Inferring Query Performance Using Pre-retrieval Predictors

Ben He and Iadh Ounis

Department of Computing Science
University of Glasgow
{ben, ounis}@dcs.gla.ac.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 a set of predictors of query performance, which can be generated prior to the retrieval process. The linear and non-parametric correlations of the predictors with query performance are thoroughly assessed on the TREC disk4 and disk5 (minus CR) collections. 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.

1 Introduction

Robustness is an important measure reflecting the retrieval performance of an IR system. It particularly refers to how an IR system deals with poorly-performing queries. As stressed by Cronen-Townsend et. al. [4], poorly-performing queries considerably hurt the effectiveness of an IR system. Indeed, this issue has become important in IR research. For example, in 2003, TREC proposed 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 [4], 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 notion of *query-difficulty* to predict query performance. Their basic idea is that the query expansion weight, which is 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, it is difficult to incorporate these methods into practical applications because they are post-retrieval approaches, involving the time-consuming computation of relevance scores.

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 looks through 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 a set of 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 predictors of query performance. Similar to previous works mentioned in Section 1, we consider the intrinsic statistical features of queries as the predictors and use them in inferring the query performance. Moreover, these features should be computed prior to the retrieval process. The proposed list of predictors is inspired by previous works related to probabilistic IR models, including the language modelling approach [11] and Amati & van Rijsbergen's Divergence From Randomness (DFR) models [3]:

- **Query length.** According to Zhai & Lafferty's work [15], 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 [7].
  For example, the optimal setting for the so-called normalisation 2 in Amati & van Rijsbergen's probabilistic framework is query-dependent [3]. 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 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 [9]. Therefore, the distribution of the $idf(t)$ factors in the composing query terms might be an intrinsic feature that affects the retrieval performance. In this paper, we investigate the following two possible definitions for the distribution of informative amount in query terms:

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 [1]:

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

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 Q respectively.

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

– **Query clarity.** Query clarity refers to the speciality/ambiguity of a query. According to the work by Cronen-Townsend et. al. [4], 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 [4]. 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 4 (SCS): *The simplified query clarity score is given by:*

$$SCS = \sum_{Q} P_{ml}(w|Q) \cdot log_2 \frac{P_{ml}(w|Q)}{P_{coll}(w)} \tag{4}$$

In the above definition, $P_{ml}(w|Q)$ is given by $\frac{qtf}{ql}$. 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 $\frac{tf_{coll}}{token_{coll}}$, 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.
Although the above definition seems simple and naive, it would be very easy to compute. 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.

– **Query scope.** Similar to the clarity score, an alternative indication of the generality/speciality of a query is the size of the document set containing at least one of the query terms. As stressed in [10], the size of this document set is an important property of the query. Following [10], in this work, we define the query scope as follows:

Definition 5 (ω): *The query scope is:*

$$\omega = -\log(n_Q/N) \tag{5}$$

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 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 dependance 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 dependance 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 [5]. We use average precision (AP) 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 $r = 0$ indicates that the two variables are independent. $r > 0$ and $r < 0$ indicates 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 AP indicates that this predictor could be useful to infer the query performance in practical applications.

3.1 Test Data and Settings

The document collection used to test the efficiency of the proposed predictors is the TREC disk4&5 test collections (minus the Congressional Record on disk4). The test queries are the TREC topics 351-450, which are used in the TREC7&8 ad-hoc tasks. For all the documents and queries, the stop-words are removed using a standard list and the 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 AP value of each query, we compute r and the corresponding p-value of the linear dependance between the two variables, i.e. AP and each of the predictors. The AP values of the test queries are given by the PL2 and BM25 term weighting models, respectively. We use two statistically different models in order to check if the effectiveness of the predictors is independent of the used term-weighting models.

PL2 is one of the Divergence From Randomness (DFR) term weighting models developed within Amati & van Rijsbergen's probabilistic framework for IR [3]. Using the PL2 model, the relevance score of a document d for query term t is given by:

$$w(t,d) = tf \cdot \log_2 \frac{tf}{\lambda} + (\lambda + \frac{1}{12 \cdot tf} - tf) \cdot \log_2 e + 0.5 \cdot \log_2(2 \cdot tf) \cdot \frac{1}{tf+1} \quad (6)$$

where λ is the mean and variance of a Poisson distribution.

