University of Glasgow at TREC 2005:
Experiments in Terabyte and Enterprise Tracks with Terrier

Craig Macdonald, Ben He, Vassilis Plachouras and Iadh Ounis
Emails: {craigm, ben, vassilis, ounis}@dcs.gla.ac.uk

University of Glasgow, G12 8QQ, Scotland, UK

Abstract

With our participation in TREC 2005, we continue experiments using Terrier, a modular and scalable Information Retrieval (IR) framework, in 4 tasks from the Terabyte and Enterprise tracks. In the Terabyte track, we investigate new Divergence From Randomness weighting models, and a novel query expansion approach that can take into account various document fields, namely content, title and anchor text. In addition, we test a new selective query expansion mechanism which determines the appropriateness of using query expansion on a per-query basis, using statistical information from a low-cost query performance predictor. In the Enterprise track, we investigate combining document fields evidence with other information occurring in an Enterprise setting. In the email known item task, we also investigate temporal and thread priors suitable for email search. In the expert search task, for each candidate, we generate profiles of expertise evidence from the W3C collection. Moreover, we propose a model for ranking these candidate profiles in response to a query.

1 Introduction

In TREC 2005, we participate in the Terabyte track and the Enterprise track. For all our experiments, we used our Terrier platform [8]. In particular, we improve and refine a distributed version of Terrier that we deploy in the Terabyte track.

In the Terabyte track adhoc task, we investigate several new techniques for effective retrieval from large document sets such as the .GOV2 collection. Firstly, we define two new weighting models based on the Divergence From Randomness framework (DFR) [1], including a variant of a parameter-free hypergeometric model. We use a method that takes document fields into account, such as content, title and anchor text, and then show how the proposed weighting models can use this field evidence. Moreover, we develop a refined query expansion mechanism that uses the fields. Finally, we propose a novel selective query expansion mechanism which helps in deciding whether to apply query expansion for a given query. For the named page finding task, we mainly focus on how to combine evidence on the Web.

Our participation in the known item task of the Enterprise track was centred on combining various types of evidence from both the Web and the email settings of the provided W3C collection. In particular, we study to which extent email evidence such as dates and threads can help in retrieval performance. Finally, for the expert search task, our objective is to identify evidence about a candidate that is appropriate for expert search and use it to suggest the right candidates for a given query.

The remainder of the paper is organised as follows: Section 2 describes our adhoc and named page finding runs in the Terabyte track. In addition to the description of our submitted runs, we also provide some additional experimentation that investigates applying full stemming and the setting of query expansion. Section 3 describes our participation in both the known item and expert search tasks of the Enterprise track. In these newly-defined tasks, we describe the sources of evidence used and how they integrate with the retrieval mechanism; Finally, in Section 4 we provide concluding remarks.

2 Terabyte Track

This year, the Terabyte track had three tasks. We participated in only the retrieval performance tasks, namely the adhoc and named page finding tasks. This section describes our proposed approaches and the results obtained.

We indexed the .GOV2[1] collection using Terrier, which was parallelised by indexing the collection in 14 parts. After indexing, each pair of parts was merged, to give 7 parts (average size 3.6 million documents). We remove standard stopwords from the collection, and apply the first two steps of Porter's stemming algorithm, which we refer to as weak stemming [3].

Following the study of Cacheda et al [4] and our experiments in the Terabyte track last year [10], we use a distributed version of Terrier to speed up the retrieval time. This year, we use one broker, and 7 query servers, each serving one index part. Moreover, a global lexicon was created in order to speed up the retrieval process, particularly for query expansion.

2.1 Adhoc Task

In the adhoc task, we propose and test a selection of new techniques, including two new Divergence From Randomness (DFR) document weighting models, a novel query expansion mechanism using different document fields, and a selective query expansion mechanism.

Last year, the PL2 weighting model performed very well in the adhoc task. This year, we aim to improve our performance for short queries. We present two new weighting models from the DFR framework, and show how they can be applied to document fields, to give robust, effective and precise results.

Moreover, in the TREC 2004 Terabyte adhoc task, we noticed that query expansion was not particularly effective. Therefore, this year, we aim to have a refined query expansion by using more fine grained data. We propose a new query expansion mechanism, which appropriately uses the various document fields available. The queries are then re-weighted and expanded using the more refined information.

Furthermore, it is now accepted that query expansion works only on queries which have a good top-ranked document set returned by the first-pass retrieval [2, 13]. We hypothesise that if query expansion using the local collection (i.e. .GOV2) is predicted to degrade performance, then using an external resource to bring new information will improve retrieval effectiveness [7]. This leads us to propose a decision mechanism that involves a selective use of a high-quality external resource for query expansion, namely the English Wikipedia[2]. The proposed mechanism also predicts the benefit of query expansion on the local (.GOV2) and the external collection and chooses the best option, if any.

