ARTICLE IN PRESS

Available online at www.sciencedirect.com

SCIENCE DIRECT®

ELSEVIER

Information Systems ▮ (▮▮▮▮) ▮▮▮–▮▮▮

www.elsevier.com/locate/infosys

Query performance prediction

Ben He*, Iadh Ounis

Department of Computing Science, University of Glasgow, Glasgow G12 8QQ, UK

Abstract

The prediction of query performance is an interesting and important issue in Information Retrieval (IR). Current predictors involve the use of relevance scores, which are time-consuming to compute. Therefore, current predictors are not very suitable for practical applications. In this paper, we study six predictors of query performance, which can be generated prior to the retrieval process without the use of relevance scores. As a consequence, the cost of computing these predictors is marginal. The linear and non-parametric correlations of the proposed predictors with query performance are thoroughly assessed on the Text REtrieval Conference (TREC) disk4 and disk5 (minus CR) collection with the 249 TREC topics that were used in the recent TREC2004 Robust Track. According to the results, some of the proposed predictors have significant correlation with query performance, showing that these predictors can be useful to infer query performance in practical applications.

Keywords: Query performance prediction; Information retrieval; Experiments

1. Introduction

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. [1], poorly performing queries considerably hurt the effectiveness of an IR system. Indeed, this issue has become important in IR research. For example, since 2003, TREC has conducted a new track, namely the Robust Track, which aims to investigate the retrieval performance of poorly performing queries. 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 [2], 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 [1], 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 [2], Amati et al. proposed the

*Corresponding author.
E-mail addresses: ben@dcs.gla.ac.uk (B. He), ounis@dcs.gla.ac.uk (I. Ounis).

0306-4379/$ - see front matter © 2005 Elsevier B.V. All rights reserved.
doi:10.1016/j.is.2005.11.003

notion of *query-difficulty* to predict query performance. Their basic idea is that the term weight, which is given by the divergence of the query terms' distribution in the top-retrieved documents from their distribution in the whole collection, provides evidence of the query performance.

Both methods mentioned above select a feature of a query as the predictor, and estimate the correlation of the predictor with the query performance. However, the use of these methods suffers from the time-consuming computation of the relevance scores. For example, for a very large-scale collection, such as the .GOV2 collection, which was used in the TREC2004 Terabyte Track, it is unpractical to compute the relevance scores just for the query performance prediction.

In this paper, we study a set of predictors that can be computed before the retrieval process takes place. The retrieval process refers to the process where the IR system scans the inverted files for the query terms and assigns a relevance score to each retrieved document. The experimental results show that some of the proposed predictors have significant correlation with query performance. Therefore, these predictors can be applied in practical applications.

The remainder of this paper is organised as follows. Section 2 proposes six pre-retrieval predictors of query performance. Sections 3 and 4 study the linear and non-parametric correlations of the predictors with average precision. Section 5 presents a smoothing method for improving the most effective proposed predictor and the obtained results. Finally, Section 6 concludes this work and suggests further research directions.

2. Predictors of query performance

In this section, we propose a list of pre-retrieval predictors of query performance. The proposed list of predictors is inspired by previous works related to probabilistic IR models, including the language modelling approach [3] and Amati and van Rijsbergen's Divergence From Randomness (DFR) models [4]. The proposed predictors are intuitive, simple to implement, and are purely based on intrinsic statistical features of the queries. Below, we list the six proposed predictors:

- *Query length.* According to Zhai and Lafferty's work [5], in the language modelling approach, the query length has a strong effect on the smoothing methods. In our previous work, we also found that the query length heavily affects the length normalisation methods of the probabilistic models [6].

For example, the optimal setting of the only parameter c in Amati and van Rijsbergen's probabilistic framework is query-dependent [4]. The empirically obtained setting of its parameter c is $c = 7$ for short queries and $c = 1$ for long queries, suggesting that the optimal setting depends on the query length. Therefore, the query length could be an important characteristic of the queries. In this paper, we define the query length as:

Definition 1 (ql). The query length is the number of non-stop words in the query.

