CNIPA, FUB and University of Rome "Tor Vergata" at TREC 2008 Legal Track

Giambattista Amati[1], Marco Bianchi[2], Alessandro Celi[3],
Mauro Draoli[2], Giorgio Gambosi[45], and Giovanni Stilo[5]

[1] Fondazione Ugo Bordoni (FUB), Rome, Italy
gba@fub.it
[2] Italian National Centre for ICT in the Public Administrations (CNIPA), Rome, Italy
marco.bianchi@cnipa.it, draoli@cnipa.it
[3] Computer Science Department at University of L'Aquila, L'Aquila, Italy
alessandro.celi@di.univaq.it
[4] Matematics Department at University of Rome "Tor Vergata", Rome, Italy
gambosi@mat.uniroma2.it
[5] NESTOR - Laboratory of the University of Rome "Tor Vergata", Rome, Italy
stilo@nestor.uniroma2.it

1 Introduction

The TREC Legal track was introduced in TREC 2006 with the claimed purpose of to evaluate the efficacy of automated support for review and production of electronic records in the context of litigation, regulation and legislation. The TREC Legal track 2008 runs three tasks: (1) an automatic ad hoc task, (2) an automatic relevance feedback task, and (3) an interactive task. We have only taken part in the automatic ad hoc task of the TREC Legal track 2008, and focused on the following issues:

1. *Indexing*. The CDIP test collection is characterized by an large number of unique terms due to OCR mistakes. We have defined a term selection strategy to reduce the number of terms, as described in Section 2.
2. *Querying*. The analysis of the past TREC results for the Legal track showed that the best retrieval strategy basically returned a ranked list of the boolean retrieved documents. As a consequence, we have defined a strategy aimed to boost the score of documents satisfying the final negotiated boolean query. Furthermore, we defined a method for automatic construction of a weighted query from the request text, as reported in Section 3.
3. *Estimation of the K value*. We have used a query performance prediction approach to try to estimate K values. The query weighting model that we have adopted is described in Section 4.

Submitted runs and their evaluation are reported in Section 5.

2 Indexing

The IIT Complex Document Information Processing (CDIP) test collection is based on a snapshot of the Tobacco Master Settlement Agreement (MSA) sub-collection of the LTDL.

The CDIP collection presents a large number of unique terms due to OCR mistakes. For example, GOV2, one of the biggest test collections (426GB), contains about 50ML unique terms for 25ML of documents, whereas CDIP (57GB of text) contains almost twice as much: 95ML unique terms for 7ML documents.

We have used the Terrier *(TERabyte RetrIEveR)* Information Retrieval platform [1] using the following indexing configuration:

```
TrecDocTags.doctag=record
TrecDocTags.idtag=tid
TrecDocTags.process=ot
TrecDocTags.casesensitive=false
Termpipelines = Stopwords, PorterStemmer
```

With this configuration all meta-data were ignored. In addition, words containing more than 20 characters were excluded.

With such settings the generated index would have had 95ML terms with 14GB of index structures (direct, inverted and lexicon files). Therefore, we have removed from index all terms having a very low frequency (i.e. ≤ 25) producing a lexicon with “only” 9ML terms. We have noticed that the presence of OCR errors, by altering both global and local statistics, affects the retrieval quality.

3 Retrieval

We have used a variation of the DFRee model, a parameter free IR model offered by the Terrier platform. With this model we have not had to tune any parameters, and we have focused only on the proposed methodology, by evaluating the gain in performance with respect to the baseline. We have extracted the most informative terms from the top-returned documents as performed in query expansion. In this expansion process, terms in the top-returned documents are weighted using a particular DFR term weighting model. During our experiment we have found that Bo1 (Bose-Einstein statistics) term weighting models best fit with this collection. We have also made tuning on the number of top-returned documents and number of expanded term; we have obtained the best performance considering 5 documents and 10 terms.

3.1 Boolean re-rank

We have boosted the scores of all documents $d_i \in B$, where B is the set of documents satisfying the final negotiated boolean query. The score s_i of a document $d_i \in B$ was augmented by a value s:

$$s_i^{'} = s_i + s$$

where $s = s_k$ is the score of the k-th document and

$$k = \alpha * b$$

b is the number of documents in B and α a parameter. Notice that s_k varies on each topic because b depends on the final negotiated boolean query. If $\alpha = 0$ then the score s_0 of the top-rank document is added to all $d \in B$, and all these documents are shifted up in all topmost positions. Documents also keep the order computed by the DFR model. On the other hand when $\alpha \rightarrow \infty$ then $s_i^{'} \sim s_i$.

