# Co-occurrence Based Predictors for Estimating Query Difficulty

**Hazra Imran and Aditi Sharan**

**School of Computer and Systems Sciences**

**Jawaharlal Nehru University,**

**New Delhi, India**

*Abstract*— **Query difficulty prediction aims to identify, in advance, how reliably an information retrieval system will perform when faced with a particular user request. The prediction of query difficulty level is an interesting and important issue in Information Retrieval (IR) and is still an open research. In order to appreciate importance of query difficulty prediction we present an example. Information Retrieval (IR) is the Science of searching the relevant documents based on user's need and a way towards discovering knowledge from text data. User's needs are often expressed in terms of query. It has been observed that there is a word mismatch problem while matching user's query to the documents. This is because users and authors of documents do not use same vocabulary. Query expansion/reformulation is a method to overcome such mismatch in terminology. Query expansion (QE) has become a well known technique that has been shown to improve average retrieval performance. However despite extensive research QE does not provide consistent gains over different query sets and collections. Therefore this technique has not been used in many operational systems as it may degrade performance of individual queries. A thorough investigation into robustness of query expansion is required in order to ensure reliability of query expansion for individual queries. It is well-known in the Information Retrieval community that methods such as query expansion can help "easy" queries but are detrimental to "hard" queries If the performance of queries can be predicted before retrieval then specific measures can be taken to improve the overall performance of the system. In this paper we do thorough investigations of various query difficulty predictors l and suggest two new query predictorsl based on co-occurrence of query terms. To evaluate the predictors, we have experimented on standard TREC collections. Our work is significant as it is a step towards judging reliability and robustness of query processing operations such as query expansion.**

**Keywords- Pre-retrieval query predictors, query difficulty, information retrieval**

## I. INTRODUCTION

Estimation of query difficulty is now being considered as an important capability for IR systems and has been investigated under different names such as query prediction, query-difficulty or query-ambiguity. The ability to predict the performance of a query in advance would enable information retrieval systems to respond more efficiently to user requests. If the performance of queries can be predicted before retrieval specific measures can be taken to improve the overall performance of the system. An important outcome of query prediction is that it allows differentiating highly-performing queries from poorly-performing queries. Our observation shows that queries that are answered well by IRS are those whose keywords agree on most of the returned documents. Difficult queries (i.e. queries for which the IRS will return mostly irrelevant documents) are those where either all keywords agree on all results or cannot agree on them. The former is usually the case where the query contains one rare keyword that is not representative of the whole query and the rest of the query terms appear together in many irrelevant documents. As stressed by Cronen-Townsend et. al. [18], poorly-performing queries considerably hurt the effectiveness of an IR system. It is well-known in the Information Retrieval community that methods such as query expansion can help "easy" queries but are detrimental to "hard" queries [3].Use of reliable query performance predictors can be a step towards determining for a specific query the most optimal corresponding retrieval strategy. For example, in [6], the use of query performance predictors allowed to devise a selective decision methodology avoiding the failure of query expansion.

Specifically query estimation is useful for improving information retrieval in several ways:-

(1) *Selective automatic query expansion*-Automatic query expansion (AQE) is a method for improving retrieval by adding terms to the query, based on frequently appearing terms in the top documents retrieved by the original query. However, this technique works only for easy queries, i.e., when the IRS is able to rank high the relevant documents. If this is not the case, AQE will add irrelevant terms, causing a decrease in performance. Thus, it is not beneficial to use AQE for every query. Instead, it is advantageous to have a switch that will estimate when AQE will improve retrieval, and when it would be detrimental to it.

*(2)Detecting missing content*- There are some queries for which all the results returned by the IRS are irrelevant. Queries for which there is no relevant document in the document collection are defined as missing content queries (MCQs). A useful application of the query predictor is to

978-0-7695-4257-7/10 $26.00 © 2010 IEEE
DOI 10.1109/ICDMW.2010.81

identify MCQs, which will enable user to know if the document collection contains any answers to his query.

Over last few years, a number of query prediction approaches have been proposed, each having their own pros and cons. In this paper we investigate these approaches and propose a method of estimating query difficulty based on co-occurrence of query terms.

The paper is organized as follows. Section II deals with query prediction taxonomy. Section III presents related work along with some existing query predictors. In section IV we discuss about different dimensions along which query can be estimate4d.. In section V we present our approach along with motivation for the work. In section VI experiments are presented. Finally section VII concludes the paper.