- *The distribution of informative amount in the query terms.* In general, each term can be associated with an inverse document frequency ($idf(t)$) describing the informative amount that a term t carries. As stressed by Pirkola and Jarvelin, the difference between the *resolution power* of the query terms, which is given as the $idf(t)$ values, could affect the effectiveness of the retrieval performance [7]. Therefore, the distribution of the $idf(t)$ factors in the composing query terms might be an intrinsic feature that affects the retrieval performance. The idea of using *idf* for prediction purposes was also applied in [8,9]. In this paper, we propose the following two novel definitions for the distribution of informative amount in the query terms. Both definitions have a very low computational cost:

Definition 2 ($\gamma 1$). Given a query Q, the distribution of informative amount in its composing terms, called $\gamma 1$, is represented as

$$\gamma 1 = \sigma_{idf}, \tag{1}$$

where σ_{idf} is the standard deviation of the *idf* of the terms in Q.

For *idf*, we use the INQUERY's *idf* formula [10]:

$$idf(t) = \frac{\log_2(N + 0.5)/N_t}{\log_2(N + 1)}, \tag{2}$$

where N_t is the number of documents in which the query term t appears and N is the number of documents in the whole collection.

Another possible definition representing the distribution of informative amount in the query terms is:

Definition 3 ($\gamma 2$). Given a query Q, the distribution of informative amount in its composing terms, called $\gamma 2$, is represented as

$$\gamma 2 = \frac{idf_{max}}{idf_{min}}, \tag{3}$$

where idf_{max} and idf_{min} are the maximum and minimum *idf* among the terms in query Q, respectively.

The *idf* of Definition 3 is also given by the INQUERY's *idf* formula.

- *Query scope.* The notion of query scope characterises the generality/specificity of a query. For example, a query like "Olympic games" is more general than a query like "Olympics Sydney" as the latter looks for documents about the Olympic games of a particular location. The query scope was originally studied in [11], where the query scope was defined as a decreasing function of the size of documents containing at least one query term. According to this study, the size of this document set is an important property of the query. Following [11], in this work, we define the query scope as follows:

Definition 4 (ω). The query scope is

$$\omega = -\log(n_Q/N), \tag{4}$$

where n_Q is the number of documents containing at least one of the query terms, and N is the number of documents in the whole collection.

In the above definition, query scope ω is a decreasing function of n_Q. A large n_Q value will result in a low query scope value.

- *Query clarity.* Query clarity is inversely proportional to the ambiguity of a query. For example, a query term like "Jordan" is ambiguous because it could have different meanings. "Jordan" could refer to the basketball player Michael Jordan or the country called Jordan. According to the work by Cronen-Townsend et al. [1], the clarity (or on the contrary, the ambiguity) of a query is an intrinsic feature of a query, which has an important impact on the system performance. Cronen-Townsend et al. proposed the clarity score of a query to measure the coherence of the language usage in documents, whose models are likely to generate the query [1]. In their definition, the clarity of a query is the sum of the Kullback–Leibler divergence of the query model from the collection model. However, this definition involves the computation of relevance scores for the query model, which is time-consuming. In this paper, we simplify the clarity score by proposing the following definition:

Definition 5 (*SCS*). The simplified query clarity score is given by

$$SCS = \sum_{Q} P_{ml}(w|Q) \cdot \log_2 \frac{P_{ml}(w|Q)}{P_{coll}(w)}. \tag{5}$$

In the above definition, $P_{ml}(w|Q)$ is given by qtf/ql. It is the maximum likelihood of the query model of the term w in query Q. *qtf* is the number of occurrences of a query term in the query and *ql* is the query length. $P_{coll}(w)$ is the collection model, which is given by $tf_{coll}/token_{coll}$, where tf_{coll} is the number of occurrences of a query term in the whole collection and $token_{coll}$ is the number of tokens in the whole collection.

The definition is simplified in terms of complexity and overhead. In Sections 3 and 4, we will show that this simplified definition has significant linear and non-parametric correlations with query performance. Moreover, in Section 5, the proposed simplified clarity score is improved by smoothing the query model.

- *Average inverse collection term frequency.* The definition of the simplified clarity score, given in Eq. (5), is actually very similar to Kwok's idea of the *inverse collection term frequency* (ICTF) [12]. According to his work, ICTF can be seen as a replacement of *idf* and is correlated with the quality of a query term.

In this paper, we use the average of inverse collection term frequency of the query terms (AvICTF) to infer query performance:

Definition 6 (*AvICTF*). The average inverse collection term frequency is given by

$$AvICTF = \frac{\log_2 \prod_Q (token_{coll}/tf_{coll})}{ql}. \tag{6}$$

In the above definition, tf_{coll} is the number of occurrences of a query term in the whole collection and $token_{coll}$ is the number of tokens in the whole collection. ql is the query length.

