Wikipedia-Based Query Performance Prediction

Gilad Katz[1] katzgila@bgu.ac.il
Anna Shtok[2] annabel@tx.technion.ac.il
Oren Kurland[2] kurland@ie.technion.ac.il
Bracha Shapira[1] bshapira@bgu.ac.il
Lior Rokach[1] liorrk@bgu.ac.il
1. Department of Information Systems Engineering, Ben-Gurion University, Beer-Sheva, Israel
2. Faculty of Industrial Engineering and Management, Technion, Haifa 32000, Israel

ABSTRACT

The query-performance prediction task is to estimate retrieval effectiveness with no relevance judgments. Pre-retrieval prediction methods operate prior to retrieval time. Hence, these predictors are often based on analyzing the query and the corpus upon which retrieval is performed. We propose a *corpus-independent* approach to pre-retrieval prediction which relies on information extracted from Wikipedia. Specifically, we present Wikipedia-based features that can attest to the effectiveness of retrieval performed in response to a query *regardless* of the corpus upon which search is performed. Empirical evaluation demonstrates the merits of our approach. As a case in point, integrating the Wikipedia-based features with state-of-the-art pre-retrieval predictors that analyze the corpus yields prediction quality that is consistently better than that of using the latter alone.

Categories and Subject Descriptors: H.3.3 [Information Search and Retrieval]: Retrieval models

Keywords: query-performance prediction, Wikipedia

1. INTRODUCTION

There is a large body of work on query-performance prediction [5]. The goal is to estimate the effectiveness of ad hoc (query-based) retrieval with no relevance judgments.

Pre-retrieval prediction methods operate prior to retrieval time [12, 15, 16, 11, 22, 10]. Most of these methods analyze the query using information induced from the corpus upon which search is performed [8, 12, 11, 22]. Post-retrieval prediction methods analyze also the result list of the documents most highly ranked [5]. Hence, their prediction quality transcends that of pre-retrieval predictors. However, pre-retrieval predictors are much more efficient as they do not rely on performing the search. Accordingly, here we focus on pre-retrieval prediction.

We present a novel *corpus-independent* pre-retrieval query-performance prediction approach. Our approach is based on using information induced from Wikipedia so as to estimate the *absolute query difficulty*. That is, the Wikipedia-based features that we present attest to the resultant retrieval effectiveness of using the query regardless of the corpus on which search is performed. Therefore, a clear advantage of our approach is for non-cooperative search settings where corpus-based information is not available [4].

Empirical evaluation performed with TREC datasets attests to the merits of our approach. First, when used alone, our approach yields decent prediction quality. Furthermore, we integrate the approach with previously proposed state-of-the-art pre-retrieval predictors that analyze the corpus used for retrieval. The resultant prediction quality is consistently better than that of using these predictors alone.

SIGIR'14, July 6–11, 2014, Gold Coast, Queensland, Australia.

http://dx.doi.org/10.1145/2600428.2609553.

2. RELATED WORK

As noted, in contrast to our approach that is corpus-independent, most pre-retrieval prediction methods analyze the corpus that serves for retrieval [8, 12, 16, 11, 22, 10].

There is very little work on corpus-independent query-performance prediction [15, 11]. Query-syntactic features and WordNet-based features were used for prediction. In contrast to our work, Wikipedia-based information was not used and corpus-independent predictors were not integrated with corpus-dependent ones. Integrating these predictors [15, 11] with ours is a future venue we intend to explore.

There is some work on pre-retrieval query-performance prediction for the entity retrieval task which was performed over Wikipedia [18]. In contrast, we present predictors for the standard ad hoc retrieval task, and are the first, to the best of our knowledge, to use Wikipedia to this end. Furthermore, almost all of the features we use for prediction here were not used in this work on entity retrieval [18]

Finally, we note that most of the Wikipedia-based information sources that we use — titles, content, links and categories — were used to improve retrieval effectiveness (e.g., [1, 19, 3, 13]) rather than to predict effectiveness.

