The University of Amsterdam at the TREC 2006 Terabyte Track

Jaap Kamps[1,2]

[1] ISLA, Informatics Institute, University of Amsterdam
[2] Archives and Information Studies, Faculty of Humanities, University of Amsterdam

Abstract: As part of the TREC 2006 Terabyte track, we conducted a range of experiments investigating the effects of larger test collections for both adhoc and known-item topics. In this paper, we document our official submissions to the TREC 2006 Terabyte track and conduct a number of more extensive experiments. First, we look at the amount of smoothing required for large-scale collections. Second, we investigate the relative effectiveness of various web-centric document representations based on document-text, incoming anchor-texts, and page titles. Third, we study the relative effectiveness of various query representations, both short and verbose statements of the topic of request, plus an intermediate query based on the most characteristic terms in the whole topic statement.

1 Introduction

As part of the TREC 2005 Terabyte track, we conducted a range of experiments investigating the effects of larger collections. We submitted runs for two of the Terabyte track's tasks: the adhoc task, and the named page finding task. In addition to the submitted runs, we also discuss post-submission results for the efficiency task. Furthermore, we discuss a range of more extensive experiments that investigate i) the amount of smoothing required for terabyte-scale collections; ii) the relative effectiveness of various web-centric document representations based on document-text, incoming anchor-texts, and page titles; and iii) the relative effectiveness of various query representations, both short and verbose statements of the topic of request, plus an intermediate query based on the most characteristic terms in the whole topic statement.

The rest of this paper is organized as follows. In Section 2, we detail the experimental set-up for the two tasks in the Terabyte track. In Section 3, we discuss our results, broken down over the adhoc task (§3.1) and the named page finding task (§3.2). In Section 4, we zoom in on a set of experiments on smoothing (§4.1), document representations (§4.2), and query representations (§4.3). Finally, we summarize our findings in Section 5.

2 Experimental Set-up

2.1 Retrieval set-up

Our retrieval system is based on the Lucene engine with a number of home-grown extensions [2, 7].

Indexes The Terabyte track uses the `GOV2` test collection, containing 25,205,178 documents (426 Gb uncompressed). The indexing approach is similar to our earlier experiments in the TREC Web and Terabyte tracks [4, 5, 6]. We created four separate indexes for

Full-text the full textual content of the documents (covering the whole collection);

Titles the text in the title tags of each document, if present (covering 86% of the collection);

Anchors the anchor-texts pointing toward the document ignoring relative links and extracting only full explicit URLs (covering 6.5% of the collection);

All anchors another anchor-texts index in which we unfold all relative links (covering 49% of the collection).

The difference between the two anchor text indexes is that the second index includes far more within-site links. In both cases, we normalized the URLs, and did not index repeated occurrences of the same anchor-text. As to tokenization, we removed HTML-tags, punctuation marks, applied case-folding, and mapped marked characters into the unmarked tokens. We used the Snowball stemming algorithm [8].

The main full document text index was created as a single, non-distributed index. The size of our full-text index is 61 Gb. Building the full-text index (including all further processing) took a massive 15 days, 6 hours, and 21 minutes.

Query representations We experimented with a variety of query representations. The main goal of the richer query representations was to target relevant pages that may not be retrieved by the standard short topic statement.

T Our first query representation is based on the short topic statement in the title field. This is the realistic approximation of end user request on current Internet search engines.

Table 1: Query representations for adhoc topic 701.

T	U.S. oil industry history
TDN	U.S. oil industry history the history of the U.S. oil industry Relevant documents will include those on historical exploration and drilling as well as history of regulatory bodies. Relevant are history of the oil industry in various states, even if drilling began in 1950 or later.
TDN10	history oil industry drilling u later bodies exploration began 1950
TDN10r	history history history history oil oil oil industry industry industry drilling drilling u u later bodies exploration began 1950

TDN By including all the fields of the topic—title, description, and narrative—we obtain a much more verbose statement of the information need.