[1] Further information on .GOV2 can found from http://ir.dcs.gla.ac.uk/test_collections/.

[2] See http://en.wikipedia.org/.

2.1.1 PLL2F and DLH13F Fields-based Document Weighting Models

The aim of this research is to devise high performance weighting models for large-scale collections of documents with less extensive tuning. We describe two weighting models, PLL2F and DLH13F which take field evidence into account. Following the convention of Zaragoza et al [14], we suffix the name of field weighting models with 'F'. Our proposed PLL2F model is a variation of the PL2F model on fields. Using the PL2F models, the relevance score of a document d for a query Q is given by:

$$score(d,Q) = \sum_{t \in Q} qtw \cdot \frac{1}{tfn+1}\big(tfn \cdot \log_2 \frac{tfn}{\lambda} + (\lambda - tfn) \cdot \log_2 e + 0.5 \cdot \log_2(2\pi \cdot tfn)\big) \tag{1}$$

where λ is the mean and variance of a Poisson distribution. It is given by $\lambda = F/N$. F is the frequency of the query term in the collection and N is the number of documents in the collection. The query term weight qtw is given by qtf/qtf_{max}. qtf is the query term frequency. qtf_{max} is the maximum query term frequency among the query terms. In the rest of this paper, we use the same notations for these variables.

The final normalised term frequency tfn is the sum of the normalised term frequencies in the three fields:

$$tfn = \sum_{f} w_f \cdot tf_f \cdot \log_2(1 + c_f \cdot \frac{avg\_l_f}{l_f}), (c_f > 0) \tag{2}$$

where f refers to a field. l_f is the length of a field in the document. avg_l_f is the average length of the field in the whole collection. tf_f is the term frequency of term t in field f. c_f is the hyper-parameter of each field.

In the TREC 2005 runs, the c_f parameter of each field is set automatically using a new technique, based on the correlation between term frequency and document length, refining our previous work [6]. w_f is the weight of each field. The weights used in all our experiments in different tasks are presented in the Appendix. The above described per-field normalisation is a generalisation of the method applied in [14].

In the above PL2F model, $\frac{1}{tfn+1}$ is an addendum to normalising the relevance score. In the PLL2F model, it is replaced with:

$$\log_2 \frac{tfn}{tfn+1}$$

Hence, the PLL2F model is given by:

$$score(d,Q) \quad = \quad \sum_{t \in Q} qtw \cdot \log_2 \frac{tfn}{tfn+1}\big(tfn \cdot \log_2 \frac{tfn}{\lambda} + (\lambda - tfn) \cdot \log_2 e + 0.5 \cdot \log_2(2\pi \cdot tfn)\big) \tag{3}$$

In PLL2F, the normalised term frequency tfn is also given by Equation (2).

The DLH13F model is an extension of our previous work on the hypergeometric model[3]. Both are generalisations of the hypergeometric model in a binomial case. The hypergeometric model assumes that the document is a sample, and the population is from the collection. Note that the hypergeometric DFR weighting model does not have any parameters that require tuning. In other words, all the variables of the hypergeometric models are automatically set from the collection statistics. DLH13F uses a different generalisation of the binomial case. The relevance score of a document d for a query Q in DLH13F is given by:

[3] See http://ir.dcs.gla.ac.uk/wiki/HypergeometricModel

$$score(d,Q) = \sum_{t \in Q} qtw \cdot \frac{1}{tf + 0.5} \cdot \left(\log_2 (\frac{tf \cdot avg\_l}{l} \cdot \frac{N}{F}) + 0.5 \log_2 \left(2\pi tf (1 - \frac{tf}{l}) \right) \right) \tag{4}$$

where avg_l is the average document length in the collection. l is the document length. Note that the hypergeometric model in Equation (4) does not have a tf normalisation component, as it is assumed to be inherent to the model. tf is the weighted sum of the within-document frequencies in the each field:

$$tfn = \sum_f w_f \cdot tf_f \tag{5}$$

where tf_f is the term frequency of term t in field f, and w_f controls the contribution of the field.

2.1.2 Query Expansion on Fields

We develop a new query expansion mechanism based on fields. The query expansion mechanism refines the DFR term weighting models by a uniform combination of evidence from the three fields. In order to combine the Web evidence into the term weighting models, the variables in the models are defined upon statistics of the three fields.

In this task, we apply the Bo1 term weighting model for query expansion [1]. It is based on the Bose-Einstein statistics. Using this model, the weight of a term t in the exp_doc top-ranked documents is given by:

$$w(t) = tf_x \cdot \log_2 \frac{1 + P_n}{P_n} + \log_2(1 + P_n) \tag{6}$$

where tf_x is the frequency of the query term in the exp_doc top-ranked documents. exp_doc usually ranges from 3 to 10 [1]. P_n is given by $\frac{F}{N}$.