The denominator ql is the reciprocal of the maximum likelihood of the query model of SCS in Eq. (5). The use of the average of ICTF is similar to measuring the divergence of a collection model (i.e. ICTF) from a query model. Therefore, AvICTF and SCS should have similar performance. Later in our experiments, we show that AvICTF is comparable with SCS in inferring query performance.

The six above proposed predictors are pre-retrieval predictors. Compared with the classical predictors introduced in Section 1, the computation of these predictor does not involve the use of relevance scores. Therefore, the use of these pre-retrieval predictors has a very low computational cost.

In the following sections, we will study the correlations of the predictors with query performance. In order to fully investigate the predictors, we check both linear and non-parametric dependence of the predictors with query performance. The latter is a commonly used measure for the query performance predictors, since the distribution of the involved variables are usually unknown. On the contrary, the linear dependence assumes a linear distribution of the involved variables. Although this strong assumption is not always true, the linear fitting of the variables can be straightforwardly applied in practical applications.

3. The linear dependence between the predictors and average precision

In this section, we measure the linear correlation r of each predictor with the actual query performance, and the p-value associated to this correlation [13]. We use average precision as the focus measure representing the query performance in all our experiments. Again, note that the linear correlation assumes a linear distribution of the involved variables, which is not always true.

The correlation r varies within $[-1, 1]$. It indicates the linear dependence between the two pairs of variables. A value of $r = 0$ indicate that the two variables are independent. $r > 0$ and $r < 0$ indicate that the correlation between the two variables is positive and negative, respectively. The p-value is the probability of randomly getting a correlation as large as the observed value, when the true correlation is zero. If p-value is small, usually less than 0.05, then the correlation is significant. A significant correlation of a predictor with average precision indicates that this predictor could be useful to infer the query performance.

3.1. Test data and settings

The document collection used to test the efficiency of the proposed predictors is the TREC disk4&5 test collection (minus the Congressional Record on disk4). The test queries are the TREC topics that are numbered from 301 to 450 and from 601 to 700, which were used in the TREC2004 Robust Track. Topic 672 was eliminated from the evaluation by the track organisers as the assessors had found no relevant document for this topic. Thus, there are 249 queries in total. For all the documents and queries, the stop-words are removed using a standard list and Porter's stemming algorithm is applied.

Each query consists of three fields, i.e. Title, Description and Narrative. In our experiments, we define three types of queries with respect to the different combinations of these three fields:

- *Short query*: Only the titles are used.
- *Normal query*: Only the descriptions are used.
- *Long query*: All the three fields are used.

The statistics of the length of the three types of queries are provided in Table 1. We run experiments for the three types of queries to check the impact of the query type on the effectiveness of the predictors, including the query length.

In the experiments of this section, given the average precision value of each query, we compute r and the corresponding p-value of the linear dependence between the two variables, i.e. average precision and each of the predictors. The average

Table 1
The statistics of the length of the three types of queries

	Short query	Normal query	Long query
avg_ql	2.62	7.94	21.75
Var(*ql*)	0.66	13.09	85.96

avg_ql is the average query length. *Var*(*ql*) is the variance of the length of the queries.

precision values of the test queries are given by the PL2 and BM25 document weighting models, respectively. We use two statistically different models in order to check if the effectiveness of the predictors is independent of the used document weighting models.

Our experiments are conducted using the Terrier system (URL: http://ir.dcs.gla.ac.uk/terrier/). Terrier is a modular platform for the rapid development of large-scale IR applications, providing indexing and retrieval functionalities. Terrier provides a range of weighting models, from classical models such as *tf-idf* and BM25, to models based on the DFR framework [4].

PL2 is one of the DFR document weighting models. Using the PL2 model, the relevance score of a document d for query Q is given by

$$\begin{aligned}score(d,Q) &= \sum_{t\in Q} qtf\cdot w(t,d)\\ &= \sum_{t\in Q} qtf\cdot\frac{1}{tfn+1}\\ &\quad\times\left(tfn\cdot\log_2\frac{tfn}{\lambda}+(\lambda-tfn)\right.\\ &\quad\left.\times\log_2 e+0.5\cdot\log_2(2\pi\cdot tfn)\right),\end{aligned}\tag{7}$$

where λ is the mean and variance of a Poisson distribution, $w(t,d)$ is the weight of document d for query term t and *qtf* is the query term frequency.

