Evaluation of Query Performance Prediction Methods by Range*

Joaquín Pérez-Iglesias and Lourdes Araujo

Universidad Nacional de Educación a Distancia
Madrid 28040, Spain
joaquin.perez@lsi.uned.es, lurdes@lsi.uned.es

Abstract. During the last years a great number of Query Performance Prediction methods have been proposed. However, this explosion of prediction method proposals have not been paralleled by an in-depth study of suitable methods to evaluate these estimations. In this paper we analyse the current approaches to evaluate Query Performance Prediction methods, highlighting some limitations they present. We also propose a novel method for evaluating predictors focused on revealing the different performance they have for queries of distinct degree of difficulty. This goal can be achieved by transforming the prediction performance evaluation problem into a classification task, assuming that each topic belongs to a unique type based on their retrieval performance. We compare the different evaluation approaches showing that the proposed evaluation exhibits a more accurate performance, making explicit the differences between predictors for different types of queries.

Keywords: Information Retrieval, Evaluation, Query Performance Prediction.

1 Introduction

Research on Query Performance Prediction, or QPP, has attracted growing attention from the information retrieval (IR) community in the last years. This topic is focused on predicting the retrieval effectiveness of a query, that is, the quality of a search engine's response to a submitted query. The relevance of a subset of documents returned by a search engine is usually measured by means of the user's relevance judgements and the Average Precision (AP). Therefore, a query which obtains a high AP value can be considered as 'Easy', since the retrieval model was able to return a relevant subset of documents. Otherwise if the query obtains a low AP it is considered as 'Hard'. Having the ability to predict the performance of a query can help us to apply some specific techniques in order to improve the relevance of the returned documents. This improvement

* This paper has been funded in part by the Spanish MICINN projects NoHNES (Spanish Ministerio de Educación y Ciencia - TIN2007-68083) and by MAVIR, a research network co-funded by the Regional Government of Madrid under program MA2VICMR (S2009/TIC-1542).

E. Chavez and S. Lonardi (Eds.): SPIRE 2010, LNCS 6393, pp. 225–236, 2010.

can be achieved with classic techniques such as relevance or pseudo-relevance feedback, or with the use of a different index whose content is more related to the submitted query.

In the last years several methods have been proposed to deal with QPP, which fall into one of the following groups, depending on the information used to make predictions:

- **Pre-retrieval** methods, where the estimations are computed using query terms statistics, such as collection frequency (CF), document frequency (DF) or query length. An extensive description about this type of prediction methods can be found in the work developed by He and Ounis [1].
- **Post-retrieval** methods, which are usually more complex and are computed using the document ranked list returned by the search engine combined with other collection statistics. The most representative example within this type of prediction methods is Clarity Score. It was proposed by Cronen-Townsend et al. [2] and it is based on measuring the divergence between the relevance language model[1] and the collection language model, where a high divergence suggests a well-performing query.

In general post-retrieval methods achieve better estimations since they use more information to compute predictions, although this entails a considerable increase of the computational cost.

Prediction methods are evaluated by means of correlation coefficients. The goal is to measure for each topic the correlation degree between the estimated value, obtained with the proposed prediction method, and the Average Precision value. Therefore a prediction method is considered more accurate if it obtains a higher value of correlation between the actual values and the generated prediction values.

The correlation degree between two random variables measures the dependence between both variables. This dependence in general is quantified with a real number in the range $[-1, 1]$, where 1 means a perfect direct correlation, -1 means a perfect inverse correlation and 0 means no correlation at all. Therefore, for values of correlation close to zero no dependence between both variables can be observed, although this fact does not imply a total independence between both random variables. Besides, high values of correlation, positives or negatives, suggest, but do not assure, a possible dependence between both variables.

The evaluation based on correlation coefficients usually produces very similar results for the different prediction methods as it can be observed in the related literature [3]. These values are usually hard to interpret, since the differences among the obtained correlation coefficients are sometimes too low.

The evaluation of prediction methods should be focused on their application to specific contexts, and therefore the evaluation framework should help us to decide if a prediction method is suitable for that specific context. Current evaluation, based on correlation, provides a very coarse measure of the method

[1] The relevance language model is built using a subset of the documents returned by the query.

accuracy, ignoring some important details of the real performance. For instance an evaluation more focused on the predictors application should be able to answer questions like: Is a new method able to outperform others in relation with the detection of 'Hard' queries?