Terrier employs two alternate methods to determine qtw, the query term weight of a re-weighted term. The first method uses the Rocchio's $beta$ [1, 12]:

$$qtw = \frac{qtf}{qtf_{max}} + \beta \cdot \frac{w(t)}{w_{max}(t)} \tag{7}$$

where $w(t)$ is the weight of term t and $w_{max}(t)$ is the maximum $w(t)$ of the expanded query terms. β is a parameter. In all our submitted runs, β is set to 0.5.

The other one is a parameter-free method, where the qtw of a re-weighted term is given by:

$$\begin{aligned} qtw &= \frac{qtf}{qtf_{max}} + \frac{w(t)}{\lim_{F \rightarrow tf_x} w(t)} \qquad (8) \\ &= F_{max} \log_2 \frac{1 + P_{n,max}}{P_{n,max}} + \log_2(1 + P_{n,max}) \end{aligned}$$

where $\lim_{F \rightarrow tf_x} w(t)$ is the theoretical upper bound of $w(t)$. $P_{n,max}$ is given by F_{max}/N. F_{max} is the F of the term with the maximum $w(t)$ in the top-ranked documents.

If a query term does not appear in the most informative terms from the top-ranked documents, its query term weight remains equal to the original one.

2.1.3 Selective Use of External Resource for Query Expansion

We propose a new low-cost selective statistical decision mechanism for the application of query expansion. The decision mechanism is based on our previously developed pre-retrieval performance prediction technique. The expansion of the query can be either local, using documents from .GOV2, or external, using an index of the English Wikipedia. As the Terabyte adhoc topics cover a wide range of general interest topics, we hypothesise that using a different high quality collection as a collection enrichment resource for automatic query expansion could bring more useful query terms and hence could retrieve more relevant documents. The selective mechanism predicts the benefit of query expansion using either options, and adopts the most effective one. If both options are predicted to lead to the degradation of the query performance, then query expansion is disabled. Note that the approach does not involve the use of an external search engine.

We use the Average Inverse Collection Term Frequency (AvICTF) [5] to infer the successfulness of query expansion. Its definition is as follows:

$$AvICTF = \frac{\log_2 \prod_Q \frac{token_{coll}}{F}}{ql} \tag{9}$$

In the above definition, $token_{coll}$ is the number of tokens in the whole collection. ql is the query length. Using the AvICTF as a predictor, the decision mechanism is presented in Table 1. From its definition, AvICTF is comparable for different collections. Therefore, we use the same threshold setting for the two collections.

AvICTF_GOV2>threshold	AvICTF_Wiki>threshold	AvICTF_GOV2>AvICTF_Wiki	Decision
True	True or False	True	local
True or False	True	False	external
False	False	True or False	disabled

Table 1: The selective query expansion mechanism. AvICTF_GOV2 and AvICTF_Wiki are the values of AvICTF on .GOV2 and Wikipedia, respectively. The column entitled *Decision* indicates if the query expansion is *local*, *external* or *disabled*.

2.1.4 Experiments and Results

We submitted 4 runs in the adhoc task. All runs used Porter's weak stemming. The first three runs use short queries (title-only) and the last run uses long queries (title+description+narrative). The four runs are as follows:

- In run *uogTB05SQE*, we test the PLL2F model, together with the query expansion mechanism on fields.
- The run *uogTB05SQEH* adopts the parameter-free DLH13F model, and the same query expansion mechanism on fields.
- Using uogTB05SQE as the baseline, in run *uogTBSQES*, we test the selective decision mechanism using PLL2F. The applied threshold setting for the decision mechanism is 13.5.
- Finally, in run *uogTB05LQEV*, we use PLL2F and the query expansion mechanism on fields. We test the usefulness of the long queries in this run.

In all the four runs, we applied the Bo1 query expansion model. For each query, we re-weight the 20 most informative terms from the top 5 returned documents in the first-pass retrieval, and add these terms to the query. Using last year's Terabyte track best setting, the Rocchio's beta is set to 0.5.

Table 2 presents the performance achieved by our submitted runs, along with that of the participants. According to the results:

- The performance of our submitted runs is considerably above the median of all of the submitted runs. This shows that the two newly proposed models achieved effective retrieval performance in the Terabyte track adhoc task.
- Among our four submitted runs, the run using long queries (i.e. uogTB05LQEV) does not have the best MAP but the best bpref and Pre@10. This seems to indicate that the descriptions and narratives in the topics of this task are not very useful.
- Although the run uogTB05SQES which uses the selective query expansion mechanism has a lower MAP than the baseline run uogTB05SQE, it achieves a better Pre@10 (i.e. 0.6580 against 0.6300). Our explanation is that the selective query expansion mechanism refines the top-ranked documents, while it introduces noise to the rest of the returned documents. Therefore, the selective query expansion mechanism provides a better early precision.