The within document term frequency tf is then normalised using the *normalisation 2*:

$$tfn = tf \cdot \log_2(1 + c \cdot \frac{avg\_l}{l}), (c > 0) \quad (7)$$

where l is the document length and avg_l is the average document length in the whole collection.

Table 1. The statistics of the length of the three types of queries. avg_ql is the average query length. $Var(ql)$ is the variance of the length of the queries

	Short Query	Normal Query	Long Query
avg_ql	2.42	7.55	21.13
$Var(ql)$	0.42	10.19	55.77

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

Replacing the raw term frequency tf by the normalised term frequency tfn in Equation (6), we obtain the final weight. c is a free parameter. It is automatically estimated by measuring the normalisation effect [7]. The first row of Table 2 provides the applied c value for the three types of queries.

As one of the most well-established IR systems, Okapi uses BM25 to measure the term weight, where the idf factor $w^{(1)}$ is normalised as follows [12]:

$$w(t,d) = w^{(1)} \frac{(k_1+1)tf}{K+tf} \frac{(k_3+1)qtf}{k_3+qtf} \tag{8}$$

where w is the final weight. K is given by $k_1((1-b)+b\frac{l}{avg\_l})$, 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 [14], 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 the normalisation 2, it is estimated by the method provided in [7]. However, due to the "out of range" problem mentioned in [7], we applied a new formula for the normalisation effect (see Appendix). 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 AP. From the results, we could derive the following observations:

- Query length (see Definition 1) does not have a significant linear correlation with AP. 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 AP with the length of a mixture of three types of queries. Thus, we had $100 \times 3 = 300$ observations of both AP and query length. Measuring the correlation, we obtained $r = 0.0585$ and a p-value of 0.3124, which again indicates a very low correlation. Therefore, query length seems to be very weakly correlated with AP.

Table 3. The correlations r of the predictors with AP, and the related p-values. The results are given separately with respect to the three types of queries. Significant correlations are shown in bold. The test queries are the topics used in TREC7&8

	PL2, Short Query					BM25, Short Query				
	ql	$\gamma 1$	$\gamma 2$	ω	SCS	ql	$\gamma 1$	$\gamma 2$	ω	SCS
r	-0.1839	**0.2398**	0.0569	**0.3772**	**0.4484**	-0.1773	0.1860	0.0332	**0.3746**	**0.4208**
p-value	0.0670	**0.0163**	0.5738	**0.0001**	**3.037e-6**	0.0776	0.0639	0.7430	**0.0001**	**1.351e-5**
	PL2, Normal Query					BM25, Normal Query				
	ql	$\gamma 1$	$\gamma 2$	ω	SCS	ql	$\gamma 1$	$\gamma 2$	ω	SCS
r	0.0830	**0.3017**	0.1259	0.1895	**0.2602**	0.0876	**0.2946**	0.1436	0.1629	**0.2293**
p-value	0.4116	**0.0023**	0.2120	0.0590	**0.0089**	0.3862	**0.0029**	0.1542	0.1054	**0.0217**
	PL2, Long Query					BM25, Long Query				
	ql	$\gamma 1$	$\gamma 2$	ω	SCS	ql	$\gamma 1$	$\gamma 2$	ω	SCS
r	0.0543	**0.3227**	**0.3029**	0.0910	**0.2401**	0.0790	**0.2822**	**0.2753**	0.0843	**0.2066**
p-value	0.5915	**0.0011**	**0.0022**	0.3679	**0.0161**	0.4349	**0.0044**	**0.0056**	0.4044	**0.0392**