The normalised term frequency *tfn* is given by the *normalisation* 2:

$$tfn = tf\cdot\log_2\left(1+c\cdot\frac{avg\_l}{l}\right)\quad (c>0),\tag{8}$$

where l is the document length, *avg_l* is the average document length in the whole collection, *tf* is the original term frequency and c is a free parameter. It is automatically estimated by the tuning method proposed in [6]. This method assumes a constant optimal normalisation effect across collections, and applies the parameter setting such that it gives this constant. The first row of Table 2 provides the applied c value for the three types of queries.

Table 2
The settings of the free parameters for different types of queries

Parameter	Short query	Normal query	Long query
c of PL2	5.90	1.61	1.73
b of BM25	0.09	0.25	0.64

As one of the most well-established IR systems, Okapi uses BM25 to perform the document ranking, where the *idf* factor $w^{(1)}$ is normalised as follows [14]:

$$w(t,d) = w^{(1)}\frac{(k_1+1)tf}{K+tf}\frac{(k_3+1)qtf}{k_3+qtf},\tag{9}$$

where $w(t,d)$ is the weight of document d for query term t. The sum of $w(t,d)$ of the query terms gives the final weight of document d. K is given by $k_1((1-b)+b(l/avg\_l))$, where l and *avg_l* are the document length and the average document length in the collection, respectively. For the parameters k_1 and k_3, we use the standard setting of [15], i.e. $k_1 = 1.2$ and $k_3 = 1000$. *qtf* is the number of occurrences of a given term in the query and *tf* is the within document frequency of the given term. b is the free parameter of BM25's term frequency normalisation component. Similar to the parameter c of normalisation 2, it is estimated by the method provided in [6]. The second row of Table 2 provides the applied b values in all reported experiments.

3.2. Discussion of results

In Table 3, we summarise the results of the linear correlations of the predictors with average precision. From the results, we could derive the following observations:

- Query length (see Definition 1) does not have a significant linear correlation with average precision, except using PL2 for normal queries, which gives a statistically significant by still very weak correlation. This might be due to the fact that the length of queries of the same type are very similar (see $Var(ql)$ in Table 1). To check the assumption, we computed the correlation of average precision with the length of a mixture of three types of queries. Thus, we had $249\times 3 = 747$ observations of both average precision and query length. Measuring the correlation, we obtained $r = 0.0081$ and a p-value of 0.8241, which again indicates an insignificant correlation. Therefore, query length seems to be very weakly correlated with average precision.
- $\gamma 1$ (see Definition 2) has significant linear correlation with average precision in all cases. It is also interesting to see that the correlations for normal and long queries are stronger than that for short queries.

Table 3
The correlations r of the predictors with average precision, and the related p-values

	ql	$\gamma 1$	$\gamma 2$	ω	SCS	AvICTF
	PL2, Short					
r	−.1114	**.2501**	.0405	**.3551**	**.4291**	**.4377**
p-value	.0795	**6.634e-5**	.5249	**8.463e-9**	**1.56e-12**	**4.974e-13**
	BM25, Short					
	ql	$\gamma 1$	$\gamma 2$	ω	SCS	AvICTF
r	−.0990	**.2243**	.0151	**.3640**	**.4296**	**.4390**
p-value	.1191	**.0004**	.8130	**3.353e-9**	**1.463e-9**	**4.17e-13**
	PL2, Normal					
	ql	$\gamma 1$	$\gamma 2$	ω	SCS	AvICTF
r	**−.1247**	**.3032**	.0071	**.2246**	**.3038**	**.3080**
p-value	**.0493**	**1.097e-5**	.9112	**.0004**	**1.042e-6**	**7.273e-7**
	BM25, Normal					
	ql	$\gamma 1$	$\gamma 2$	ω	SCS	AvICTF
r	−.1223	**.2981**	.0059	**.2119**	**.2878**	**.2912**
p-value	.0539	**1.686e-6**	.9266	**.0008**	**3.939e-6**	**2.985e-6**
	PL2, Long					
	ql	$\gamma 1$	$\gamma 2$	ω	SCS	AvICTF
r	−.0608	**.3038**	.0841	.1206	**.2511**	**.2670**
p-value	.3391	**1.044e-6**	.1857	.0574	**6.168e-5**	**1.967e-5**
	BM25, Long					
	ql	$\gamma 1$	$\gamma 2$	ω	SCS	AvICTF
r	−.0390	**.2878**	.0767	.0975	**.2161**	**.2461**
p-value	.5401	**2.55e-6**	.2278	.1249	**.0006**	**8.699-5**

Each predictor corresponds to a column in the table. The results are given separately with respect to the three types of queries and the use of two different document weighting models. Significant correlations are shown in bold. The test queries are the topics used in TREC2004 Robust Track.