Run id	uogTB05SQE	uogTB05SQEH	uogTB05SQES	uogTB05LQEV	best	median	worst
MAP	**0.3755**	0.3548	0.3687	0.3650	0.5056	0.2815	0.0109
bpref	0.3751	0.3629	0.3698	**0.3770**	0.5236	0.3030	0.0255
Pre@10	0.6300	0.5860	0.6580	**0.6780**	N/A	N/A	N/A

Table 2: The mean average precision (MAP), binary preference (bpref) and precision at 10 (Pre@10) of our submitted runs, and that achieved by all participants. Pre@10 achieved by all participants is not available. The measures in **bold** are the best in our submitted runs.

Overall, the two newly proposed models, as well as the query expansion mechanism on fields are shown to be effective. Moreover, the selective query expansion mechanism increases the early precision performance of the system. Finally, in terms of MAP, the long queries are shown not to be useful in this task.

2.1.5 Additional Query Expansion Runs

In this section, we conduct some additional runs to further evaluate the query expansion mechanism on fields. We vary the number of re-weighted terms (exp_term), the number of top-ranked documents (exp_doc) from which the re-weighted terms are selected, and the Rocchio's beta value used for query expansion. We also apply the parameter-free query expansion (see Equation 8), which is an alternative to the Rocchio's beta. Tables 3 and 4 contain the MAP measures obtained using the PLL2F and DLH13F models, respectively.

According to the results in Tables 3 and 4, the query expansion mechanism on fields is shown to be robust with various query expansion settings. With some settings, we outperform our best submitted runs. In particular, one setting achieves an MAP of 0.3816.

From Table 3, note that our parameter-free query expansion mechanism achieves an MAP that is better than our best submitted run (0.3782 vs. 0.3755).

2.1.6 Experiments with Stemming

In our previously presented experiments, Porter's weak stemming is applied, as this can increase the precision of the results without over-impacting recall. To compare with the submitted runs of other participating groups, we experiment with applying full stemming.

exp_doc	exp_term	$\beta = 0.1$	$\beta = 0.2$	$\beta = 0.5$	$\beta = 0.8$	parameter-free
3	10	0.3590	0.3666	0.3635	0.3529	0.3696
5	20	0.3713	**0.3816**	0.3755	0.3633	**0.3782**
10	40	0.3737	0.3810	0.3655	0.3491	0.3669
15	60	0.3758	0.3740	0.3562	0.3401	0.3610

Table 3: Obtained MAP using PLL2F with Bo1 query expansion model for different settings.

exp_doc	exp_term	$\beta = 0.1$	$\beta = 0.2$	$\beta = 0.5$	$\beta = 0.8$	parameter-free
3	10	0.3231	0.3322	0.3403	0.3357	0.3360
5	20	0.3279	0.3426	0.3567	0.3552	0.3521
10	40	0.3394	0.3576	0.3640	0.3589	0.3615
15	60	0.3424	0.3591	***0.3655***	0.3608	0.3588

Table 4: Obtained MAP using DLH13F with Bo1 query expansion model for different settings.

Table 5 shows the results in terms of MAP, bpref and Pre@10 of the three runs using short queries: uogTB05SQE is the submitted run, using PLL2F and weak stemming; uogTB05SFullQE uses the same setting as uogTB05SQE but applies full Porter stemming. From the table, we can see that applying full stemming improves the bpref and Pre@10 achieved compared to weak stemming. However, as the setting of uogTB05SFullQE was determined from training on weak stemming, we also show the results for uogTB05SFullQEbis, where the setting has been trained on full stemming. This increases the performance of full stemming over all three evaluation measures. Note, that the run uogTB05SFullQEbis outperforms the best submitted short queries run of all participants, indri05AdmfS by the University of Massachusetts, by a 4% margin on all three evaluation measures (indri05AdmfS achieved MAP 0.3886, bpref 0.3920, and Pre@10 0.6340).

Run id	uogTB05SQE	uogTB05SFullQE	uogTB05SFullQEbis
MAP	0.3755	0.3622	**0.4021**
bpref	0.3751	0.3785	**0.4089**
Pre@10	0.6300	0.6400	**0.6600**

Table 5: The mean average precision (MAP), binary preference (bpref) and Precision at 10 (Pre@10) of different stemming runs.

2.2 Named Page Finding Task

In the named page finding task, we focused our participation on applying and refining techniques which had successfully worked during our previous Web track participations on the .GOV collection.