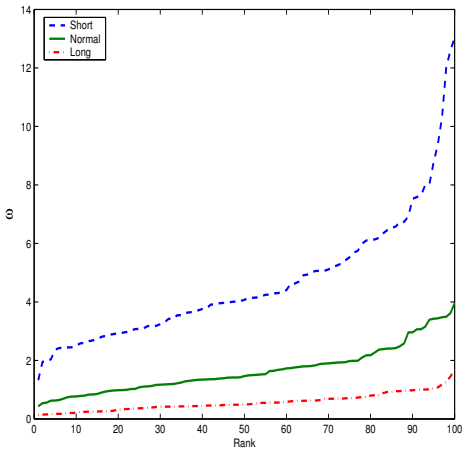

Fig. 1. The ranked ω values in ascending order for the three types of queries

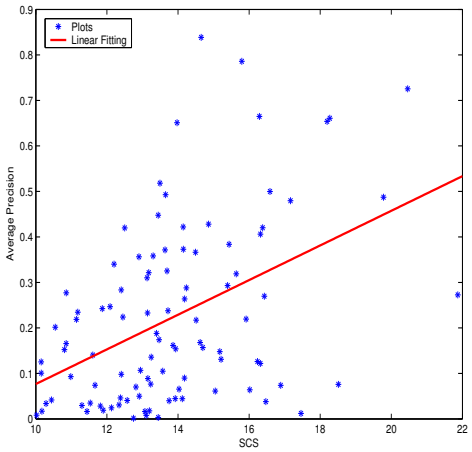

Fig. 2. The linear correlation of SCS with AP using PL2 for short queries

- $\gamma 1$ (see Definition 2) has significant linear correlation with AP in all cases except for the short queries when BM25 is used. It is also interesting to see that the correlations for normal and long queries are stronger than that for short queries.
- The linear correlation of $\gamma 2$ (see Definition 3) with AP is only significant for long queries. Also, the correlation is positive, which indicates that a larger gap of informative amount between the query terms would result into a higher AP. Moreover, the results show that on the used test collection, $\gamma 1$ is more effective than $\gamma 2$ in inferring query performance.
- For ω, the query scope (see Definition 4), its linear correlation with AP is only significant for short queries. Perhaps this is because when queries are getting longer, the query scope tends to be stable. Figure 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 AP in all circumstances. For the short queries, the use of PL2 results in the highest linear correlation among all the predictors (the linear fitting is given in Figure 2). However, when the query length increases, the correlation gets weaker.
– Moreover, it seems that the predictors are generally less effective when BM25 is used as the term-weighting model. For the same predictor, the AP given by BM25 is usually less correlated with it than the AP given by PL2.

In summary, query type has a strong impact on the effectiveness of the predictors. Indeed, the correlation of a predictor with AP varies for diverse query types. For short queries, SCS and ω have strong linear correlations with AP. For normal queries, $\gamma 1$ has moderately significant linear correlation with AP. For long queries, $\gamma 1$ and $\gamma 2$ have significant linear correlations with AP.

In general, among the five proposed predictors, SCS is the most effective one for short queries, and $\gamma 1$ is the most effective one 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 [10] and SCS is more effective than ω, SCS could also be a good option for Web IR. Note that, although some previous works found that query length affects the retrieval performance [7, 15], it seems that query length is not significantly correlated with AP, at least on the used collection.

Finally, we found that, in most cases, the predictors are slightly less correlated with the AP obtained using BM25 than that obtained using PL2. The difference of correlations is usually marginal, except for short queries, where $\gamma 1$ is significantly correlated with the AP obtained using PL2, but not BM25. Overall, the use of different term-weighting models does not considerably affect the correlations of the proposed predictors with AP.

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 AP. An appropriate measure for the non-parametric test is the Spearman's rank correlation [6]. In this paper, we denote the Spearman's correlation between variables X and Y as $rs(X, Y)$.

The test data and experimental setting for checking the 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. SCS is again the most effective predictor, which has significant Spearman's correlations with AP for the three types of queries. Also, $\gamma 1$ seems to be the most effective predictor for normal and long queries. Moreover, the predictors are generally slightly less correlated with the AP obtained using BM25 than that obtained using PL2. Again, the difference of correlations is usually marginal, except the correlation of $\gamma 1$ with short queries, where $rs(\gamma 1, AP)$ for PL2 is significant, while $rs(\gamma 1, AP)$ for

Table 4. The Spearman's correlation rs of the predictors with AP for three types queries using PL2 and BM25 respectively. Significant correlations are shown in bold. The test queries are the topics used in TREC7&8