TDN10 The verbose statement also contains generic stopwords (like function words), or specific phrases related to the search procedure (like "find documents that"). Hence, we decide to include only those terms that are most characteristic for a single topic, with reference to the whole topic set. That is, the terms that best distinguish the topic at hand from the other topics in the topic set. For this we use a variant of the parsimonious language modeling techniques [1], and create a query by selecting the 10 terms that are most characteristic for the topic.

TDN10r The repeated occurrence of the same term in the topic may be an important indicator of its relevance. In order to boost these terms we create an alternative query, with the same 10 terms, but now each term is repeated as often as it occurs in the entire topic statement.

Table 1 shows examples of the four different queries. All queries were further processed analogous to the documents.

Retrieval model For ranking, we work within the language modeling framework. Our language model is an extension to Lucene [2], i.e., for a collection D, document d and query q:

$$P(d|q) = P(d) \cdot \prod_{t \in q} ((1-\lambda) \cdot P(t|D) + \lambda \cdot P(t|d)),$$

where $P(t|d) = \frac{tf_{t,d}}{|d|}$, $P(t|D) = \frac{\text{doc\_freq}(t,D)}{\sum_{t' \in D} \text{doc\_freq}(t',D)}$, and $P(d) = \frac{|d|}{\sum_{d' \in D} |d'|}$. The standard value for the smoothing parameter λ is 0.15. In last year's TREC Terabyte track, we found out that the GOV2 collection requires substantially less smoothing [4]. That is, a value of λ close to 1.0. We use a standard length prior.

2.2 Official runs

We submitted nine runs in total. For the *adhoc task*, we submitted five runs. We submitted a full-text index run:

UAmsT06aTeLM Language model (λ = 0.90) on the full-text index, using only the short topic statement in the title.

Next, we submitted a plain anchor-text index run:

UAmsT06aAnLM Language model (λ = 0.90) on the anchor-text index containing only explicitly spelled-out URLs, using only the short topic statement in the title.

Since the anchor-texts provide a document representation completely disjoint from the document's text, it is of interest to investigate how different both sets of retrieved documents are. Hence, we also submitted a run that combines different sources of evidence:

UAmsT06a3SUM Weighted CombSUM of language model (λ = 0.90) runs on the full-text index (relative weight 0.8), anchor-text index (relative weight 0.1), and titles index (relative weight 0.1), all using only the short topic statement in the title.

Since the short title statement is a relatively poor representation of the underlying (pseudo) information need, we also experimented with different representations of the query.

UAmsT06aTDN Language model (λ = 0.70) on the full-text index, using a query based on all three fields of the topic statement. The query consists of the 10 most significant terms in the topic statement, where each of these 10 terms is repeated as often as it occurs.

UAmsT06aTTDN Unweighted CombSUM combination of UAmsT06aTeLM and UAmsT06aTDN.

For the *named page finding task*, we submitted four runs all using only the short topic statement in the title. We submitted a plain language model run on the full-text index:

UAmsT06nTeLM Language model (λ = 0.90) on the full-text index.

Next, we submitted a plain anchor-text index run:

UAmsT06nAnLM Language model (λ = 0.90) on the larger anchor-text index containing both relative and explicitly spelled-out URLs.

And, similar to the Ad hoc Task, we also submitted a run that combines different sources of evidence:

UAmsT06n3SUM Weighted CombSUM of language model (λ = 0.90) runs on the full-text index (relative weight 0.8), anchor-text index (relative weight 0.1), and titles index (relative weight 0.1).

We also experimented with a web-centric prior that assumes that pages with shorter URLs are more likely to be relevant [3]:

UAmsT06nTurl Language model (λ = 0.90) on the full-text index, with a URL prior instead of the standard length prior.

Table 2: Results for the adhoc task over the 50 new topics: (top half) title-only runs, (bottom half) verbose topic statement runs.