## II. QUERY TAXONOMY

Over last few years, a number of query performance prediction approaches have been proposed. These can be divided into three broad categories: pre-retrieval predictors, post-retrieval predictors, and learning predictors

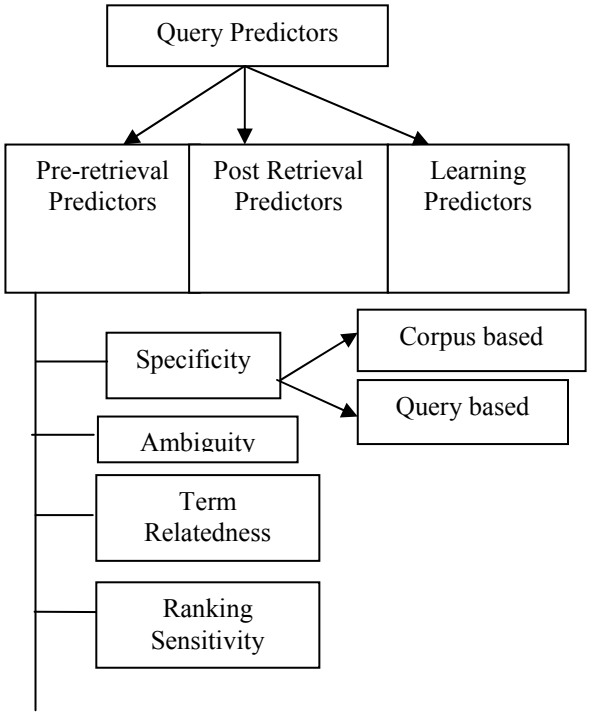


Figure 1 : Hierarchy of Predictors

In this paper we focus on pre-retrieval predictors. Pre-retrieval predictors estimate the performance of a query before the retrieval stage is reached and are, thus, independent of the ranked list of results; essentially, they are search-independent. .They can be calculated without needing to first evaluate the query and obtain an answer set, and are therefore more efficient than post-retrieval predictors.

## III. RELATED WORK

Estimation of query difficulty has been recently recognized as an important capability for IR systems and has been investigated under different names such as query-difficulty or query-ambiguity. Some researchers have used IDF-related (inverse document frequency) weights and information theoretic measures as predictors, with a mixed success. Initial success at addressing this task was demonstrated by the clarity score method proposed in [18].Tomlinson et al. [19] adopted the weighted average IDF of the query terms for predicting. He and Ounis [7] proposed a predictor based on the standard deviation of the IDF of the query terms. Their predictor is based on the assumption that the terms of a poorly-performing query tend to have similar idf values. Plachouras et al. [16] experimented with the idf-based predictor and with a similar variant based on the average inverse collection term frequency (avICTF) of the query terms. Amati [6] proposed to use the KL-divergence between a query term's frequency in the top retrieved documents and the frequency in the whole collection, which is very similar to the definition of the clarity score. Diaz and Jones [4] used meta-data, attached to documents in the form of time stamps, to measure the distribution of documents retrieved over the time domain. They showed that using this temporal distribution together with the content of the documents retrieved can improve the prediction of average precision for a query. Carmel et al. [2] found that the distance measured by the Jensen-Shannon divergence between the retrieved document set and the collection is significantly correlated to average precision. Vinay et al. [20] proposed four measures to capture the geometry of the top retrieved documents for prediction. Other researchers have applied machine-learning techniques for prediction. For example, Elad Yom-Tov et al. [23] proposed a histogram-based predictor and a decision tree based predictor. Kwok et al. [12, 13] built a query predictor using support vector regression. Jensen et al. [10] trained a regression model with manually labeled queries to predict precision at the top 10 documents .Mandl et al [15], in their work focus if some query features could be correlated to system performance. Focusing on documents instead of queries, Karlgren [11] also used linguistic features in order to characterize documents in IR. In [3] several classes of topic failures were drawn manually, but no elements were given on how to assign automatically a topic to a category.

### *A. Existing Pre-Retrieval Predictors*

In this sub-section we will present all the 11 existing query predictors on which we have performed experiments and compared their performance with our predictor. Note that each predictor gives a value which in some way gives amount of information present in query. This mean lower value of predictor implies a difficult query; the query becomes easier as value estimated by the predictor increases.