	PL2, Short Query					BM25, Short Query				
	ql	$\gamma 1$	$\gamma 2$	ω	SCS	ql	$\gamma 1$	$\gamma 2$	ω	SCS
rs	-0.0476	**0.2141**	0.0279	**0.3627**	**0.4236**	-0.0354	0.1449	-0.0217	**0.3393**	**0.3752**
p-value	0.6359	**0.0331**	0.7794	**0.0003**	**2.504e-5**	0.7243	0.1497	0.8280	**0.0007**	**0.0002**
	PL2, Normal Query					BM25, Normal Query				
	ql	$\gamma 1$	$\gamma 2$	ω	SCS	ql	$\gamma 1$	$\gamma 2$	ω	SCS
rs	-0.0646	**0.3627**	0.1240	0.1790	**0.2721**	-0.0640	**0.3439**	0.1129	0.1647	**0.2583**
p-value	0.5203	**0.0003**	0.2183	0.0748	**0.0068**	0.5242	**0.0006**	0.2615	0.1013	**0.0102**
	PL2, Long Query					BM25, Long Query				
	ql	$\gamma 1$	$\gamma 2$	ω	SCS	ql	$\gamma 1$	$\gamma 2$	ω	SCS
rs	0.0132	**0.3272**	**0.2236**	0.1324	**0.2668**	-2.1e-05	**0.2972**	0.1875	0.1544	**0.2556**
p-value	0.8958	**0.0011**	**0.0266**	0.1861	**0.0079**	0.9998	**0.0030**	0.0628	0.1238	**0.0110**

BM25 is not. Finally, $\gamma 1$ is still more effective than $\gamma 2$ as a query performance predictor.

We also compare $rs(SCS, AP)$ with the $rs(CS, AP)$ for the TREC7&8 and TREC4 ad-hoc tasks reported in [4]. CS stands for Cronen-Townsend et. al.'s clarity score. To do the comparison, besides $rs(SCS, AP)$ for TREC7&8 provided in Table 4, we also run experiments checking the $rs(SCS, AP)$ values for the queries used in TREC4. The test queries for TREC4 are the TREC topics 201-250, which are normal queries as they only consist of the descriptions. There was no experiment for long queries reported in [4]. The parameter c of the normalisation 2 (see Equation (7)) is also automatically set to 1.64 in our experiments for TREC4.

Regarding the generation of AP, Cronen-Townsend et. al. apply Song & Croft's multinomial language model for CS [13], and we apply PL2 for SCS. Since $rs(SCS, AP)$ is stable for statistically diverse term-weighting models, i.e. PL2 and BM25 (see Table 4), we believe that the use of the two different term-weighting models won't considerably affect the comparison.

Table 5 compares $rs(SCS, AP)$ with the $rs(CS, AP)$ reported in [4]. 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.

The moderately weak correlations of SCS with AP 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 effective-

Table 5. The Spearman's correlations of clarity score (CS) and SCS with AP. For SCS and CS, AP is obtained using PL2 and Song & 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 [4]

	TREC7&8 Short Query		TREC4 Normal Query	
	rs	p-value	rs	p-value
CS	0.536	4.8e-8	0.490	3.0e-4
SCS	0.424	2.5e-5	0.252	0.0779

ness 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 [4], 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 query Q.

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

$$\rho = C \cdot ql^{\beta} \tag{9}$$

where ρ is the density and C is a constant of the density function. The exponential β should be larger than 0. 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 smoothing function:

$$qtfn = \int_{ql}^{ql+avg\_ql} \rho d(ql) = \nu \cdot ((ql + avg\_ql)^{1.5} - ql^{1.5}) \tag{10}$$

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

Table 6 summarises the obtained $rs(SCS, AP)$ values using the smoothing function. For short queries, no significant effect is noticed. However, for normal and long queries, the rs values are considerably larger than the values obtained without the use of the smoothing function (see Table 4). It is also encouraging to see that for TREC4, compared to the rs value in Table 5, the obtained rs value using the smoothing function is significant. Therefore, the effectiveness of SCS has improved for normal and long queries by smoothing the query model.