In order to stress these differences in terms of prediction performance we propose a new method for measuring the effectiveness of a query performance predictor which provides information for different levels of retrieval quality. This task can be achieved by assuming a discrete classification of topics, that is, assuming that each topic belongs to a unique type based on their retrieval performance. This assumption avoids the drawbacks which arise with the use of the correlation approach, since it allows us to measure the predictor performance partially, i.e. for each type of topic, and globally as it is done by correlation coefficients.

The rest of this paper is organised as follows. In Section 2 current evaluation approaches are introduced, with a special emphasis on describing their main weaknesses. Section 3 is devoted to describe a new evaluation framework for Query Performance Prediction, which we introduce in order to overcome some of the previously analysed limitations. In Section 4 the proposed evaluation method is tested against current approaches using a standard TREC collection, and a detailed analysis of the obtained results is carried out. Finally, the main conclusions drawn from this work appears at Section 5.

2 Current Evaluation Approaches

In the context of Query Performance Prediction *Pearson*(r) and *Kendall*(τ) are the most commonly applied correlation methods[2] to evaluate the accuracy of the estimations.

Pearson is a parametric method, which assumes a linear relationship between both data series indicating the strength and direction of this relationship, whereas *Kendall* computes the correlation value counting the pairwise swappings needed to transform one ranking into the other. *Kendall* is a non-parametric method, and thus does not make assumptions about the input data, providing less information about the data relationship. Therefore, while *Pearson* is focused on establishing if both data series are produced by two linearly dependent functions, *Kendall* measures how similar are the orderings produced by the prediction method in comparison to the one produced by the 'actual' values. The obtained *Kendall's* τ coefficient can be interpreted as a degree of certainty, which indicates if a topic would obtain a higher AP value than other, although the AP value itself cannot be predicted as *Pearson's* r does.

Due to their different nature both methods frequently produce dissimilar values, and thus a direct comparison between both coefficients is not possible.

Pearson drawbacks have been extensively treated in the literature. It is well-known that *Pearson* produces very different results for data series which show

[2] In some works we can find Spearman (ρ) correlation coefficient, although its use remains rare in this context.

strong dependence, whenever a data outlier appears or a small fraction of the data takes values far from the mean. Another known problem occurs when both random variables show a strong dependence but through a non linear relationship. In this case the correlation coefficient r will be low even if the data are strongly correlated. The main problems of the application of *Pearson* correlation coefficient in the context of QPP were previously pointed out by Hauff et al.[4].

Due to the described problems many works on query performance prediction report their results with the *Kendall* correlation coefficient at the expense of using a less informative evaluation measure.

Although *Kendall* is considered as a more appropriate measure to evaluate estimations, in our opinion this measure does not make explicit the real effectiveness of the predictor, since we only obtain a unique value describing the average accuracy, but ignoring the performance for different types of queries.

Kendall is only focused in the number of disordered elements and the distance of them to the position where they should be placed. Therefore *Kendall* gives the same importance to all elements within the data series. On the context of QPP we can be more interested on analysing the performance on topics of different difficulty ('Hard' or 'Easy'), that is, topics placed at the top or bottom of the topic list sorted by AP. For instance a key factor that should be highlighted in the operation of a predictor is its ability to detect those topics that obtain a low AP value. This feature is of extreme importance for tasks such as pseudo-relevance feedback, since it will help us decide in which cases to expand the original user query, as it has been recently studied by He et. al [5].

For instance, let us consider the data series: $X = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$ and $Y = \{\mathbf{4}, 2, 3, \mathbf{1}, 5, \mathbf{10}, 7, 8, 9, \mathbf{6}\}$, where we consider elements 1 to 5 as the *'best'* elements and the rest as the *'worst'* elements. Since elements 1,4,6 and 10 are not placed at their right position a Kendall correlation of 0.46 is obtained, which suggests a significant but not strong correlation between data. On the other hand if we are only interested on evaluating by type of elements, i.e. *'best'* and *'worst'*, we can conclude that both data series X and Y group both types of elements in the same fashion, as best elements are grouped within the first five positions, and therefore the worst elements are within the last places.