UAmsT06	Topic	map	bpref	infAP	P@10
...aTeLM	T	**0.2958**	**0.3528**	**0.2363**	**0.5260**
...aAnLM	T	0.0143	0.0336	0.0081	0.1340
...a3SUM	T	0.2759	0.3273	0.1982	0.5060
...aTDN	TDN	0.2848	**0.3879**	**0.2446**	0.5020
...aTTDN	TDN	**0.3284**	0.3837	0.2379	**0.5740**

Table 3: Results for the named page finding task.

UAmsT06	MRR	S@1	S@5	S@10	not found
...nTeLM	0.262	33/18.2%	58/32.0%	72/39.8%	**43**/23.8%
...nAnLM	0.218	29/16.0%	52/28.7%	58/32.0%	95/52.5%
...n3SUM	**0.363**	**49**/27.1%	**85**/47.0%	**100**/55.2%	**43**/23.8%
...nTurl	0.241	26/14.4%	64/35.4%	75/41.4%	44/24.3%

We calculated the number of components in the domain and file path of the URL, e.g, trec.nist.gov/act_part/act_part.html has 3 (domain) plus 2 (file path) components. Since our implementation of the language model calculates the logs of the probabilities, we took the exponent of the retrieval score, and multiplied it with the reciprocal of the length of the URL.

3 Results

3.1 Adhoc task

The topic set contains the combined of 2004 (topic numbers 701–750); 2005 (topic numbers 751–850); and 2006 (topic numbers 801–850). We look here only at the 50 "fresh" topics of 2006. The number of relevant documents per topic varies from 5 to 571, with a mean of 118 and a median 87.

Table 2 shows the results for the adhoc task. Let us first focus on the short topic statement in the title-fields of the topics. Here, the run using the massive full-text index (UAmsT06aTeLM) clearly outperforms the run on the anchor-text index (UAmsT06aAnLM). The anchor text index seems to be of some use in the first 10 ranks. For the runs using the verbose topic statement, we see that the UAmsT06aTDN run outperforms the T-only run (UAmsT06aTeLM) on the bpref and infAP measures, but loses out on the map and P@10 measures. The combination of these two runs (UAmsT06aTTDN) is improving over the T-only run on all measures, but is no equivocal improvement over the verbose run alone.

3.2 Named page finding task

In total there are 181 named page finding topics numbered 901–1081. The minimal number of relevant documents per topic is 1 and the maximum is 257. For 138 topics there is a unique relevant page, there are 7 topics with 10 or more relevant pages (caused by page-duplicates in the collection). This leads to a skewed distribution with a mean of 4.5 and a median of 1 relevant page. Table 3 shows the results for the named page finding task. We make a number of observations. First, although runs using the full-text index outperform runs using the anchor-text index on all measures, the anchor-text runs turn out to be fairly competitive, with 4 less topics solved at rank 1, and 6 less topics solved at rank 5. Second, the combination run, based on the full-text index, the anchor-text index, and a titles index, comfortably outperforms runs based on only the full-text index. The success of the combination run shows the value of different document representations. Third, the URL prior leads to mixed results: a loss of mean reciprocal rank, but a gain in the number of topics with the relevant page in the top 5 and the top 10. Finally, the overall performance is, with the targeted page in the top 3 on average, quite impressive. More worrying though is that the performance is not equally good for all topics: at rank 10, no targeted page is found for 45% of the topics, and at rank 1,000, there are still more than 20% of the topics unsatisfied. There appears to be room for further improvements.

4 Additional Experiments

We now discuss a number of additional experiments on smoothing, different document representations, and different query representations.

4.1 Smoothing

In the language modeling framework, smoothing plays an important role: it helps to overcome data-sparseness, it introduces an inverted document frequency effect, and it expresses the relative importance of query terms [9]. In practice, smoothing is also a handle to tune a run toward recall (much smoothing) or precision (little smoothing). At last year's edition of the TREC Terabyte track, we observed that our runs required very little smoothing. We redo the smoothing experiments on the Terabyte 2006 data, focusing on varying the smoothing parameter in linear or Jelinek-Mercer smoothing.