**Predictor 1 (query length):** Given a query $Q(t_1, \ldots, t_n)$, the *length of the query is the number of non-stop words in the query.*

**Predictor 2 (inf_amt_in_query_terms1):** Given *a query Q, the distribution of informative amount in its composing terms is represented as:*

$$\inf\_amt\_in\_query\_terms1 = \sigma_{idf}$$

*where* $\sigma_{idf}$ *is the standard deviation of the idf of the terms in Q. where idf*, can be calculated as

$$idf(c) = \log_{10}(N / N_c)$$

Where
N = number of documents in the corpus
$N_c$ = number of documents in the corpus that contain c

In general, each term is 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. Therefore, the distribution of the *idf*(*t*) factors in the composing query terms might be an intrinsic feature that affects the retrieval performance.

Another possible predictor based on distribution of informative amount in the query terms is

**Predictor 3 (inf_amt_in_query_terms2):** *Given a query Q, the distribution of informative amount in its composing terms, is represented as:*

$$\inf\_amt\_in\_query\_terms2 = \frac{idf_{\max}}{idf_{\min}}$$

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

**Predictor 4 (query_clarity)** According to the work by Cronen-Townsend et. al. [18], 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. In their definition, the clarity of a query is the sum of the Kullback- Leibler divergence of the query model from the collection model.

S*implified query clarity score is given by*

$$\text{query\_clarity} = \sum_{Q} P_{mt}(w \mid Q).\log_2 \frac{P_{mt}(w \mid Q)}{P_{coll}(w)}$$

Where, $P_{mt(w|Q)}$ is given by $\frac{q_{tf}}{q_l}$ . It is the maximum likelihood 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. *Pcoll*(*w*) 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. Query clarity refers to the specificity of a query and gives good result in most of the cases. However, this definition involves the computation of relevance scores for the query model, which is time-consuming.

**Predictor 5 (query_scope) :** The query scope is

$$query\_scope = -\log(n_Q / N)$$

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. An alternative indication of the generality/specificity of a query includes the size of the document set containing at least one of the query terms

**Predictor 6 (SQWC):** Given a query $Q(t_1, \ldots, t_n)$, the similarity score of the query with corpus can be defined as:

$$SQWC = \sum (1 + \ln(f_{c,t})) \times \ln(1 + \frac{N}{f_t})$$

Where *N* is the total number of documents in the collection *C*, $f_{c,t}$ is the frequency of term *t* in the collection, and $f_t$ is the number of documents that contain term *t*. In this version of the metric the contributions of the collection term frequencies and inverse document frequencies of all query terms are summed. Such a process will be biased towards longer queries. Therefore this predictor can be normalized leading to another query predictor NSQWC.

**Predictor 7 (NSQWC):** We define the score as the SQWC score, divided by the query length, where only terms in the collection vocabulary are considered:

$$NSQWC = \frac{SQWC}{|Q|_{t \in V}}$$

where *V* is the vocabulary (all unique terms in the collection).

An alternative approach is to choose the maximum SQWC score of any query term instead of normalized SQWC score. The intuition behind this approach is that, since search queries tend to be short, if at least one of the terms has a high score then the query as a whole can be expected to perform well:

**Predictor 8 (MSQWC):** It considers that the performance of a query is determined by the "best" term in the query—the term that has the highest SQWC score:

$$MSQWC = \max_{\forall t \in V}\left[(1+\ln(f_{c,t}))\times \ln(1+\frac{N}{f_t})\right]$$

It may be noted that it is not rare to encounter a query term $t$ that is missing in $V$. For simplicity, such terms are assigned with 0 scores in the query.

The above predictors (predictor 1-8) explored the features of the corpus, such as the frequency with which terms occur in the collection as a whole. The next predictors are concerned with the distribution of query terms over the collection, taking into account the variability of term-occurrences within individual documents. It is based on the hypothesis that if the standard deviation of term weights is high, then the term should be easier to evaluate. This is because the retrieval system will be able to differentiate with higher confidence between answer documents.