3. OUR APPROACH

Let q and D be a query and a document corpus, respectively. The task we pursue, a.k.a. pre-retrieval query-performance prediction [5], is estimating, *prior to retrieval time*, the effectiveness of a search that will be performed by *some* retrieval method over D in response to q.

The most effective pre-retrieval predictors utilize information induced from the corpus D [11]. In contrast, we devise predictors that use information induced from the Wikipedia

corpus. We use the predictors as features in a regression framework to estimate the Average Precision (AP) of retrieval performed over D. Thus, the approach we present is independent of D and is, in essence, designed to estimate the *absolute difficulty* of q.

Notational conventions. Herein, we refer to a subset of q's terms as a sub-query. A Wikipedia page p is defined to be associated with a set of terms if its title contains at least one of the terms in the set. A_q denotes the set of pages associated with q (i.e., with its set of terms $\{q_i\}$). A page p is *exact match* to a sub-query, if there is a 1-1 match between the sub-query's terms and the page title. The maximal exact match length, $MEML$, is the size of the largest sub-query (i.e., the number of its terms) for which an exact match holds. The set of Wikipedia pages for which the maximal match holds is referred to as M_{MEML}. The set of pages that are an exact match to the entire query q is denoted E_q.

We use S_1, S_2 and S_3 to denote the sets of sub-queries that contain one, two and three terms, respectively. Predictors marked with * are computed separately for each of the sets. Predictors that are not marked with * are computed only for single query terms (i.e., sub-queries of size 1).

Computing some of our proposed features requires retrieval over Wikipedia. However, we note that Wikipedia is a very small corpus (4.5 million documents), specifically, in comparison to large-scale Web corpora.

The predictors we propose can be categorized into three groups, based on the information provided by Wikipedia that they utilize: *titles and content*, *links and categories* and *previously proposed corpus-based predictors adapted for Wikipedia*. We next describe each of these groups.

3.1 Titles and Content

The following predictors use information in the titles and content of Wikipedia pages to (mainly) measure aspects of the Wikipedia coverage of sub-queries of q.

- $TNWP^*$ is the sum of the sizes of sets of pages associated with a sub-query in S_i; $i \in \{1,2,3\}$. This predictor can be thought of as quantifying the "scope" of q in Wikipedia and is conceptually similar a previously proposed corpus-based predictor [12].
- NWP^*_{avg} is the average size of a set of pages associated with a sub-query in S_i; $i \in \{1,2,3\}$.
- NWP^*_{stdev} is the standard deviation of the sizes of sets of pages that are associated with a sub-query in S_i; $i \in \{1,2,3\}$.
- NEM^* is the number of pages that constitute an exact match for a sub-query in S_i; $i \in \{1,2,3\}$.
- WEM is the maximum size, denoted $|q|_{max}$ ($\in \{1,2,3\}$), of a sub-query for which there is an exact match page.
- $PWEM$ is the average size of a sub-query which has an exact match.
- $LPEM_{avg}$ is the average length (measured by the number of terms) of the pages in $E_q \cup M_{MEML}$; i.e., those that are exact match to q or a maximal exact match.
- SPT_{avg} is a conceptual reminiscent of a previously proposed post-retrieval predictor [20]. For each pair of query terms we determine the number of pages associated with both. We normalize this number with respect to the number of pages associated with each of the two terms and average the two resultant values. The values obtained for all pairs of query terms are then averaged. Thus, the prediction value presumably attests to the coherence of query aspects represented by Wikipedia pages associated with the query terms.

3.2 Links and Categories

The most common pages in Wikipedia are *entity pages*, which refer to people, objects, etc. Entity pages are assigned with human-generated categories which constitute a rich source of structured information. In addition, almost every page in Wikipedia contains links to other related pages. We next present predictors that extract information from links and categories.