On the other hand, this drawback can not be overcome by measuring the Kendall correlation partially for each type of elements. For instance, if we consider the next data serie $Z = \{6, 7, 8, 9, 10, 1, 2, 3, 4, 5\}$ produced by a predictor, where as before we consider elements 1 to 5 as the *'best'* elements and the rest as the *'worst'* elements. The τ obtained between X and Z for the so-called *'best'* and *'worst'* elements is in both cases equals to 1. However it can be observed that the predictor shows a pretty poor performance since it has predicted the best elements as worst and viceversa.

Some extensions to *Kendall* try to overcome this drawback assigning a relative importance for each element or groups of elements to the final τ value. This family of Kendall variants is known as 'Weighted Kendall', and are usually applied to measure the similarity between responses of different search engines [6,7]. Using this approach it is possible to measure if a method shows a better

performance for a specific type of topics than other, although we must set a suitable weight for each element. Another weakness of this approach is that it is not able to provide partial results for the different types of topics.

3 Evaluating by Range

Previous sections have introduced the idea that using correlation coefficients as a quality measure only provides a global view of the predictor performance, ignoring its specific behaviour on different types of topics, i.e. topics of different difficulty. However, it is expected that different prediction methods show different performance on 'Easy' or 'Hard' topics. Detecting this disparity in terms of prediction effectiveness can be a key factor in order to apply these methods to improve the retrieval quality for a specific type of queries.

In this section we propose a new QPP evaluation framework with a two fold goal: *a*) evaluate the ability of the method to detect 'Easy' or 'Hard' queries; and *b*) to make explicit the accuracy of the method for different types of topics.

For the development of this evaluation framework, we should be able to partition the topic set of study into n blocks of interest, where each topic is uniquely assigned to one of these partitions by its corresponding AP value, which establishes the retrieval quality of the partition. Thus, the best topics in terms of AP would be assigned to the first partition and likewise the *n-th* partition groups the worst topics. The same process is applied using the values provided by the prediction method, instead of AP.

After partitioning the full set of topics, each topic is labelled according to the prediction and the average precision obtained. Therefore, it is possible to test for each topic if the AP value and the estimation belong to the same partition. Thus, the evaluation of the quality of this labelling resembles a problem of classification evaluation.

Grouping Data

An interesting problem which arises with the proposed evaluation is how to group topics by their retrieval performance. The application of a suitable method to group topics is a key task for the evaluation of QPP methods, since different systems gives rises to different retrieval performance.

For instance, we can imagine a hypothetical search system where almost all queries obtain as response a set of documents which satisfy the user. In this case the application of a prediction method is not necessary, since it is known that all queries are correctly answered. A similar case occurs when a search system is unable to respond correctly to a great majority of queries. Therefore, a prediction method is only useful when there is a significant divergence in the quality of the search engine responses. In general, this divergence occurs in actual search systems.

Focusing on the TREC environment, a typical run usually shows an exponential probability distribution[3], as Figure 1 illustrates, where a great number of topics obtain a low AP value.

In order to adapt the different partitions of topics to each search system, we propose to group them following the probability distribution of the AP values, which represents the overall search system performance. For this task the k-means[8] clustering algorithm is a suitable approach, since our goal is to create groups such that the distance between elements in the same group is minimum and the distance among the means of each group is maximum. More formally, given a set of topic AP values $(t_1, t_2, \cdots, t_n)$, the k-means clustering aims to partition the n observations into k sets $(k < n)$ $C = \{C_1, C_2, \ldots, C_k\}$ so as to minimize the within-cluster distance to the mean:

$$\arg\max_{C} \sum_{i=1}^{k} \sum_{t \in C_i} (t - \bar{C}_i)^2$$

where $\bar{C}_i$ is the mean in C_i.