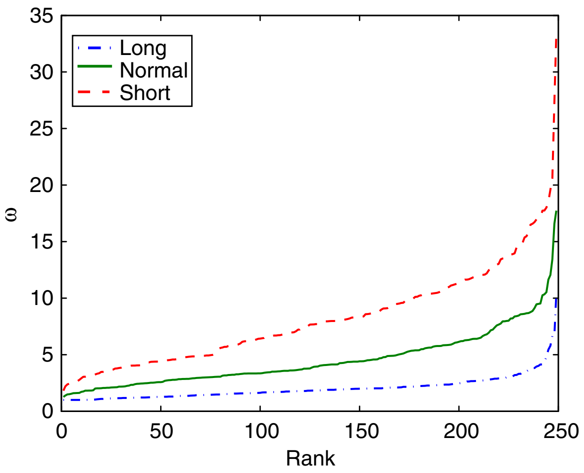

Fig. 1. The ranked ω values in ascending order for the three types of queries.

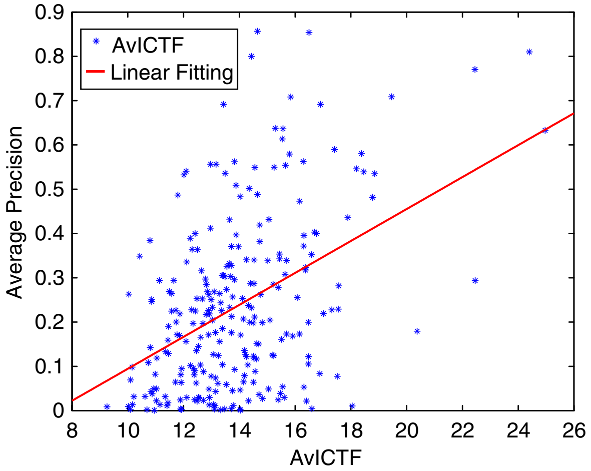

Fig. 2. The linear correlation of AvICTF with average precision using BM25 for short queries. There is a strong linear correlation between these two measures, which is $r = .4390$.

- The effectiveness of $\gamma 2$ (see Definition 3) is not as good as $\gamma 1$. Its correlation with average precision is not significant in all cases.
- For ω, the query scope (see Definition 4), its linear correlation with average precision, is significant for short and normal queries. However, its effectiveness decreases with the increase of query length. Perhaps this is because when queries are getting longer, the query scope tends to be stable. Fig. 1 supports this assumption. We can see that the ω of normal and long queries are clearly more stable than those of short queries.
- The simplified clarity score (SCS, see Definition 5) has significant linear correlation with average precision in all circumstances. However, similar to the query scope, when the query length increases, the correlation gets weaker, although still statistically significant.
- As expected, the average inverse collection term frequency (AvICTF, see Definition 6) achieves a similar performance to SCS. For short queries, the use of BM25 results in the highest linear

correlation among all the predictors (the linear fitting is given in Fig. 2).

In general, among the six proposed predictors, AvICTF and SCS are the most effective for short queries, while $\gamma 1$, AvICTF and SCS are the most effective for normal and long queries. For all the three types of queries, $\gamma 1$ is more effective than $\gamma 2$ in inferring query performance. Moreover, since ω was proposed for Web IR [11], and SCS and AvICTF are more effective than ω, we suggest that both SCS and AvICTF could be applied for Web IR. Note that, although some previous works found that query length affects the retrieval performance [5,6], it seems that query length is not a good predictor, at least on the collection used in our experiments. In summary, query type has a strong impact on the effectiveness of the predictors. The correlation of a predictor with average precision varies for diverse query types.

Finally, we found that the use of two different document weighting models, i.e. PL2 and BM25, does not affect the correlations of the proposed predictors with average precision. Both models produce very similar correlation values.

4. Non-parametric correlation of the predictors with average precision

In this section, instead of the linear correlation, we check the non-parametric correlations of the predictors with average precision. An appropriate measure for the non-parametric test is the Spearman's rank correlation [16]. In this paper, we denote Spearman's correlation between variables X and Y as $rs(X, Y)$.