In general, each query term $t$ can be assigned with a weight value $w_{d,t}$ if it occurs in document $d$. From all the documents that contain term $t$ in a collection, the distribution of $t$ can then be estimated. We use a simple *TF.IDF* approach to compute the term weight, $w_{d,t}$ within a document :

$$w_{d,t} = 1+\ln(f_{d,t})\times \ln(1+\frac{N}{f_t})$$

Again, for query terms that are missing in $V$, we assign $w_{d,t} = 0$. Considering these weights, we define following query predictors:

**Predictor 9 ($var_1$):** Given a query $Q$ ($t_1 \ldots t_n$), the basic variance score is defined as the sum of the deviations:

$$\text{var}_1 = \sum_{t\in Q}\sqrt{\frac{1}{f_t}\sum_{d\in D_t}(w_{d,t}-\overline{w}_t)^2}$$

Where

$$\overline{w}_t = \frac{\sum_{d\in D} w_{d,t}}{|D_t|}$$

where $w_{d,t}$ is the weight of term $t$ in document $d$, and $D_t$ is the set of documents that contain query term $t$. This predictor sums the deviations across query terms, and thus reflects the variability of the query as a whole. An alternative is to use a metric normalized for query length:

**Predictor 10 ($var_2$):** Normalizing the $var_1$ score by the number of valid query terms, gives $var_2$ score for a given query $Q$:

$$\text{var}_2 = \frac{\text{var}_1}{|Q|_{t\in V}}$$

**Predictor 11 ($var_3$):** *It* estimates the performance of a query based on the maximum deviation from the mean that is observed for any one query term:

$$\text{var}_3 = \max_{\forall t\in Q}\left[\sqrt{\frac{1}{f_t}\sum_{d\in D_t}(w_{d,t}-\overline{w}_t)^2}\right]$$

Where $w_{d,t}$ is defined as in **Predictor 9.**

## IV. Dimensions For Estimating Performance of Pre-retrieval Predictors

Pre-retrieval predictors can be evaluated at least along four dimensions, according to the heuristic they exploit: specificity, ambiguity, term relatedness and rank sensitivity.

### A. *Specificity*

The specificity based predictors predict a query to perform better with increased specificity. How the specificity is determined, further divides these predictors into two classes

- Corpus Based - These predictors are based on corpus based statistics such as term/document frequency, inverse term/document frequency of the query terms and their variants. Averaged Inverse Document Frequency (AvIDF) [18] assumes the more discriminative the query terms on average, the better the query will perform. Maximum Inverse Document Frequency (MaxIDF) [5] on the other hand bases its prediction on the most discriminative term of all query terms only. For other variations on number of slight variations on AvIDF refer to [7,12].

- Query Based- As clear from the name, these predictors are based solely on the information contained in query itself, usually length of query. It is based on the assumption that longer terms are more specific. One such predictor is average query length(AVQ)[9].

### B. *Ambiguity*

A different variety of predictors exploit the query term's ambiguity. Low ambiguity indicates an easy query. In addition to exploiting corpus related information, these measures use information obtained from external sources such as Word net to predict level of ambiguity in a query. One of methods of determining ambiguity is based on number of senses of words, which can be determined from WordNet. Higher the sense counts, higher the term's ambiguity. In other schemes, context based information is used to determine level of ambiguity. If a term always appears in the same or similar contexts, the term is considered to be unambiguous. Accordingly if query terms

always appears in the same or similar contexts across all documents, query is considered to be unambiguous.

### C. Term Relatedness

Term relatedness based predictors consider the relationship between query terms. A strong relationship between query terms suggests a well performing query. Predictors of this kind include: Average joint wise Mutual Information (AvPMI) and Maximum Pointwise Mutual Information (MaxPMI) which capture the dependency between query terms, such that a high score is assumed to be correlated with better performance. Semantic similarity between the query terms (their relatedness) can also be derived from WordNet.

### D. Ranking Sensitivity

Finally, ranking sensitivity based predictors predict a query to be difficult if the retrieval algorithm cannot distinguish the documents containing the query terms from each other. Predictors in this category exploit the potential sensitivity of the result ranking by predicting how easy it will be for the retrieval method to rank the documents containing the query terms. If all documents "look the same" to the retrieval method, it is difficult to rank them and the query is deemed difficult. In [22] three predictors of this nature are proposed: Summed Term Weight Variability (SumVAR), Averaged Term Weight Variability (AvVAR) and Maximum Term Weight Variability (MaxVAR). These predictors rely on the distribution of term weights across the collection.

## V. Our Approach

In this section we present our predictor for finding query difficulty level based on co-occurrence of query terms. In our approach we expect that higher co-occurrence value means more information is conveyed, which means easier query or less query difficulty level.