The k-means algorithm is not a deterministic method of clustering because it depends on the initial selection of cluster means. In order to circumvent this problem we propose to set the initial means in such a way that they do not introduce any bias in the construction of the groups. Therefore, initial means should be uniformly distributed along the whole set of topics in such a way that, the number of topics between the proposed means is roughly the same. This method can be implemented computing the percentiles for each desired group based on the next equation: $\bar{C}_i = percentile\left(100 \cdot \frac{i-1}{n+1}\right)$ where $\bar{C}_i$ is the initial mean of the *i-th* cluster, and n is the number of clusters. For instance if n=3, the initial means will be fixed at the *25th*, *50th* and *75th* percentile of the data.

Using k-means in combination with the initialisation method proposed above ensures that the groups created will depend on the data distribution. A proof of this fact is that those groups with a larger number of topics appear where the density function has a maximum, as observed in Figure 1.

The k-means algorithm requires a parameter indicating the desired number of groups in which the input data will be divided. In our opinion a reasonable approach is to set this value manually, since this number only depends on the granularity level desired for the study[4]. For example, a typical setup can include three groups: 'Easy', 'Average' and 'Hard' topics. Obviously, in order to compare different prediction methods the number of partitions must be the same for all the evaluated methods.

The described methodology for grouping topics is also applied to the prediction values. Applying to both data series the same partition method has a strong effect on the results of typical evaluation measures as *precision* or *recall*. In general, those prediction methods whose probability distribution is not similar

[3] Although best runs approach a Gaussian distribution as the MAP value increases.

[4] Although there are several methods which set automatically the number of clusters for the k-means algorithm.

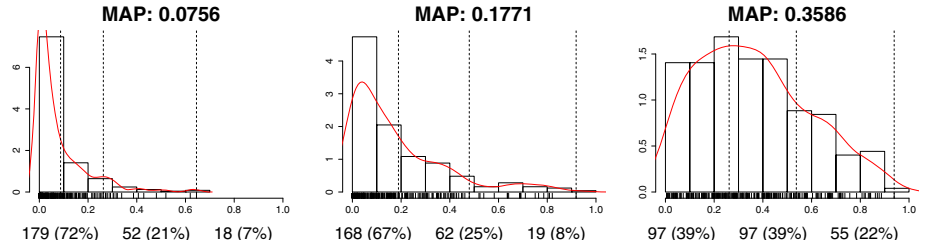

Fig. 1. AP histograms, density function and the obtained partitions applying k-means with k=3 for the worst, average and best run submitted to the Robust 2004 track

to the AP distribution produce poorer results, because this dissimilarity leads to the creation of partitions with different sizes, and this affects the evaluation measures.

In the next section we apply the introduced evaluation framework to a subset of different prediction methods in a standard TREC collection, in order to test the ability of our proposal to circumvent some of the drawbacks of the current approaches for the evaluation of QPP methods.

4 Experiments and Results

For the experimental evaluation of our proposal the set of documents from TREC Disk4 & 5, along with the full set of topics from the Robust 2004 track[9], are employed. We have selected this set of topics since a majority of prediction methods obtain their best results when they are tested with them, as it appears in related literature [3].

All prediction methods are executed against a base run which was obtained using the query likelihood language modeling [10], with a Dirichlet prior smoothing parameter[11] equal to 1500[5]. This run achieves a MAP value of 0.24, which is of the order of a typical TREC run for this collection.

A significant set of prediction methods considered as state of the art have been implemented. For the pre-retrieval case, we have tested some of the methods proposed by He et. al [1], including those based on the query terms IDF or ICTF, such as the maximum IDF, the average ICTF, Simplify Clarity Score (SCS) and the QueryScope method based on the number of documents returned by each query term. On the other hand, the post-retrieval methods tested include Clarity Score[2], for which the number of feedback documents have been fixed to 500, and the ScoreDesv method, which is based on measuring the standard deviation of the top k scores, fixing the top k to 120, as recommended in [12].

The set of topics is grouped into three blocks: 'Easy', for topics with the highest AP; 'Hard', for those with lowest AP; and 'Average', for the rest. The details of the different groups obtained after the application of the k-means

[5] For this task The Lemur Toolkit software was employed.