The test data and experimental setting for checking Spearman's correlation are the same as the previous section. As shown in Table 4, the results are very similar to the linear correlations provided in Table 3. AvICTF and SCS are again the most effective predictors for short queries. Also, $\gamma 1$, AvICTF and SCS seem to be the most effective predictors for normal and long queries. Moreover, the predictors are generally slightly less correlated with the average precision obtained using BM25 than that obtained using PL2. Again, the difference of correlations with the use of both document weighting models is usually marginal. Finally, $\gamma 1$ is still more effective than $\gamma 2$.

Table 4
Spearman's correlations rs of the predictors with average precision, and the related p-values

	PL2, Short					
	ql	$\gamma 1$	$\gamma 2$	ω	SCS	AvICTF
rs	.0528	**.2663**	.0963	**.2639**	**.3528**	**.3664**
p-value	.4055	**2.745e-5**	.1294	**3.243e-5**	**2.779e-8**	**7.942e-9**
	BM25, Short					
	ql	$\gamma 1$	$\gamma 2$	ω	SCS	AvICTF
rs	.0724	**.2345**	.0683	**.2686**	**.3478**	**.3672**
p-value	.2544	**.0002**	.2818	**2.239e-5**	**4.326e-8**	**7.371e-9**
	PL2, Normal					
	ql	$\gamma 1$	$\gamma 2$	ω	SCS	AvICTF
rs	**−.1744**	**.3152**	−.0016	**.2727**	**.3366**	**.3094**
p-value	**.0060**	**6.944e-7**	.9799	**1.755e-5**	**1.153e-7**	**1.101e-6**
	BM25, Normal					
	ql	$\gamma 1$	$\gamma 2$	ω	SCS	AvICTF
rs	**−.1678**	**.3110**	.0008	**.2645**	**.3301**	**.3050**
p-value	**.0082**	**9.682e-7**	.9897	**3.12e-5**	**2.013e-7**	**1.567e-6**
	PL2, Long					
	ql	$\gamma 1$	$\gamma 2$	ω	SCS	AvICTF
rs	−.0830	**.3042**	.0577	**.1459**	**.2763**	**.2882**
p-value	.1913	**1.662e-6**	.1857	**.0216**	**6.168e-5**	**1.967e-5**
	BM25, Long					
	ql	$\gamma 1$	$\gamma 2$	ω	SCS	AvICTF
rs	−.0505	**.2987**	.0646	**.1309**	**.2481**	**.2371**
p-value	.4267	**2.55e-6**	.3090	**.0393**	**9.325e-5**	**1.671e-5**

Each predictor corresponds to a column in the table. The results are given separately with respect to the three types of queries and the use of two different document weighting models. Significant correlations are shown in bold. The test queries are the topics used in TREC2004 Robust Track.

We also compare $rs(SCS, AP)$ with the $rs(CS, AP)$ for the TREC7&8 and TREC4 ad hoc tasks reported in [1]. *CS* stands for Cronen-Townsend et al.'s clarity score. To do the comparison, we run experiments checking the $rs(SCS, AP)$ values for the queries used in TREC4 and TREC7&8. The test queries for TREC4 are the TREC topics 201–250, which are normal queries as they only consist of the descriptions. The test queries for TREC7&8 are numbered from 351 to 450, which are also used in the TREC2004 Robust Track. There was no experiment for long queries reported

Table 5
Spearman's correlations of clarity score (CS) and SCS with average precision

	TREC7&8 short query		TREC4 normal query	
	rs	*p*-value	*rs*	*p*-value
CS	**.536**	**4.8e-8**	**.490**	**3.0e-4**
SCS(PL2)	**.424**	**2.5e-5**	.252	.0779

For SCS and CS, average precision is obtained using PL2 and Song and Croft's multinomial language model, respectively. For TREC7&8, the queries are of short type. For TREC4, the queries are of normal type as they only consist of descriptions. The data in the first row are taken from [1]. Significant correlations are in bold.

in [1]. The parameter c of normalisation 2 (see Eq. (8)) is again automatically set to 1.64 in our experiments for TREC4.

Regarding the generation of average precision, Cronen-Townsend et al. apply Song and Croft's multinomial language model for CS [17], and we apply PL2 for SCS. Since $rs(SCS, AP)$ is stable for statistically diverse document weighting models, i.e. PL2 and BM25 (see Table 4), we believe that the use of the two different document weighting models will not considerably affect the comparison.