### A. Motivation for using co-occurrence based information for estimating query difficulty

We use co-occurrence based information for estimating query difficulty for following reasons

#### 1) Considering importance of co-occurrence as a measure for finding relatedness between query terms

Term co-occurrence has been widely used as a technique for query expansion. The idea behind this is based on the assumption that a query expresses user's need and is therefore expected to be good at discriminating relevant documents from non relevant document. Further it is expected that if an index term is good at discriminating relevant from non relevant document then any closely associated index term is likely to be good at this. Term co-occurrence data has been extensively used in information retrieval system for identification of indexing terms that are similar to those that have been specified in user query : these similar terms can be used to augment original query. Main source of extracting co-occurring terms is the corpus from where documents are coming . These terms can be either selected globally (from entire corpus) or locally (top n retrieved relevant documents), each having its own pros and cons. On the basis of our study and work we came to following observations.

Query expansion based on co-occurrence based information has shown varying degree of success. This means while increasing retrieval efficiency for some queries, it may drastically fail and may deteriorate retrieval efficiency for other set of queries. One important reason for highly varying performance can be that if the query terms themselves are related, the terms co-occurring with them are related to each other and to whole query in totality, otherwise terms are neither related to each other nor to the entire query. In second case, we will generally see deterioration in retrieval performance after performing query expansion. Such queries are considered to be difficult queries. This observation motivated us to suggest a co-occurrence based predictor for estimating query clarity (difficulty).

#### 2) Considering query terms in totality

We also observe that most of the existing query predictors are based on evaluating query terms separately and then combining their result in various ways. Whereas our predictor considers all query terms in totality, as will be clear in next sub-section.

#### 3) Need of predicting along different dimensions

As discussed in previous subsections a number of pre retrieval predictors have been proposed. These predictors may focus along one or more dimensions of query difficulty. Sometimes it may be useful to develop predictors focusing along a particular dimension, but in order to consider overall level of query difficulty it may be fruitful to develop predictors that consider all dimensions in totality. Some researchers tried to find average of different predictors, but found that the result was worse than the individual predictors. On the basis of our observation we think that most of the time combining predictors deteriorates the performance because different predictors may focus along different dimensions. Two predictors focusing along different dimensions may not improve the result. We think that our predictors are different from other predictors as they focus along all the dimensions to a certain extent viz : specificity, ambiguity, term relatedness and ranking sensitivity. Though mainly our predictor is based on term relatedness but it implicitly considers all dimensions. This is because if query terms are more related query is more specific and less ambiguous. Also rank sensitivity of such query is also high.

### B. *Proposed QueryPredictors*

Our motivation was to use co-occurrence based information for predicting query difficulty. Though co-occurrence information can be derived using various measures such as cosine similarity, jaccard coefficient, and here we suggest a simple co-occurrence score. One idea behind presenting this score is that it considers all the query terms in totality, in comparison to other measures, which find co-occurrence with individual query terms and then combine them to find final value. Secondly this measure is computationally easy to calculate. Another aspect of this measure is that in addition to estimating query clarity level, it provides an initial insight into nature of query. If co-occurrence score is high that means query focuses on one topic, otherwise query is expected to focus on multiple topics. This may suggest more in-depth analysis of query terms such as finding pair wise co-occurrence. Such type of information can be very useful in selecting appropriate query expansion method for specific queries

**Predictor 12(co-degree-score):** Given a query $Q$ ($t_1$. . . $t_n$), co-degree-score (CDS) score is defined as :

$CDS = D_{all}$ where $D_{all}$ is number of documents containing all query terms. In our case CDS measures co-degree of co-occurrence of all query terms (eliminating stop words). However, instead of co-degree we can use any other term which measures degree of co-occurrence of query terms. This predictor may be biased along short queries as obviously smaller the number of terms, query terms are expected to be co-occurring in larger number of documents. Thus another predictor is suggested that uses normalized measure.

**Predictor 13(normalized-co-degree_score):** Given a query $Q$ ($t_1$. . . $t_n$), the normalized co-degree-score (NCDS) is defined as :

$$NCDS = \frac{D\,all}{\sum_{i \in Q} D_i - D_{all}}$$

where $D_{all}$ is again number of documents containing all query terms, $D_i$ is number of documents containing $i^{th}$ query term. In this case CDS is normalized by dividing CDS by number documents in union of documents containing all the query terms.

We expect that co-occurrence based information has a great potential in estimating query difficulty level. We have suggested some simple measures; however other co-occurrence measures might lead to development of new predictors.