Table 6. The Spearman's correlation of SCS with AP for different types of queries using the smoothing function. AP is obtained using PL2

Task	Query Type	ν	rs	p-value
TREC7&8	Short	e-5	0.4268	2.471e-5
TREC7&8	Normal	2.5e-4	0.3017	0.0027
TREC7&8	Long	2.5e-4	0.3002	0.0028
TREC4	Normal	5e-5	0.2847	0.0463

6 Conclusions and Future Work

We have studied a set of 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 AP. According to the results, the query type has an important impact on the effectiveness of the predictors. Among the five proposed predictors, a simplified definition of clarity score (SCS) has the strongest correlation with AP for short queries. $\gamma 1$ is the most correlated with AP 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 and $\gamma 1$ can be useful for practical applications. Moreover, according to the results, the use of two statistically diverse term-weighting models does not have an impact on the overall effectiveness of the proposed predictors.

In the future, we will investigate improving the predictors using various methods. For example, we plan to develop a better smoothing function for the query model of SCS. We will also incorporate the proposed predictors into our query clustering mechanism, which has been applied to select the optimal term-weighting model, given a particular query [8]. The use of better predictors would hopefully allow the query clustering mechanism to be improved. As a consequence, the query-dependence problem of the term frequency normalisation parameter tuning, stressed in [7], could be overcome.

7 Acknowledgments

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 framework (EPSRC, grant GR/R90543/01, URL: http://ir.dcs.gla.ac.uk/terrier). We would also like to thank Gianni Amati for his helpful comments on the paper.

References

1. J. Allan, L. Ballesteros, J. Callan, W. Croft. Recent experiments with INQUERY. In *Proceedings of TREC-4*, pp. 49-63, Gaithersburg, MD, 1995.

2. G. Amati, C. Carpineto, G. Romano. Query difficulty, robustness, and selective application of query expansion. In *Proceedings of ECIR'04*, pp. 127-137, Sunderland UK, 2004.
3. G. Amati and C. J. van Rijsbergen. Probabilistic models of information retrieval based on measuring the divergence from randomness. In *TOIS*, 20(4), pp. 357-389, 2002.
4. S. Cronen-Townsend, Y. Zhou, W. B. Croft. Predicting query performance. In *Proceedings of SIGIR'02*, pp. 299-306, Tampere, Finland, 2002.
5. M. DeGroot. *Probability and Statistics*. Addison Wesley, 2nd edition, 1989.
6. J. D. Gibbons and S. Chakraborti. *Nonparametric statistical inference*. New York, M. Dekker, 1992.
7. B. He and I. Ounis. A study of parameter tuning for term frequency normalization. In *Proceedings of CIKM'03*, pp. 10-16, New Orleans, LA, 2003.
8. B. He and I. Ounis. A query-based pre-retrieval model selection approach to information retrieval. In *Proceedings of RIAO'04*, pp. 706-719, Avignon, France, 2004.
9. A. Pirkola and K. Jarvelin. Employing the resolution power of search keys. *JASIST*, 52(7):575-583, 2001.
10. 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 TREC2003*, pp. 248-254, Gaithersburg, MD, 2003.
11. J. M. Ponte and W. B. Croft. A language modeling approach to information retrieval. In *Proceedings of SIGIR'98*, pp. 275-281, Melbourne, Australia, 1998.
12. S. Robertson, S. Walker, M. M. Beaulieu, M. Gatford, A. Payne. Okapi at TREC-4. In *Proceedings of TREC-4*, pp. 73-96, Gaithersburg, MD, 1995.
13. F. Song and W.Croft. A general language model for information retrieval. In *Proceedings of SIGIR'99*, pp. 279-280, Berkeley, CA, 1999.
14. K. Sparck-Jones, S. Walker, S. Robertson. A probabilistic model of information retrieval: Development and comparative experiments. *IPM*, 36(2000):779-840, 2000.
15. C. Zhai and J. Lafferty. A study of smoothing methods for language models applied to ad hoc information retrieval. In *Proceedings of SIGIR'01*, pp. 334-342, New Orleans, LA, 2001.

Appendix

The new formula for the normalisation effect NE_D is the following:

$$NE_D = Var\big(\frac{NE_{d_i}}{NE_{d,max}}\big), d_i \in D \tag{11}$$

where D is the set of documents containing at least one of the query terms. d_i is a document in D. $NE_{d,max}$ is the maximum NE_{d_i} in D. Var denotes the variance. NE_{d_i} is given by:

$$\frac{1}{(1-b)+b\cdot\frac{l}{avg\_l}} \tag{12}$$

where l is the length of the document d_i. b is a free parameter of BM25. avg_l is the average document length in the whole collection.