As in the adhoc task, we use a combination of evidence from three fields: content, title and anchor text. However, obtaining the correct parameter settings is important in order to achieve the best retrieval performance.

2.2.1 Training

For our training, we started from the fact that the .GOV collection is a significant subset of the .GOV2 collection. We used two forms of training for our setting this year: Firstly we used 300 topics from the named page finding tasks of the 2002 and 2003 Web tracks to find a good setting for the system on the .GOV collection. We then directly transferred this setting to the .GOV2 collection. Alternatively, using the named page finding topics from the 2002-2004 Web tracks, we were able to generate 190 named page finding topics for the .GOV2 collection, by mapping the .GOV document numbers into .GOV2.

2.2.2 Experiments and Results

We submitted 4 runs to the named page finding task. These were created using the previously described indexing method, and the distributed version of Terrier described in Section 2. The PL2F weighting model on fields, see Equations (1) & (2), was used to rank documents.

- *uogNP05Base* is our baseline run. It uses the Porter's weak stemming index, and the best setting as found using the 300 topics on .GOV.
- *uogNP05BaseN* tests the difference in applying no stemming for this task. The parameter setting is taken from the best setting using the 300 topics on .GOV.
- *uogNP05bis* tests training a retrieval system using the topics migrated from .GOV. It also uses a Porter's weak stemming index.
- Finally, *uogNP05bisP* is based on uogNP05bis, but applies proximity search.

Table 6 shows the performance achieved by our submitted runs in terms of MRR, along with that of the participants. According to the results, all our submitted runs were above the median. The runs uogNP05Base and uogNP05BaseN perform better than the other two. We surmise that the setting migrated from .GOV performs better than training on the real collection using a smaller number of topics. Finally, applying either no stemming or proximity search produces marginal increases in performance.

Run id	uogNP05Base	uogNP05BaseN	uogNP05bis	uogNP05bisP*	best	median	worst
MRR	0.400	**0.401**	0.381	0.382	0.6660	0.3786	0.0380

Table 6: The mean reciprocal rank (MRR) achieved by the submitted TREC Terabyte named page finding runs, and that by all participants. *Note that the submitted run uogNP05bisP was affected by a technical error. Here we show the corrected MRR for this run - the previous MRR was 0.381.

2.2.3 Additional Runs

In this section, we test applying other common sources of evidence in Web IR that can improve the obtained retrieval performance. In particular, we apply PageRank [9], and the Static Absorbing Model [11] as link analysis, the In-Degree of a document, and the length of the URL path in characters as additional sources of evidence. The baseline for all additional runs was uogNP05Base (MRR 0.400).

The results for the additional runs are shown in Table 7. All forms of additional evidence produced slight increases in performance. Surprisingly, applying the In-Degree evidence produces the highest observed increase in performance.

	PageRank	Absorbing Model	In-Degree	URL Scoring	PageRank + URL Scoring
MRR	0.408	0.408	**0.417**	0.406	0.411

Table 7: The mean reciprocal rank (MRR) achieved by the additional TREC Terabyte named page runs, showing the improvements in applying additional evidence.

3 Enterprise Track

In the Enterprise track, we chose to submit runs for two tasks: email known item, and expert search. Our experiments focused around the application of Web IR techniques, and the development of new techniques specific to Enterprise search tasks. In particular, we investigate the usefulness of document metadata in an email setting, as well as various sources of evidence in expert search.

3.1 Email Known Item Task

Our aim in the known item task was to identify sources of evidence suitable for use in an Enterprise setting. As the W3C collection is distributed as a Web crawl, we chose firstly to use Web evidence, indexing separately content, title and anchor text fields of a document. The anchor text field of a document consisted of the anchor text from all incoming hyperlinks to the document, from any part of the collection. When indexing, we removed standard stopwords from the collection, and apply Porter's weak stemming.

In applying Enterprise evidence, we used two forms of evidence exhibited by emails to form three priors, which will be used to refine the ranking. Firstly, one from the thread structure of the emails, and two from temporal evidence that can be obtained from emails. The priors act upon the $score(d, Q)$ of an email document d with respect to a query Q, as given by the PL2F weighting model on fields - see Equations (1) & (2). The hyper-parameters and weights were trained using the provided training topics. In the following, we describe our three priors.

Threads: We assume that the relevant emails in a known item task will be at the start of a thread, rather than a reply. Hence, we boost the score of emails that occurred higher in the thread tree. The score of a retrieved document d is altered as follows:

$$score(d, Q) = score(d, Q) + \frac{\mathit{weight}}{\mathit{offset} + \mathit{Generation}(d)} \tag{10}$$

where *weight* and *offset* are two free parameters, and *Generation* is a function that returns the thread depth of document d. The values 2.557 and 2.2 were used for *weight* and *offset* respectively, determined using the provided training topics.