## VI. EXPERIMENTS AND RESULTS

In order to evaluate efficiency of a query predictor, some quantitative measure is required. One of the established methodologies used by different researchers is to find co-relation between query predictor and average precision of the query. The argument being that an easy query is one which gives a high average precision as determined by retrieval system, otherwise query is difficult. We have performed experiments on all the 13 predictors. In our experiment, for each query in a set of test queries, we calculated a predicted difficulty based on one particular prediction approach. For the same set of queries, an actual system performance metric (Average Precision) was calculated. Jaccard similarity measure was used for retrieving relevant documents Finally, a correlation coefficient was calculated between these two sets of numbers, to quantify the strength of the relationship between the predicted and actual system performance. The higher the correlation coefficient, the "better" the predictor is considered to be. In the literature, three correlation coefficients are commonly used: Pearson (linear) correlation, Spearman (rank) correlation, and Kendall's tau (also based on ranks).

In this paper we experimented with 11 different predictors which were already there in the literature and 2 predictors proposed by us. The experiment was performed on document collections from the TREC collection. The testing was performed using 50 TREC topics over the TREC. We experimented with short queries based on the topic title. With a sample size of 50 queries, relatively weak correlations with a coefficient of around 0.2 are usually enough to establish statistical significance at the 0.05 level; where larger query sets are evaluated, even lower correlation values are significant. In this paper we report significance at the 0.05 level. Table1 shows the summary of the predictors used with respect to the dataset used .The results obtained can be visualized through the graph(Figure 2) between query difficulty predictor and average precision (which measures actual query difficulty level). Overall evaluation of efficiency of query predictor is shown in Table II. This table measures the performance of query predictor based on correlation coefficients between query predictor and actual performance of query (average precision in our case) For the purpose of evaluating the predictors, we report the linear correlation coefficient, Spearman and Kendall's tau (Table 2).

TABLE I. STATISTICAL SUMMARY INFORMATION OF THE PREDICTORS BASED ON TREC DATASET.

| | Mean | Variance | SD | Min | Max | Normal Distr. |
|---|---|---|---|---|---|---|
| **Average precision of unexpanded** | 0.119 | 0.01528 | 0.1236 | 0 | 0.42 | 0.0178 |
| **P1-Length of Query** | 3.98 | 5.122 | 2.2632 | 1 | 11 | 0.0104 |
| **P2-Standard** | 0.772 | 0.1137 | 0.3373 | 0.0627 | 1.557 | 0.8548 |

| | | | | | | |
|---|---|---|---|---|---|---|
| **Deviation** | | | | | | |
| **P3-inf amt** | 9.263 | 161.4624 | 12.7068 | 0 | 78.481 | <0.0001 |
| **P4-query clarity** | 4.148 | 8.6903 | 2.9479 | -1.672 | 11.694 | 0.3222 |
| **P5-query Scope** | 1.216 | 0.6094 | 0.7807 | -0.0989 | 3.643 | 0.0007 |
| **P6-SQWC** | 128.108 | 3484.3977 | 59.0288 | 41.718 | 282.166 | 0.1027 |
| **P7-NSQWC** | 34.679 | 42.5375 | 6.5221 | 17.916 | 46.857 | 0.2961 |
| **P8-MSQWC** | 42.182 | 7.8587 | 2.8033 | 34.066 | 48.712 | 0.0298 |
| **P9-var1** | 7.043 | 14.0936 | 3.7541 | 2.155 | 23.531 | <0.0001 |
| **P10-var2** | 2.009 | 0.756 | 0.8695 | 0.848 | 4.773 | <0.0001 |
| **P11-var3** | 2.866 | 1.4652 | 1.2105 | 1.488 | 8.179 | <0.0001 |
| **P12-CDS** | 74.26 | 55664.727 | 235.9337 | 0 | 1406 | <0.0001 |
| **P13-NCDS** | 0.00657 | 0.000356 3 | 0.01888 | 0 | 0.0965 | <0.0001 |