4.1.1 Named page finding task

First, we focus on the named page finding task. Since finding a 'unique' page requires precision rather than recall, we may expect a relatively high value for the smoothing parameter. Table 4 shows the results while varying the smoothing parameter over the interval between 0 and 1. We make a few observations. As expected, we see that the named page finding topics do not require much smoothing. In fact, as long as we put some weight on the collection model, the less smoothing the better.

4.1.2 Adhoc task

Next, we focus on the adhoc task. Since adhoc topics require a delicate balance between precision and recall, the standard is to use a relatively low value for the smoothing parameter (i.e., $\lambda = 0.15$). Table 5 shows the results while varying the smoothing parameter over the interval between 0 and 1. On

Table 4: Smoothing for the named page finding task using the full-text index.

λ	MRR	S@1	S@5	S@10	not found
0.0	0.0002	0/ 0.0%	0/ 0.0%	0/ 0.0%	178/98.3%
0.1	0.0877	10/ 5.5%	22/12.2%	27/14.9%	115/63.5%
0.2	0.1434	19/10.5%	32/17.7%	38/21.0%	89/49.2%
0.3	0.1681	23/12.7%	36/19.9%	42/23.2%	71/39.2%
0.4	0.1902	26/14.4%	40/22.1%	49/27.1%	62/34.3%
0.5	0.2061	28/15.5%	44/24.3%	53/29.3%	56/30.9%
0.6	0.2242	29/16.0%	49/27.1%	60/33.1%	52/28.7%
0.7	0.2368	32/17.7%	50/27.6%	62/34.3%	45/24.9%
0.8	0.2463	**33**/18.2%	52/28.7%	68/37.6%	45/24.9%
0.9	**0.2616**	**33**/18.2%	58/32.0%	**72**/39.8%	**43**/23.8%
1.0	0.2534	32/17.7%	**60**/33.1%	68/37.6%	48/26.5%

Table 5: Smoothing for the adhoc task using the full-text index.

λ	MAP	B-Pref	P@1	P@5	P@10
0.0	0.0001	0.0036	0.0007	0.0000	0.0000
0.1	0.0950	0.1760	0.5607	0.3320	0.2920
0.2	0.1502	0.2414	0.6043	0.3760	0.3520
0.3	0.1824	0.2665	0.6330	0.4240	0.3860
0.4	0.2034	0.2811	0.6762	0.4520	0.4200
0.5	0.2221	0.2954	0.7066	0.4920	0.4600
0.6	0.2404	0.3067	**0.7227**	0.5280	0.4820
0.7	0.2571	0.3179	0.7012	0.5320	0.4980
0.8	0.2737	0.3290	0.7206	0.5480	0.5140
0.9	0.2878	0.3402	0.7225	**0.5440**	**0.5260**
1.0	**0.2903**	**0.3474**	0.7192	**0.5440**	**0.5260**

the large scale GOV2 collection, we see that also for adhoc retrieval the performance increases if we apply less smoothing. Hence our experiments confirm our findings of last year: the adhoc task evaluated by average precision seems to behave very much like an early precision task.

4.2 Document representations

We experiment with the four different document representations introduced in Section 2:

Full-text All textual content of the documents;

Anchors Incoming anchor-texts based on only fully explicit URLs in the collection;

All anchors Incoming anchor-texts based on both absolute and relative links in the collection;

Title Content of the title field of the documents, if present.

All runs use little smoothing ($\lambda = 0.9$).

4.2.1 Adhoc task

We run the adhoc topics on all four indexes. The runs using the **Full-text** (UAmsT06aTeLM) and **Anchors** (UAmsT06aAnLM) indexes were also official submissions. We also include the three-way combination of **Full-text**, **Titles**, and **Anchors** (official submission UAmsT06a3SUM), and a variant using the other **All anchors** index.