Email Dates: The topics were generated in 2004/2005 with a particular focus on more recent years of the collection. Hence, we chose to alter the scores of the retrieved documents by altering the score of a retrieved document d, as follows:

$$score(d, Q) = score(d, Q) + g \cdot \mathit{Date}(d) \tag{11}$$

where $Date(d)$ is the date when the email was sent. g is a free parameter, which was set to 1.154e-3 using the provided training topics.

Topics Dates: We experiment with a mechanism that boosts the retrieved documents that were sent near the date mentioned in a topic, by applying a Gaussian function to the scores of the retrieved documents. The score of retrieved document d that was sent around the $topicDate$ is increased as follows:

$$score(d,Q) = score(d,Q) + \frac{h}{\sqrt{2\pi}} \exp(-\frac{(Date(d) - \mathit{topicDate})^2}{2\sigma^2}) \tag{12}$$

where h is a free parameter; $Date(d)$ is the date when the email was sent; σ is the parameter of the Gaussian function. We use it to control the width of the temporal boosting. We apply a narrow boost ($\sigma = 30$) when the target date in the topic is a particular day, a wider boost ($\sigma = 70$) when the target date is a month, and wider still ($\sigma = 450$) when the topic mentions only a year. The value used for h was 5.99, determined using the provided training topics.

3.1.1 Experiments and Results

We submitted 4 runs to the known item email task of the Enterprise track: *uogEBase* is the baseline run; *uogEDates1* applies the Topics Dates evidence to the baseline; *uogEDates2* applies the Email Dates evidence to the baseline; finally *uogEDates12T* combines all three priors, including the Threads evidence.

Run id	uogEBase	uogEDates1	uogEDates2	uogEDates12T	best	median	worst
MRR	0.619	0.618	0.619	**0.621**	0.7524	0.4545	0.027

Table 8: The mean reciprocal rank (MRR) achieved by our submitted TREC Enterprise known item runs, and that achieved by all participants.

Table 8 shows the results of the runs we submitted to the known item task, as well as the results by all participants. We can see that all 4 runs performed considerably above the median. Compared to the baseline uogEBase, run uogEDates12T, which combines all three forms of evidence, performs the best. Email Dates (uogEDates2) appears to have made a very slight improvement in retrieval performance. Topic Dates (uogEDates1) has no impact on retrieval performance.

Overall, we found that the retrieval approaches worked extremely well for this email known item task. Our best MRR (0.621) was much higher than the median of 0.4545. This was the best submitted run from all participants for this task.

In conclusion, our participation to the known item task was extremely successful. In addition to the usefulness of the Web evidence, we found that the Thread structure was the most effective evidence, while using dates appear to be less effective. Moreover, all four of our submitted runs outperformed the runs of all other participants in this task.

3.1.2 Additional Runs

In our additional runs, we test two hypotheses. Firstly, whether using anchor from the entire collection is no better than using anchor text from only the lists subset of the collection. Secondly, we wanted to test the effect of stemming in this task, in particular whether weak stemming was the most effective form of stemming to apply.

To test our anchor text hypothesis, we compared runs using two different anchor text indices: the anchor text used by the submitted runs, which uses the anchor text of hyperlinks from the entire W3C collection; and anchor text of hyperlinks from only the lists subset of the collection. By excluding the ‘external anchor text’, the number of tokens in the anchor text field dropped significantly. Table 9 shows the achieved MRR results for varying the external anchor text applied - using external anchor text corresponds to the submitted run uogEBase. From this table, we can see that using the external anchor text had a small improvement (from 0.615 to 0.619 MRR). On closer inspection of these results, we determined that the improvement was not due to any different occurrences of query terms in the relevant documents, but merely two pairs of document swaps between the two rankings of the affected two topics.

To test our stemming hypothesis, we varied the stemming applied to the run uogEBase. Table 10 shows the results for the applied stemming. Weak stemming appears to be the best form for this task, closely followed by full stemming. No stemming performs considerably lower at MRR 0.600. Hence, it would seem that applying weak stemming is in fact a good choice for this high precision task.

Anchor Text	From lists only	From entire collection	best	median	worst
MRR	0.615	**0.619**	0.7524	0.4545	0.027

Table 9: The mean reciprocal rank (MRR) achieved by runs using differing amounts of anchor text.

Stemming	None	Weak	Full	best	median	worst
MRR	0.600	**0.619**	0.616	0.7524	0.4545	0.027

Table 10: The mean reciprocal rank (MRR) achieved by runs applying differing levels of stemming.

3.2 Expert Search Task

Our aims in this task are two-fold: to identify important evidence for expert search; and to determine how to combine and weight the evidence.