TABLE II. RESULTS OF THE PREDICTOR EVALUATIONS GIVEN BY THE LINEAR, SPEARMAN AND KENDALL'S TAU CORRELATION COEFFICIENT

| Query | Predictors | Pearson | Kendall | Spearman |
|---|---|---|---|---|
| 51-100 | **P1-Length of Query** | -0.382 | -0.382 | -0.287 |
| | **P2-Standard Deviation** | -0.209 | -0.196 | -0.12 |
| | **P3-inf amt** | -0.069 | -0.078 | -0.0581 |
| | **P4-query clarity** | 0.179 | 0.287 | 0.202 |
| | **P5-query Scope** | 0.093 | 0.164 | 0.121 |
| | **P6-SQWC** | -0.322 | -0.309 | -0.217 |
| | **P7-NSQWC** | 0.339 | 0.355 | 0.259 |
| | **P8-MSQWC** | 0.115 | 0.096 | 0.0588 |
| | **P9-var1** | -0.227 | -0.182 | -0.126 |
| | **P10-var2** | 0.294 | 0.318 | 0.24 |
| | **P11-var3** | 0.034 | 0.139 | 0.108 |
| | **P12-CDS** | 0.346 | 0.341 | 0.264 |
| | **P13-NCDS** | 0124 | 0.276 | 0.36 |

We observe that overall results of linear and rank based co-relations are very similar. From the results, we could derive the following observations:

- In our case we found that query length has a negative linear and rank correlation with AP. In fact there might not be a direct relationship between query length and query difficulty. Moreover as we have worked on short queries, the length of query ranges between 2 -11 words and average query length was of 3.4 word length, we cannot comment much on effect of query length on query difficulty.
- P2 , P3 has negative linear and raking correlation with AP in all cases. Both these predictors are based on idf of the terms present in the query. In our dataset the idf range is very small. The min idf is 1.39 and max idf is 4.39. Thus we may not be able to analyze the effect of idf on our experimental set up.
- P4 is strongly correlated with AP. Measuring the correlation, we obtained $r$ =0.356and a p-value 0.244 which shows positive correlation. The ability of query_clarity to predict query performance is based on the following assumption: a query whose highly ranked documents contain many relevant documents is likely to receive a high clarity score because these highly ranked documents tend to be about a single topic and therefore have unusual word usage.

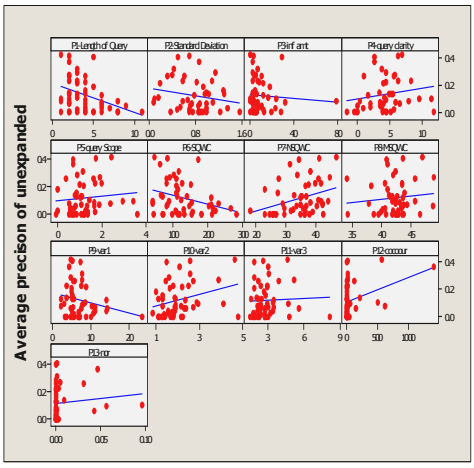


Figure 2: Scatter plot of Average Precision and all 13 Predictors

- P5 (query_scope) also have positive correlation .This predictor considers documents having at least one of the query term out of the whole dataset.
- P6(SQWC) is negatively correlated with p-value 0.022. This predictor is based on the criteria that the similarity between a query and the collection. SQWC can provide useful information for the prediction of how well a query will perform. In our opinion, however if a query is very much similar to entire document collection, it is difficult to identify relevant documents for such query. Therefore this measure can actually be negatively correlated with average precision.
- However in our collection P7 and P8(normalized and maximum SQWC measures) are positively correlated.
- Again P9, basic variance score gave negative result in our experiments but var2 and var3, which are normalized and maximum variance scores gave positive results.

- Experimental evidence shows that our proposed predictors P12, P13 perform well. These predictors are positively correlated with actual prediction(average precision in our case).

The predictor performances depend on the particular test collection as well as the particular retrieval approach. Among the assessed predictors, there is not a single predictor that outperforms all others across all settings evaluated. Analysis of result shows that P7(NSCQ) and our predictors P12 1nd P13 outperform other predictors in most of the cases. These predictors are positively correlated with average precision.NSCQ gives best Kendall coefficient(0.355), where our predictor P12(CDS) give second best performance (0.341).Whereas our predictor P12(CDS) gives best Pearson coefficient (0.346) and our second predictor P13(NCDS) gives best Spearmen coefficient value(0.36).

## VII. CONCLUSION

In this paper we have introduced two query difficulty pre-retrieval predictors based on co-occurrence information among query terms. There were two strong motivations for suggesting these predictors. Firstly co-occurrence among query terms is a strong indicator of term relatedness. Higher the co-occurrence score more are chances that query terms are related, indicating that query is less difficult (more clear). Secondly we think that our predictors are different from other predictors as they focus along all the dimensions of query difficulty to a certain extent viz : specificity, ambiguity, term relatedness and ranking sensitivity. More over these measures give an insight into analyzing nature of query. We performed extensive experiments on TREC data set. Results have been compared with 11 existing query predictors. The results suggest that among different query predictors proposed in literature no predictor can be considered as best. However our results are motivating and set a background for exploiting various co-occurrence based measure for predicting query difficulty. Our work addresses the basic issue of reliability in query processing operations.