Table 6 shows the results for the adhoc task. We see that runs on the full-text index outperform all other runs on the other indexes, and all combinations with runs on other indexes. Only in terms of early precision, the alternative representation perform to a certain degree. The performance at early ranks is still much inferior to the full-text index, but—considering that they are substantially smaller—the anchor and title indexes offer reasonable "value-for-money."

Table 6: Results for the adhoc task over the 50 new topics (over 1,000 retrieved results).

	map	bpref	P@1	P@5	P@10
1.Full-text	**0.2878**	**0.3402**	0.7225	**0.5440**	**0.5260**
2.Anchors	0.0142	0.0289	0.4348	0.1720	0.1340
3.All anchors	0.0306	0.0727	0.5164	0.2520	0.2160
4.Titles	0.0354	0.0942	0.4698	0.2400	0.1980
1+2+4	0.2759	0.3273	0.7609	0.5080	0.5060
1+3+4	0.2761	0.3297	**0.7623**	0.4960	0.4920

Table 7: Results for the named page finding task.

	MRR	S@1	S@5	S@10	not found
1.Full-text	0.262	33/18.2%	58/32.0%	72/39.8%	**43**/23.8%
2.Anchors	0.136	17/ 9.4%	34/18.8%	39/21.6%	129/71.3%
3.All anchors	0.218	29/16.0%	52/28.7%	58/32.0%	94/51.9%
4.Titles	0.256	38/21.0%	59/32.6%	65/35.9%	86/47.5%
1+2+4	0.353	47/26.0%	**86**/47.5%	97/53.6%	**43**/23.8%
1+3+4	**0.363**	**49**/27.1%	85/47.0%	**100**/55.3%	**43**/23.8%

4.2.2 Named page finding task

We run the known-item topics on all four indexes. The runs using the **Full-text** (UAmsT06nTeLM) and **Anchors** (UAmsT06nAnLM) indexes were also official submissions. We also include the three-way combination of **Full-text**, **Titles**, and **Anchors**, and a variant using the other **All anchors** index (official submission UAmsT06n3SUM).

Table 7 shows the results for the named page finding task. We make a number of observations. Here the situation is quite different from the adhoc task: the full-text index is still the best performing of all the individual indexes, but the titles index is a close second, followed again closely by the all-anchors index. The relative effectiveness of the titles-index, usually indexing but a few words per document, seems to reveal a clear bias for the topic creators to base their query on (their recollection of) the page's title. The document representations of the full-text and anchor-text indexes are based on text from disjoint sources, and—as a result—the combination of these different sources of evidence leads to a substantial improvement over the performance of the individual indexes.

4.3 Query representations

The experiments with different query representations are restricted to the adhoc task; there is only a short topic statement available for named page finding task.

We experiment with the four query representations introduced in Section 2:

T short topic statement from the title field of the topic statement;

Table 8: Results for the different query representations for the adhoc task over the 50 new topics.

	map	bpref	P@1	P@5	P@10
T	0.2878	0.3402	0.7225	0.5440	0.5260
TDN	0.3063	**0.4254**	0.7806	0.5348	0.5130
TDN10	0.2887	0.4106	0.7968	0.5600	0.5320
TDN10r	0.3042	0.4044	0.8188	0.5560	0.5360
T-TDN	0.3383	0.4012	0.8476	0.6040	0.5720
T-TDN10	**0.3601**	0.4246	**0.8729**	**0.6560**	**0.6220**
T-TDN10r	0.3405	0.3997	0.8441	0.6200	0.5860

TDN verbose topic statement combining all the fields of the topic statement;

TDN10 10 most characteristic terms in any of the fields of the topic statement;

TDN10r 10 most characteristic terms in any of the fields of the topic statement, repeated by their term frequency in the topic;

All runs use little smoothing ($\lambda = 0.9$), the run using the **T** query is identical to the official run `UAmsT06aTeLM`; the run using the **TDN10r** query is similar to the official submission `UAmsT06aTDN` which used $\lambda = 0.7$. We also include combinations of the **T** query run with each of the verbose queries, using an unweighted CombSUM combination method. The combination **T-TDN10r** is a variant of the official run `UAmsT06aTTDN` which used $\lambda = 0.7$.

