights (both for terms in the main document, as well as terms in anchor text) were calculated as the $\log_{10}$ of (termfreq + 1).

For the `03wume206` run, the final document score was calculated as follows:

$$\begin{aligned} \text{docs[i].score} = & \log_{10}(\text{docs[i].termweight}+1) \\ & + \log_{10}(\text{docs[i].anchorweight}+1) \\ & + \text{docs[i].rlinkweight;} \end{aligned}$$

where docs is an array of documents found to contain the queries, termweight is the number of times the keywords appear in this document, anchorweight is the number of URLs that contain query terms and link to the document, and rlinkweight stands for reverse link weight, which records how many other documents link to this page. Term and anchor weights are not normalized, but the reverse link weight is normalized by dividing by the sum of all incoming links to any document in the relevant set.

The `03wume359` run employed some slightly more sophisticated approaches. We used a different term weighting approach — a variant of Salton and Buckley's method [1], and a more subtle approach for calculating link weights. The final score still followed the equation above, but the term weight portion was calculated as (0.5 + (0.5 * termfreq)) * $\log_{10}$(docs/termdocs) where docs is the number of all documents containing at least one query term, termdocs is the number of all documents containing this term. Additionally, instead of simply counting the number of incoming links, rlinkweight was defined as the number of incoming links from this relevant subset divided by the total number of incoming links to this page. In this way we hoped to emphasize pages that were predominantly cited within this query topic.

Navigational task

We did not attempt a different approach for the mixed homepage and named page queries. All queries were treated in the same way as in topic distillation. These runs only employed vector space and anchor text. To obtain term weight and anchor weights, the same algorithm was used as in topic distillation. The only difference was a 20% reduction for standard term weights in the `03wume296` run.

`03wume296`: docs[i].score = $\log_{10}$(docs[i].termweight+1) * 0.8
+ $\log_{10}$(docs[i].anchorweight+1)

`03wume298`: docs[i].score = $\log_{10}$(docs[i].termweight+1)
+ $\log_{10}$(docs[i].anchorweight+1)

Results after bug fixes

After fixing a number of bugs (after the competition was complete), but without changing the logic, we re-ran our system on both tasks. The performance metrics of the original and corrected system are shown in Table 1. The corrections almost tripled our system's performance on the navigational task, and improved performance on the topic distillation task by approximately 60%.

While the relative score improvement was large for the navigational task, the overall performance was still low, and would only change our relative ranking by a couple of positions (assuming all others stayed the same). In contrast, the smaller relative improvement in the topic distillation translates to a movement of 16 positions in the system rankings.

Topic distillation task

Rank	R-Prec	MAP	P@10	Group	Run	D	A	L
(70)	0.0636	0.0517	0.0380	lehighu	03wume206corrected	-	A	L
86.	0.0395	0.0343	0.0280	lehighu	03wume206	-	A	L
(89)	0.0357	0.0295	0.0160	lehighu	03wume359corrected	-	A	L
91.	0.0204	0.0225	0.0180	lehighu	03wume359	-	A	L

Navigational task

Rank	MRR	S@10	Group	Run	D	A	L
(69)	0.189	28.0	lehighu	03wume298corrected	-	A	-
71.	0.067	9.3	lehighu	03wume298	-	A	-
73.	0.065	8.7	lehighu	03wume296	-	A	-

Table 1: Original and corrected scores for topic distillation and navigational tasks.

In the end, however, while all improvements are welcome, the corrected scores are still not particularly competitive, and point to the need for fundamentally better algorithms.

Conclusion

Even after coding errors were corrected, the performance of this simplistic implementation was not competitive. However, it does provide a foundation on which we expect future work to build.

References

[1] G. Salton and C. Buckley. Term-weighting approaches in automatic text retrieval. *Information Processing and Management*, 24(5):513–523, 1998.