We determine documents from the W3C collection to include in the generation of an implicit profile of the expertise of each candidate. As shown below, our proposed profile is the merging of up to three sources of evidence from the collection, namely the expert's homepage, occurrences of his/her name in the collection, and email threads he/she was involved in.

Each candidate will have a unique corresponding profile. However, we assume that each of the three sources of evidence has a different importance. Therefore, when merging the sources of evidence into a single profile, each document in the profile has a weight.

Once the profile is built, for a given topic, we rank profiles as in a standard IR system. However, we have two document length normalisation problems. The obvious one is related to the fact that experts that have more presence in the collection will end up with having a much bigger profile than those who are less active. Hence, we normalise with respect to the profile length. The second normalisation accommodates the variance of document length in the W3C corpus.

We use the In_expC2 DFR weighting model [1] to rank profiles. More specifically, the relevance score of a profile *pro* to a query Q in In_expC2 is given by:

$$score(pro, Q) = \sum_{t \in Q} qtw \cdot \frac{F_{pro} + 1}{Nt_{pro} \cdot (tfn_{pro} + 1)} \big(tfn_{pro} \cdot \log_2 \frac{N_{pro} + 1}{n_e + 0.5} \big) \tag{13}$$

where qtw is the query term weight as defined in Section 2.1.1, N_{pro} is the total number of generated implicit profiles, F_{pro} is the term frequency of t in all profiles and Nt_{pro} is the number of candidates having a profile that contains t. tfn_{pro} is the normalised term frequency in the profile. n_e is given by $N_{pro} \cdot \big(1 - (1 - \frac{Nt_{pro}}{N_{pro}})^{F_{pro}}\big)$.

To accommodate both required normalisations above, for the length variation of the profiles, we use the following normalisation function:

$$tfn_{pro} = tf_{pro} \cdot \log_e(1 + c_{pro} \cdot \frac{avg\_l_{pro}}{l_{pro}}), (c_{pro} > 0) \tag{14}$$

where c_{pro} is the hyper-parameter for the profiles, l_{pro} is the length of the profile, and avg_l_{pro} is the average length of all profiles. tf_{pro} is the term frequency in the merged documents of that profile.

For the document length normalisation, we apply a variation of the normalisation in Equation (2), but rather than using fields, we use the documents of the profile. In this case, the weights are related to the importance of each source of evidence. In the next sections, we provide contents of the profiles and the submitted runs.

3.2.1 Identifying Documents for each Candidate

From the list of candidates provided by the track organisers, we determine documents that could be included in each candidate's profile, in three different ways:

Occurrences of Person in Corpus: We generated queries which were used to identify documents that mentioned each candidate, based on the occurrences of variations of the candidate's name and email address in the collection. The documents returned form the $Occurrence$ set of each candidate.

Personal Websites: From the candidate list, we identified the username of candidates with an email address ending '@w3.org', and used this to identify the personal website of the candidate in the collection, should it exist. From the URL list of the W3C collection, all documents under the personal website of each candidate were added to the $Homepage$ set of each candidate.

Email Threads: We used the threading evidence of the emails in the collection to identify additional documents for each candidate's profile. For each email in the $Occurrence$ of a candidate, the other emails in the same thread were added to the $EmailThread$ set of each candidate.

3.2.2 Experiments and Results

We have submitted five runs for the expert search task. We indexed the W3C corpus by removing standard stopwords and applying Porter's weak stemming.

Run id	uogES05B2	uogES05Cbis	uogES05CbiH	uogES05CbaDT	uogES05Cbase	best	median
MAP	0.1834	0.1836	**0.1851**	0.1730	0.1740	0.3707	0.1402
BPref	0.4456	**0.4662**	0.4543	0.4477	0.4293	0.6590	0.4375
Pre@10	0.3140	**0.3240**	0.3160	0.3180	0.3200	N/A	N/A

Table 11: The mean average precision (MAP), binary preference (BPref) and precision at 10 (Pre@10) of our submitted runs, as well as that achieved by all participants. Pre@10 achieved by all participants is not available.

Run *uogES05B2* is our baseline run - it only uses one source of evidence, namely the $Occurrence$ sets, for generating the candidate profiles. All parameters were set using the provided training topics. *uogES05Cbis* and *uogES05CbiH* differ from the baseline as they use the fields of each document, as described in Section 2.1.1.

Additionally, *uogES05CbiH* adds the $Homepage$ sets into the generated profiles. Compared to uogES05CbiH, *uogES05CbaDT* uses the $EmailThread$ sets rather than the $Homepage$ sets. Finally, *uogES05Cbase* changes the parameter setting of uogES05Cbis by further training.