The results for each of these runs is in Table 8. The results are interesting. First, runs using the verbose topic statement indeed improve over those using the short topic statement. Second, the retrieval model seems to deal well with straightforward combination of all topic fields, which also contain many term without relation to the topical content of the search request. In fact, the TDN runs outperform the runs using only selected terms from the verbose topic. Of course, the straightforward TDN query contains many terms causing a performance penalty. Third, the topic frequency of terms seems not to help performance, although more sophisticated query term weighting could be applied. Finally, in combination with a run based on the short title statement, the runs using 10 selected terms are more effective than the combination with straightforward TDN.

5 Conclusions

Our participation in the Terabyte track was inspired by a number of aims related to the size of the Terabyte track collection, we now draw some initial conclusions.

For the smoothing experiments, we found that the large-scale collections require little smoothing. This confirms earlier results on the TREC 2005 Terabyte track [4]. This may even suggest that modern, advanced retrieval models are not necessarily more effective than simpler ranking formula's (such as straightforward term-frequency).

For the different document representation, we found that these are of little value for the adhoc task, but can provide crucial additional retrieval cues for the named page finding task. The full-text and anchor-texts indexes are derived from disjoint sources, and the combination of these different sources of evidence leads to a substantial improvement of retrieval effectiveness.

For the different query representations, we found that using a more verbose query leads to an improvement of retrieval effectiveness. Modern retrieval models seem to have no problem with long verbose queries also containing many off-topic terms. Selecting the terms that are most characteristic for the topic at hand, leads to an improvement of efficiency without a loss of retrieval effectiveness.

Acknowledgments This research was supported by the Netherlands Organization for Scientific Research (NWO, grants # 612.066.302, 612.066.513, 639.072.601, and 640.-001.501), and by the E.U.'s 6th FP for RTD (project MultiMATCH contract IST-033104).

References

[1] D. Hiemstra, S. Robertson, and H. Zaragoza. Parsimonious language models for information retrieval. In *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 178–185. ACM Press, New York NY, 2004.

[2] ILPS. The ILPS extension of the Lucene search engine, 2006. http://ilps.science.uva.nl/Resources/.

[3] J. Kamps. Web-centric language models. In *Proceedings of the Fourteenth ACM Conference on Information and Knowledge Management (CIKM 2005)*. ACM Press, New York NY, USA, 2005.

[4] J. Kamps. Effective smoothing for a terabyte of text. In E. M. Voorhees and L. P. Buckland, editors, *The Fourteenth Text REtrieval Conference (TREC 2005)*. National Institute of Standards and Technology. NIST Special Publication, 2006.

[5] J. Kamps, G. Mishne, and M. de Rijke. Language models for searching in Web corpora. In E. M. Voorhees and L. P. Buckland, editors, *The Thirteenth Text REtrieval Conference (TREC 2004)*. National Institute of Standards and Technology. NIST Special Publication 500-261, 2005.

[6] J. Kamps, C. Monz, M. de Rijke, and B. Sigurbjörnsson. Approaches to robust and web retrieval. In E. M. Voorhees and L. P. Buckland, editors, *The Twelfth Text REtrieval Conference (TREC 2003)*, pages 594–599. National Institute of Standards and Technology. NIST Special Publication 500-255, 2004.

[7] Lucene. The Lucene search engine, 2006. http://lucene.apache.org/.

[8] Snowball. Stemming algorithms for use in information retrieval, 2006. http://www.snowball.tartarus.org/.

[9] C. Zhai and J. Lafferty. A study of smoothing methods for language models applied to ad hoc information retrieval. In *ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pages 49–56, Tampere, Finland, 2001. ACM Press.