3.2 Topic processing

According to TREC Legal Track 2008 guideline each topic is composed by 4 fields: *RequestText*, *ProposalByDefendant*, *RejoinderByPlaintiff*, and*FinalQuery*. The union of all these fields would have produced a very long query. We have thus indexed all topics to obtain a query lexicon with the aim to remove all query-terms that are non informative. Then for each topic we weighted the remaining query-terms by the number of occurrences in the four fields of the topic. We show here an example of a produced query (topic n.52):

affect^1 *agricultur*^{2} *fertil*^{2} *commerci*^{2} *yield*^{4} *rai*^{1} *output*^{2} *hpf*^{3} *phosphoru*^{2} *crop*^{6} *greater*^{1}*multipl*^{1} *augment*^{1} *increa*^{1} *introduc*^{1} *boost*^{3} *fertiliz*^{3} *soil*^{1} *phosphat*^{5} *doubl*^{1} *high*^{4} *purpos*^{1} *tripl*^{1}

4 Estimating K values

4.1 Query performance prediction

Robustness is an important measure reflecting the retrieval performance of an Information Retrieval (IR) system. It particularly refers to how an IR system deals with poorly-performing queries. As stressed by Cronen-Townsend et. al. [2], poorly-performing queries considerably hurt the effectiveness of an IR system. Indeed, this issue has become important in IR research. Moreover, the use of reliable query performance predictors is a step towards determining for each query the most optimal corresponding retrieval strategy. For example, in [3], the use of query performance predictors allowed to devise a selective decision methodology avoiding the failure of query expansion. In order to predict the performance of a query, the first step is to differentiate the highly-performing queries from the poorly-performing queries. This problem has recently been the focus of an increasing research attention. In [2], Cronen-Townsend et. al. suggested that query performance is correlated with the clarity of a query. Following this idea, they used a clarity score as the predictor of query performance. In their work, the clarity score is defined as the Kullback-Leibler divergence of the query model from the collection model. In [3], Amati et. al. proposed the notion of query-difficulty to predict query performance. Their basic idea is that the term weight, as obtained in query expansion, provides evidence of the query performance.

We use a query performance prediction approach to try to estimate a correct K measure. The basic idea is quite simple: with an ”easy” query we doesn’t need to go deep in the retrieval because all relevant documents are on top; otherwise if we try with an ”hard” one, we need to goo much deeper to find all relevant documents:

$$''easy''\ query \Longrightarrow ''\ small''\ K\ value$$

$$''hard''\ query \Longrightarrow ''\ large''\ K\ value$$

4.2 Query weighting model

Since we had not time to extract expanded query weights we simply used the inverse document frequency:

$$D_i = \sum_{t \in q_i} - \log \left(\frac{n_t}{N} \right)$$

Where t is a term of a query q_i, n_t is the number of relevant documents where the term t appears, N is the number of all the documents in the collection. The correlation factor between D_i and K_i, with K_i the optimal value for the $F_1@K$, was about 30% with TREC data of 2007, whereas an higher correlation there was between D_i and the estimated number of relevant documents (37%).

4.3 Normalization's variants

To turn the coefficient D_i into an estimated K value we used the Z-Score [4]. For the new queries we have formulated three different models:

1. $$K_O = \left(1 - \frac{D - \mu_{idf}}{\sigma_{idf}}\right) \cdot \mu_k$$
   where μ_{idf} and σ_{idf} are the mean and the standard deviation of all D_i respectively and μ_k is the mean of the optimal K.
2. $$K_R = \left(1 - \frac{D - \mu_{idf}}{\sigma_{idf}}\right) \cdot \mu_R$$
   where μ_R is the mean of the cardinalities of relevant documents.
3. $$K_B = \left(1 - \frac{D - \mu_{idf}}{\sigma_{idf}}\right) \cdot \mu_B$$
   where μ_B is the mean of the cardinalities of B.

5 Submitted runs and results

For the participation to the ad hoc task we have submitted the following runs:

- The run labeled as *CTFrtSk* represents the compulsory baseline. It is computed just using the request text field as query. The K values are computed using the first model described in Section 4.3.
- The run labeled as *CTFrtSkBr0* is computed just using the (typically one-sentence) request text field as query. The boolean re-rank strategy is applied with $\alpha = 0.0$. The K values are computed using the first model described in Section 4.3.
- The run labeled as *CTFggeSkBr0* computed using the query elaboration method described in Section 3. The boolean re-rank strategy is applied with $\alpha = 0.0$. The K values are computed using the first model described in Section 4.3.
- The run labeled as *CTFgge4kBr0* is computed using the query elaboration method described in Section 3. The boolean re-rank strategy is applied with $\alpha = 0.0$. The K values are equal to 40,000 for each topic.