Table 1. Statistics for 'Easy', 'Average' and 'Hard' topic groups, including number of topics per group, maximum, mean and minimum AP per group and the AP standard deviation per group

	Num	Max	Mean	Min	Sd
Hard	121	.02	.001	.0005	.006
Average	97	.21	.10	.02	.05
Easy	31	.91	.41	.21	.17

algorithm appear in Table 1. As expected, for a typical TREC run the largest group corresponds to the set of 'Hard' topics, while the group with the smallest number of topics corresponds to the 'Easy' partition.

In order to describe in detail the performance obtained by the tested prediction methods, several measures from the classification topic can be applied. The wide range of available measures allow us to define the experimental setup as a function of the desired type of study. For the current setup we have decided to apply the measures described below with the main goal of highlighting the differences in terms of prediction not shown by the correlation coefficients.

The simplest approach to compare the accuracy of the different evaluated methods is to compute the number of hits per partition and the total number of hits. We will employ the classic F-measure, since it is able to combine precision and recall in a single number and can be applied globally and for each defined partition. Formally, the F-measure is defined as $2 \cdot \frac{precision \cdot recall}{precision + recall}$.

In previous measures misclassified elements are equivalent. However, in QPP all errors should not penalise the measure in the same manner. For instance, in a set-up where we have decided to label topics as 'Easy', 'Average' and 'Hard', predicting a topic as 'Hard' when it is actually 'Easy' should imply a greater penalty than if the element had been misclassified as 'Average', a partition closer to 'Easy'. For this purpose we introduce a new measure: the Distance Based Error Measure (DBEM), which is able to indicate the global performance along all the partitions but focused only on the misclassified topics. We define the distance between two partitions C_i and C_j in a set of topics that have been grouped in k partitions as

$$distance(C_i, C_j) = \|i - j\|$$

where $0 < i,j \leq k$. Then

$$DBEM = \frac{\sum_i^n distance(P_{t_i}, C_{t_i})}{\sum_i^n max\left[\forall_{j \in n} distance(P_{t_i}, P_{t_j})\right]}$$

where P_{t_i} is the predicted partition for the topic t_i, C_{t_i} is the AP partition for the topic t_i, and n is the total number of topics. This is, the distance between all topics normalised by the maximum possible distance, where lower distances imply better performances.

Results

Table 2 shows the number of correctly classified topics for the whole set of topics and for each partition. These results indicate that the subset of methods based on IDF is strongly biased toward grouping topics as 'Hard', which leads them to achieve the best result for the 'Hard' partition and very poor results for the 'Average' and 'Easy' partitions. Furthermore, this subset of prediction methods obtain very similar results, as it is shown by the total number of hits achieved by them. However, this similarity in their results is not captured by the *Pearson* and *Kendall* correlation. Moreover, some important differences appear in the correlation coefficients obtained by these methods, suggesting a different performance for these predictors, which has proved false using the simple accuracy. For instance, according to *Pearson* IDFMin (r=0.24) should outperforms clearly IDFAvg (r=0.17), while according to *Kendall* we should conclude exactly the opposite, since IDFMin obtains a τ of 0.16 to be compared with the 0.32 obtained by IDFAvg.

One key factor when comparing prediction methods by their accuracy is to observe the performance not only globally but for each partition too. For example, the obtained global accuracy for the QueryScope method will guide us to the wrong conclusion of a worse performance of this compared to the IDF based methods. But if we check the partial results of QueryScope we can observe a much better performance for this method in the detection of 'Easy' and 'Average' topics in comparison with the IDF based methods. Therefore, we can conclude that the strong performance of the IDF based methods on 'Hard' topics is a consequence of considering more than 95% of topics as 'Hard', which does not correspond to the real performance of the tested run.

The global accuracy is not able to detect the previous situation either, because it is computed as a sum of partial accuracies. However, this weakness is overcome by the F-measure for the whole set of topics, as it can be observed in Table 3 where the IDF based methods obtains the worst results among all predictors.

Table 2. Results for the proposed predictors. The first three columns show the number of hits per type of topic, including in brackets the whole number of topics classified as the type of the column title. Besides, the last three columns show the total accuracy $\frac{hits}{total}$ and the Pearson and Kendall correlation coefficient obtained.