Table 5 compares $rs(SCS, AP)$ with the $rs(CS, AP)$ reported in [1]. We can see that for normal queries, $rs(CS, AP)$ is clearly higher than $rs(SCS, AP)$. However, for short queries, although $rs(CS, AP)$ is larger than $rs(SCS, AP)$, the latter is still a significant high correlation.

In summary, SCS is effective in inferring the performance of short queries. Since the actual queries on the World Wide Web are usually very short, SCS can be useful for Web IR, or for other environments where queries are usually short. Moreover, SCS is very practical as the cost of its computation is indeed insignificant. However, comparing with CS, SCS seems to be moderately weak in inferring the performance of longer queries, including normal queries, although the obtained $rs(SCS, AP)$ values are still significant according to the corresponding p-values.

In addition, we have also run experiments on the WT10G collection [18]. Results show that the predictors have relatively weak correlations with average precision on this collection. This concurs with the conclusion in [8] that the clarity score is not effective in inferring query performance on the WT10G collection. Future research is needed to understand the reason of the relatively low correlations on this collection.

The moderately weak correlations of SCS with average precision for longer queries might be due to the fact that the maximum likelihood of the query model ($P_{ml}(w|Q)$) is not reliable when the query length increases. As mentioned before, the effectiveness of those predictors, which are positively correlated with the query length, decreases as the query gets longer. Therefore, we might be able to increase the correlation by smoothing the query model, which is directly related to the query length. We will discuss this issue in the next section.

5. Smoothing the query model of SCS

In this section, we present a method for smoothing the query model of SCS. For the estimation of the query model $P(w|Q)$, instead of introducing the document model by a total probability formula [1], we model the *qtf* density of query length *ql* directly, so that the computation of SCS does not involve the use of relevance scores. Note that *qtf* is the frequency of the term in the given query Q.

Let us start with assuming an increasing *qtf* density of query length *ql*, then we would have the following density function:

$$\rho = C \cdot ql^{\beta}, \tag{10}$$

where ρ is the density and C is a constant of the density function. The exponential β should be larger than 0. In [4], this density function has been applied for the term frequency normalisation, where β is negative. However, in our case, empirically, an appropriate value is $\beta = 0.5$.

Let the average query length be the interval of the integral of ρ, we then have the following

Table 6
Spearman's correlation of SCS with average precision for different types of queries with and without the use of the smoothing function

Task	Query type	*rs*	*p-value*	*v*	*rs$_s$*	*p-value$_s$*
TREC7&8	Short	**.4236**	**.0089**	e-5	**.4268**	**2.471e-5**
TREC7&8	Normal	**.2721**	**.0068**	2.5e-4	**.3017**	**.0027**
TREC7&8	Long	**.2668**	**.0079**	2.5e-4	**.3002**	**.0028**
TREC4	Normal	.2520	.0779	5e-5	**.2847**	**.0463**

rs$_s$ and *p-value$_s$* stand for Spearman's correlation with the use of the smoothing function and the corresponding p-value, respectively. Average precision is obtained using PL2. Significant correlations are in bold.

smoothing function:

$$qtfn = qtf \cdot \int_{ql}^{ql+avg\_ql} \rho d(ql) = qtf \cdot v \cdot ((ql + avg\_ql)^{1.5} - ql^{1.5}), \qquad (11)$$

where *qtfn* is the smoothed query term frequency *qtf*. Replacing *qtf* with *qtfn* in Definition 4, we will obtain the smoothed query model. *avg_ql* is the average query length. *v* is a free parameter. It is empirically set in our experiments (see the fifth column of Table 6).

Table 6 summarises the obtained Spearman's correlation values using the smoothing function. As mentioned in the previous section, the use of two different document weighting models produce similar correlation values, therefore, we only experiment on PL2 in this section. For short queries, no significant effect is noticed. However, for normal and long queries, the rs_s values are considerably larger than the values obtained without the use of the smoothing function. It is also encouraging to see that for TREC4, compared to the *rs* value obtained without smoothing, the obtained rs_s value using the smoothing function is significant. Therefore, the effectiveness of SCS has improved for normal and long queries by smoothing the query model.

6. Conclusions

We have studied six pre-retrieval predictors for query performance. The predictors can be generated before the retrieval process takes place, which is more practical than current approaches to query performance prediction. We have measured the linear and non-parametric correlations of the predictors with average precision. According to the results, the query type has an important impact on the effectiveness of the predictors. Among the six proposed predictors, a simplified definition of clarity score (SCS) and the average inverse collection term frequency (AvICTF) have the strongest correlation with average precision for short queries. The standard deviation of *idf* ($\gamma 1$), SCS and AvICTF are the most correlated with average precision for normal and long queries. Also, we have shown that SCS can be improved by smoothing the query model. Taking the complexity of generating a predictor into consideration, SCS, AvICTF and $\gamma 1$ can be useful for practical applications. Moreover, according to the results, the use of two statistically diverse document weighting models does not have an impact on the overall effectiveness of the proposed predictors.