– The run labeled as *CTFgge10kBr0* is computed using the query elaboration method described in Section 3. The boolean re-rank strategy is applied with $\alpha = 0.0$. The K values are equal to 100,000 for each topic.
– The run labeled as *CTFggeBkBr0* is computed using the query elaboration method described in Section 3. The boolean re-rank strategy is applied with $\alpha = 0.0$. The K values are computed using the third model described in Section 4.3.
– The run labeled as *CTFggeBkBr1* computed using the query elaboration method described in Section 3. The boolean re-rank strategy is applied with $\alpha = 1.0$. The K values are computed using the third model described in Section 4.3.
– The run labeled as *CTFggeRkBr0* computed using the query elaboration method described in Section 3. The boolean re-rank strategy is applied with $\alpha = 0.0$. The K values are computed using the second model described in Section 4.3.

Results of submitted runs are shown in Table 1.

Table 1. Summary of the CTF Legal track runs.

	F_1		HF_1	
Run label	est_F_1@K	est_F_1@R	est_$HF_1@K_h$	est_$F_1@R_h$
Best run of the ad hoc task '08	0.2204	0.2458	0.1064	0.1770
Median run of the ad hoc task '08	0.1429	0.1823	0.0702	0.1109
CTFrtSk (baseline)	0.1282	0.1736	0.0481	0.0844
CTFrtSkBr0	0.1346	0.1822	0.0723	**0.1113**
CTFggeSkBr0	0.1544	0.2152	0.0811	0.1008
CTFgge4kBr0	0.1849	0.2152	0.0818	0.1008
CTFgge10kBr0	**0.2160**	0.2152	0.0788	0.1008
CTFggeBkBr0	0.1800	0.2152	0.0799	0.1008
CTFggeBkBr1	0.1856	**0.2196**	0.0824	0.1016
CTFggeRkBr0	0.1785	0.2152	**0.0947**	0.1008

With respect to the main metric (i.e. estimated F_1@K) the evaluation of our baseline (CTFrtSk) is 0.1282. Six of the eight submitted runs are over the median of the ad hoc task of Legal track 2008.

Applying the boolean re-rank to the baseline (CTFrtSkBr0) we improve our performance of +4.99%. The introduction of the the query elaboration method (CTFggeSkBr0) allows to improve our performance of +20.43% with respect to the baseline.

If we consider only those runs in which the K value is computed on basis of the query complexity, the best improvement with respect to the baseline is 44.77% (CTFggeBkBr1). This is not our best result. In fact the best run is CTFgge10kBr0. It improves the baseline performance of 68.48% considering K=10000 for each topic. CTFgge10kBr0 differs only -2.03% from the best run among all 64 runs submitted to the ad hoc task of Legal track 2008.

With respect to the estimated $HF_1@K_h$ metric the baseline is evaluated 0.0481. Seven of the eight submitted runs are over the median of the ad hoc task of Legal track 2008. Our best run is CTFggeRkBr0 with an evaluation of 0.0947 and an improvement of 96.88% over the baseline. CTFggeRkBr0 differs 12.35% from the best run among all 64 runs submitted for the ad hoc task of Legal track 2008.

We have already started a experimentation aimed to better analyze the relationship between the D_i values, introduced in Section 4.2, and the retrieval performance.

References

1. I. Ounis, G. Amati, V. Plachouras, B. He, C. Macdonald, and C. Lioma. Terrier: A High Performance and Scalable Information Retrieval Platform. In *Proceedings of ACM SIGIR'06 Workshop on Open Source Information Retrieval (OSIR 2006)*, 2006.
2. S. Cronen-Townsend, Y. Zhou, and W.B. Croft. Predicting query performance. In Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pages 299–306, Tampere, Finland, 2002.
3. G. Amati, C. Carpineto, and G. Romano. Query difficulty, robustness, and selective application of query expansion. In Advances in Information Retrieval, Proceedings of the 26th European Conference on IR Research, ECIR 2004, pages 127–137, Sunderland UK, 2004.
4. E. Altman. Financial ratios, discriminant analysis and the prediction of corporate bankruptcy. Journal of Finance, 1968.