## REFERENCES


[1] Buckley, C and Harman, D. Reliable Information Access FinalWorkshopReport.,http://nrrc.mitre.org/NRRC/Docs_Data/RIA_2003/riafinal.pdf ,2004.

[2] Carmel D, Yom-Tov E, Darlow A et al. What makes a query difficult? In: Proceedings of the 29th annual international ACM SIGIR conference on research and development in information retrieval, pp 390–397,2006.

[3] David Carmel, Eitan Farchi, Yael Petruschka, and Aya Soffer. Automatic query refinement using lexical affinities with maximal information gain. In Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval, pages 283–290. ACM Press, 2002.

[4] F. Diaz and R. Jones. Using temporal profiles of queries for precision prediction. In SIGIR '04: Proceedings of the 27th annual international conference on Research and development in information retrieval, pages 18–24. ACM Press, 2004.

[5] F. Scholer, H. Williams, and A. Turpin. Query association surrogates for web search. Journal of the American Society for Information Science and Technology, 55(7):637–650, 2004.

[6] 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.

[7] He B, Ounis I . Inferring query performance using pre-retrieval predictors. In: proceedings of the SPIRE 2004, pp 43–54,2004.

[8] J. He, M. Larson, and M. de Rijke. Using coherence-based measures to predict query difficulty. In ECIR'08, pages 689–694, 2008.

[9] J. Mothe and L. Tanguy. Linguistic features to predict query difficulty - a case study on previous trec campaigns. In SIGIR'05 Query Prediction Workshop, 2005.

[10] Jensen EC, Beitzel SM, Chowdhury A et al. Predicting Query Difficulty on the Web by Learning Visual Clues. In: Proceedings of the 2005 ACM conference on research and development in information retrieval, pp 615–616, 2005.

[11] Karlgren, J. Stylistic Experiments in Information Retrieval, in Natural Language Information Retrieval, Kluwer,1999.

[12] Kwok KL, Grunfeld L, Dinstl et al . TREC 2005 Robust Track Experiments Using PIRCS. In: The Online proceedings of 2005 text REtrieval conference,2005.

[13] Kwok KL, Grunfeld L, Sun HL et al. TREC 2004 Robust Track Experiments Using PIRCS. In: The Online Proceedings of 2004 text REtrieval conference, 2004.

[14] Kwok, K.L. An attempt to identify weakest and strongest queries. In: Predicting Query Difficulty, SIGIR 2005 Workshop, 2005.

[15] Mandl, T. Womser-Hacker, C. Linguistic and Statistical Analysis of the CLEF Topics, CLEF Workshop ,2002.

[16] Plachouras V, He B, Ounis I. University of Glasgow at TREC2004: experiments in Web, robust, and terabyte tracks with terrier. In: The online proceedings of 2004 text REtrieval conference,2004.

[17] S. Banerjee and T. Pedersen. Extended gloss overlaps as a measure of semantic relatedness. In IJCAI-03, pages 805–810, 2003.

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

[19] Tomlinson S . Robust, Web and terabyte retrieval with hummingbird searchServer at TREC 2004. In: The online proceedings of 2004 text REtrieval conference,2004.

[20] Vinay V, Cox IJ, Mill-Frayling N et al.On ranking the effectiveness of searcher. In: Proceedings of the 29th annual international ACM SIGIR conference on research and development in information retrieval, pp 398–404,2006.

[21] WordNet - An Electronic Lexical Database. The MIT Press, 1998.

[22] Y. Zhao, F. Scholer, and Y. Tsegay. Effective pre-retrieval query performance prediction using similarity and variability evidence. In ECIR'08, pages 52–64, 2008.

[23] Yom-Tov E, Fine S, Carmel D et al. Learning to estimate query difficulty with applications to missing content detection and distributed information retrieval. In: The proceedings of 28th annual international ACM SIGIR conference on research and development in information retrieval, pp 512–219, 2005.

[24] Zhou, Y., Croft, W.B. Query performance prediction in web search environments. In: Proceedings of the ACM SIGIR International Conference on Research and Development in Information Retrieval, Amsterdam, The Netherlands, pp. 543–550 ,2007.