Table 11 shows the results of the submitted runs. Both our best runs in terms of MAP (i.e. uogES05CbiH & uogES05Cbis respectively) used the $Occurrence$ sets and fields. Comparing uogES05B2 & uogES05Cbis, shows that applying fields improves Pre@10, but can degrade performance if not properly tuned (uogES05Cbase). In addition, adding the $Homepage$ sets in the run uogES05CbiH increased MAP, but adding the $EmailThread$ documents into each profile did not (uogES05CbaDT).

Overall, all our runs performed above median MAP. However, we also noticed that our performance on the provided training queries was not consistent with the submitted runs. This suggests that our used setting in this task is perhaps not optimal.

4 Conclusions

Overall, the performance we achieved in the adhoc task of the Terabyte track, and the known item task of the Enterprise track were both extremely effective. We achieved the best four submitted runs by MRR in the known item task of the Enterprise track, and the second highest run by MAP in the Terabyte track adhoc task. Moreover, in the Terabyte track, we showed that if full stemming was applied, we were able to exceed the performance of the best submitted short queries run. In the named page finding task of the Terabyte track our results were excellent, and were further improved when link or URL structure was taken into account. In the the expert search task of the Enterprise we developed a promising model that can be improved upon with additional evidence in the future. We surmise that overall our participation in TREC 2005 was very successful.

Appendix

Task	$c_{Content}$	c_{Title}	c_{Anchor}	w_{Title}	w_{Anchor}
TB Adhoc	6	5000	10	1	0.5
TB Named page					
Base(Weak Stemming)	1.039	15.475	28.82	2.656	0.515
Base(No Stemming)	1.220	17.845	84.545	5.424	0.597
Bis	0.529	15.199	10.906	5.932	1.500
Ent Known item	9.600	67.000	3.578	8.449	0.850

Table 12: The used weights of each field in different tasks. The weight of the content in a document is always set to 1. Note that in the Terabyte adhoc task, the hyper-parameter values were determined automatically, as mentioned in Section 2.1.1.

Acknowledgements

Part of this work is supported by The Leverhulme Trust, grant number F/00179/S, as part of the Smooth project on length normalisation (URL: http://ir.dcs.gla.ac.uk/smooth).

References

[1] G. Amati. *Probabilistic Models for Information Retrieval based on Divergence from Randomness*. PhD thesis, Department of Computing Science, University of Glasgow, 2003.

[2] G. Amati, C. Carpineto and G. Romano. Query Difficulty, Robustness, and Selective Application of Query Expansion. In *Proceedings of ECIR 2004.* Sunderland, UK.

[3] R. K. Belew Finding Out About: A Cognitive Perspective on Search Engine Technology and the WWW. Cambridge University Press, 2000.

[4] F. Cacheda, V. Plachouras and I. Ounis. A Case Study of Distributed Information Retrieval Architectures to Index One Terabyte of Text. In *Information Processing and Management Journal*, 41(5), 2005.

[5] B. He and I. Ounis. Query Performance Prediction. In *Information Systems, Special Issue for the String Processing and Information Retrieval: 11th International Conference (SPIRE2004).* 2005.

[6] B. He and I. Ounis. A study of the Dirichlet Priors for term frequency normalisation. In *Proceedings of ACM SIGIR 2005*, Salvador, Brazil.

[7] K.L. Kwok and M. Chan. Improving two-stage ad-hoc retrieval for short queries. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval.* Pages: 250 – 256, Melbourne, Australia, 1998.

[8] I. Ounis, G. Amati, V. Plachouras, B. He, C. Macdonald, and D. Johnson. Terrier Information Retrieval Platform. In *Proceedings of ECIR 2005*, Santiago de Compostela, Spain.

[9] L. Page, S. Brin, R. Motwani, and T. Winograd. *The PageRank citation ranking: Bringing order to the Web.* Technical report, Stanford Digital Library Technologies Project, 1998.

[10] V. Plachouras, B. He, and I. Ounis. University of Glasgow at TREC2004: Experiments in Web, Robust and Terabyte tracks with Terrier. In *Proceedings of TREC 2004.* Gaithersburg, MD.

[11] V. Plachouras, I. Ounis, and G. Amati. The Static Absorbing Model for the Web. *Journal of Web Engineering*, 2005.

[12] J. Rocchio. Relevance feedback in information retrieval. In *The Smart Retrieval system—Experiments in Automatic Document Processing.* Salton, G., Ed. Prentice-Hall Englewood Cliffs. NJ.

[13] E. Yom-Tov, S. Fine, D. Carmel and A. Darlow. Learning to estimate query difficulty: including applications to missing content detection and distributed information retrieval. In *Proceedings of ACM SIGIR 2005.* Salvador, Brazil.

[14] H. Zaragoza, N. Craswell, M. Taylor, S. Saria and S. Robertson. Microsoft Cambridge at TREC 13: Web and Hard Tracks. In *Proceedings of TREC 2004.* Gaithersburg, MD.