	Hard	Average	Easy	Total	Accuracy	Pearson	Kendall
AVICTF	73(122)	47(110)	9(17)	129	0.52	0.45	0.26
IDFAvg	120(245)	1(3)	1(1)	122	0.49	0.17	0.32
IDFDesv	119(245)	1(3)	1(1)	121	0.48	0.12	0.25
IDFMax	118(243)	1(5)	1(1)	120	0.48	0.15	0.32
IDFMin	121(245)	1(3)	1(1)	123	0.49	0.24	0.16
QScope	91(171)	22(67)	5(11)	118	0.47	0.37	0.18
SCS	60(87)	50(109)	18(53)	128	0.51	-0.45	-0.26
CS	78(110)	54(103)	13(36)	145	0.58	0.51	0.41
ScoreDesv	84(128)	47(91)	17(30)	148	0.59	0.55	0.40

Table 3. Results for the proposed predictors. The first three columns show the F-measure for each type of topic. Besides, the last four columns show the F-measure for the whole set of topics, the Distance Based Error Measure and the Pearson and Kendall correlation coefficient obtained.

	Hard	Average	Easy	Total	DBEM	Pearson	Kendall
AVICTF	0.60	0.45	0.37	0.51	0.32	0.45	0.26
IDFAvg	0.65	0.02	0.06	0.33	0.39	0.17	0.32
IDFDesv	0.65	0.02	0.06	0.33	0.39	0.12	0.25
IDFMAX	0.63	0.02	0.06	0.33	0.39	0.15	0.32
IDFMin	0.66	0.02	0.06	0.36	0.38	0.24	0.16
QScope	0.62	0.26	0.23	0.43	0.35	0.37	0.18
SCS	0.58	0.48	0.43	0.52	0.34	-0.45	-0.26
CS	0.67	0.54	0.39	0.59	0.29	0.51	0.41
ScoreDesv	0.67	0.50	0.56	0.59	0.27	0.55	0.40

Table 4. Confusion Matrix for Clarity Score (left) and ScoreDesv (right), the number of correctly classified topics appears in boldface

	Hard	Average	Easy	Total
Hard	**78**	36	7	121
Average	27	**54**	16	97
Easy	5	13	**13**	31
Total	110	103	36	**145**

	Hard	Average	Easy	Total
Hard	**84**	35	2	121
Average	39	**47**	11	97
Easy	5	9	**17**	31
Total	128	91	30	**148**

A final conclusion that can be extracted from the results in Table 2 is that, as it was expected, the most accurate methods in terms of grouping predictions are CS and ScoreDesv, which outperform clearly the rest of prediction methods.

Table 3, shows the F-measure obtained, which can help us to extract some other conclusions. These results show that in general prediction methods show a better performance detecting 'Hard' topics than 'Average' or 'Easy' topics, something not revealed by means of the correlation coefficients. Besides, while both correlation coefficients suggest an equivalent performance for SCS and AVICTF prediction methods, using the F-measure we observe how SCS outperforms in more than a 16% the AVICTF in relation with the detection of 'Easy' topics.

In relation with the most accurate methods, CS and ScoreDesv, although the correlation methods and the global F-measure suggest a similar performance, a more detailed comparison leads us to observe some interesting differences between them. For instance, with the partial F-measure we can observe that although CS

is slightly better for 'Hard' and 'Average' topics, ScoreDesv improves CS around a 43% detecting 'Easy' topics, which makes ScoreDesv a more reliable option in those contexts where we would expect an accurate detection of 'Easy' topics.

The differences between these last prediction methods are shown as well by the proposed DBEM measure. Although this measure shows a strong correlation with the rest of the global performance measures, as it appears in Table 5, it reveals some important details which are not shown with the rest of measures. Focusing on CS and ScoreDesv, DBEM suggests a minor fail ratio for the last method. This fact can be confirmed observing the confusion matrix of both methods, which appears in Table 4. In this table we observe that while the number of misclassified topics is similar for both methods (104 for CS and 101 for ScoreDesv), CS is labelling 12 topics as 'Easy' when they are actually 'Hard' or viceversa, while these errors only occurs 7 times with ScoreDesv. This error rate implies that the proportion of strong errors by CS is around an 11% against the 6% obtained by the ScoreDesv.