Acknowledgements

This work is funded by the Leverhulme Trust, Grant number F/00179/S. The project funds the development of the Smooth project, which investigates the term frequency normalisation (URL: http://ir.dcs.gla.ac.uk/smooth). The experimental part of this paper has been conducted using the Terrier platform (URL: http://ir.dcs.gla.ac.uk/terrier). We would also like to thank Gianni Amati, as well as the reviewers, for their helpful comments on the paper.

References

[1] S. Cronen-Townsend, Y. Zhou, W.B. Croft, Predicting query performance, in: Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Tampere, Finland, 2002, pp. 299–306.

[2] G. Amati, C. Carpineto, G. Romano, Query difficulty, robustness, and selective application of query expansion, in: Adcances in Information Retrieval, Proceedings of the 26th European Conference on IR Research, ECIR 2004, Sunderland, UK, 2004, pp. 127–137.

[3] J.M. Ponte, W.B. Croft, A language modeling approach to information retrieval, in: Proceedings of the 21th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Melbourne, Australia, 1998, pp. 275–281.

[4] G. Amati, C.J. van Rijsbergen, Probabilistic models of information retrieval based on measuring the divergence from randomness, ACM Trans. Inf. Syst. (TOIS) 20 (4) (2002) 357–389.

[5] C. Zhai, J. Lafferty, A study of smoothing methods for language models applied to ad hoc information retrieval, in: Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, New Orleans, LA, 2001, pp. 334–342.

[6] B. He, I. Ounis, Tuning term frequency normalisation for BM25 and DFR models, in: Proceedings of the 27th European Conference on Information Retrieval (ECIR'05), Santiago de Compostela, Spain, March, 2005, pp. 200–214.

[7] A. Pirkola, K. Jarvelin, Employing the resolution power of search keys, J. Am. Soc. Inf. Sci. Technol. 52 (7) (2001) 575–583.

[8] F. Scholer, H. Williams, A. Turpin, Query association surrogates for web search, J. Am. Soc. Inf. Sci. Technol. 55 (7) (2004) 637–650.

[9] L. Si, J. Callan, Using sampled data and regression to merge search engine results, in: Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Tampere, Finland, 2002, pp. 19–26.

[10] J. Allan, L. Ballesteros, J. Callan, W. Croft, Recent experiments with INQUERY, in: NIST Special Publication 500-236: The Fourth Text REtrieval Conference (TREC-4), Gaithersburg, MD, 1995, pp. 49–63.

[11] V. Plachouras, I. Ounis, G. Amati, C.J. van Rijsbergen, University of Glasgow at the Web track: dynamic application of hyperlink analysis using the query scope, in: Proceedings of the 12th Text REtrieval Conference (TREC 2003), Gaithersburg, MD, 2003, pp. 248–254.

[12] K.L. Kwok, A new method of weighting query terms for adhoc retrieval, in: Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Zurich, Switzerland, 1996, pp. 187–195.

[13] M. DeGroot, Probability and Statistics, second edition, Addison-Wesley, Reading, MA, 1989.

[14] S. Robertson, S. Walker, M.M. Beaulieu, M. Gatford, A. Payne, Okapi at TREC-4, in: NIST Special Publication 500-236: The Fourth Text REtrieval Conference (TREC-4), Gaithersburg, MD, 1995, pp. 73–96.

[15] K. Sparck-Jones, S. Walker, S.E. Robertson, A probabilistic model of information retrieval: development and comparative experiments, Inf. Process. Manage. 36 (2000) 779–840.

[16] J.D. Gibbons, S. Chakraborti, Nonparametric Statistical Inference, Marcel Dekker, New York, 1992.

[17] F. Song, W. Croft, A general language model for information retrieval, in: Proceedings of the 22nd Annual International ACM SIGIR Conference, Berkeley, CA, 1999, pp. 279–280.

[18] D. Hawking, Overview of the TREC-9 web track, in: Proceedings of the Ninth Text REtrieval Conference (TREC-9), Gaithersburg, MD, 2000, pp. 87–94.