Finally it should be highlighted that the proposed measures applied show a strong correlation with the classic evaluation approach, as it appears in Table 5. Thus the proposed evaluation method provides an information equivalent to correlation approach, but showing a higher level of detail for the topics subset of interest.

Table 5. Pearson correlation coefficient between pairs of proposed measures

	Accuracy	**F-Measure**	**DBEM**
Pearson	0.78	0.77	-0.95
Kendall	0.77	0.66	-0.57

5 Conclusions

In this paper a novel method for the evaluation of Query performance Prediction Methods has been introduced. The goal of this proposal is to avoid some of the drawbacks which appear with the use of correlation coefficients when they are applied to evaluate Query Performance Prediction methods.

Our proposal overcomes previous drawbacks avoiding the use of correlation coefficients, and transforming the performance prediction evaluation into a classification task by assuming that each topic belongs to a unique type based on their retrieval performance. For this task we have proposed an automatic method to group topics based on their retrieval quality, according to the overall retrieval quality of the search system in study.

While the application of correlation coefficients to this topic can hide the specific performance of prediction methods for different types of topics, our proposal makes explicit these differences guiding us to the selection of the more suitable method depending on the application context. Furthermore, each topic is automatically labelled by their retrieval effectiveness according to the prediction

method. Based on this label, a system would be able to decide which is the most suitable technique to improve the quality of the response for this topic, opposite to current approach where this decision is taken based on a numeric value assigned by the prediction method.

The novel evaluation framework has been tested against a set of different prediction methods, providing with a more detailed information about the tested predictors performance. Besides, the proposed measures have shown a strong correlation degree with the current evaluation, which suggests a similar behaviour of our proposal with correlation approach but being at the same time able of revealing some performance differences that are not detected with the current approaches.

References

1. He, B., Ounis, I.: Inferring Query Performance Using Pre-retrieval Predictors. In: Apostolico, A., Melucci, M. (eds.) SPIRE 2004. LNCS, vol. 3246, pp. 43–54. Springer, Heidelberg (2004)
2. Cronen-Townsend, S., Zhou, Y., Croft, W.B.: Predicting query performance. In: Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval - SIGIR 2002. ACM Press, New York (2002)
3. Hauff, C.: Predicting the Effectiveness of Queries and Retrieval Systems. PhD thesis, Univ. of Twente, Enschede (January 2010)
4. Hauff, C., Azzopardi, L., Hiemstra, D.: The combination and evaluation of query performance prediction methods. In: ECIR, pp. 301–312 (2009)
5. He, B., Ounis, I.: Studying query expansion effectiveness. In: ECIR, pp. 611–619 (2009)
6. Melucci, M.: Weighted rank correlation in information retrieval evaluation. In: Kuriyama, K. (ed.) AIRS 2009. LNCS, vol. 5839, pp. 75–86. Springer, Heidelberg (2009)
7. Yilmaz, E., Aslam, J.A., Robertson, S.: A new rank correlation coefficient for information retrieval. In: SIGIR 2008: Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval, pp. 587–594. ACM, New York (2008)
8. Witten, I.H., Frank, E.: Data Mining: Practical Machine Learning Tools and Techniques, 2nd edn. Morgan Kaufmann Series in Data Management Sys. Morgan Kaufmann, San Francisco (June 2005)
9. Voorhees, E.M.: Overview of the TREC 2004 Robust Retrieval Track. In: Proceedings of the Thirteenth Text REtrieval Conference, TREC (2004)
10. Ponte, J.M., Croft, W.B.: A language modeling approach to information retrieval. In: SIGIR 1998: Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval, pp. 275–281. ACM, New York (1998)
11. Zhai, C., Lafferty, J.: A study of smoothing methods for language models applied to ad hoc information retrieval. In: SIGIR 2001: Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval, pp. 334–342. ACM, New York (2001)
12. Pérez-Iglesias, J., Araujo, L.: Ranking list dispersion as a query performance predictor. In: Azzopardi, L., Kazai, G., Robertson, S., Rüger, S., Shokouhi, M., Song, D., Yilmaz, E. (eds.) ICTIR 2009. LNCS, vol. 5766, pp. 371–374. Springer, Heidelberg (2009)