- LAC^* is the overall number of links that contain at least one of q's terms in their anchor text.
- PDC^*_{avg} & PDC^*_{stdev} are the average and standard deviation, respectively, computed over sub-queries in S_i, of the number of categories that appear in at least one of the pages associated with a sub-query; $i \in \{1,2,3\}$.
- NOL_{avg} & NOL_{stdev} are the average and standard deviation, respectively, of the number of outgoing links for a Wikipedia page in M_{MEML}.
- NIL_{avg} & NIL_{stdev} are the average and standard deviation, respectively, of the number of incoming links for a Wikipedia page in M_{MEML}.
- $ROIL_{avg}$ & $ROIL_{stdev}$ are the average and standard deviation, respectively, of the ratio between the number of incoming and outgoing links for a page in M_{MEML}. Pages with no outgoing links are ignored.
- LBG_{avg}, LBG_{stdev} & LBG_{max}. We count the overall number of links that point from pages associated with a sub-query in S_1 (i.e., query term) to pages associated with another sub-query in S_1. We aggregate these counts over all ordered pairs of sub-queries in S_1 using the average, standard deviation, and maximum functions, respectively. The resultant predictors presumably quantify the coherence of query aspects represented by Wikipedia pages associated with the query terms.
- IGC_{avg} & IGC_{stdev}. For a set of pages associated with a sub-query in S_1 (i.e., a query term) we compute the percentage of outgoing links which point from a page in the set to another page in the set. The average and standard deviation of the percentages over all the sets that correspond to the different sub-queries in S_1 serve for prediction. This is yet another method that quantifies the presumed coherence of the query's aspects as manifested in the Wikipedia pages associated with the query terms.

3.3 Adaptations of previously proposed corpus-based pre-retrieval predictors

We next adapt previously proposed pre-retrieval predictors to predict q's performance over Wikipedia. We hypothesize that this predicted performance can attest to q's performance over *any* corpus.

- $WSCS$ (Wikipedia Simplified Clarity Score). This is our Wikipedia-based variant of the Simplified Clarity

Collection	Data	# of Docs	Topics
ROBUST	Disks 4&5-CR	528,155	301-450, 601-700
WT10G	WT10G	1,692,096	451-550
GOV2	GOV2	25,205,179	701-850
ClueWeb	ClueWeb'09 (Category B)	50,220,423	1-150

Table 1: TREC data used in Section 4.1.

Score [12]. The prediction value is the KL divergence between a unigram language model induced from q and that induced from the document that results from concatenating the pages in A_q, i.e., the pages associated with at least one of q's terms.

- $SDQT_{avg}$ & $SDQT_{max}$ quantify the *semantic distance between query terms*. For each sub-query in S_1, i.e., a query term, we construct a language model from the concatenation of all pages associated with it. We then compute the KL divergence between each pair of language models that correspond to sub-queries, and use for prediction the average and maximal values computed over pairs.
- PMI_{avg} & PMI_{max} are the *Pairwise Mutual Information* predictors. We compute the average and maximal PMI between two query terms as described in [10]. The documents set used for computing PMI is A_q.

4. EVALUATION

We turn to evaluate the prediction quality of the proposed methods. In Section 4.1 we evaluate the prediction quality for a standard language-model-based retrieval. Then, in Section 4.2 we study the effectiveness of our methods in predicting the performance of *runs* submitted to TREC.

4.1 Predicting the performance of language-model-based retrieval

Table 1 presents the details of the TREC datasets used for evaluation. ROBUST consists of (mainly) newswire articles, WT10G contains Web pages that can be of low quality (e.g. spam), and GOV2 is a crawl of the .GOV domain; ClueWeb ('09) is a large scale noisy Web collection, which contains also Wikipedia pages.

We used the TREC topics titles as queries. Krovetz stemming and stopwords removal, using the INQUERY list, were applied to both documents and queries. The experiments were conducted using the Indri toolkit[1]. The query likelihood method [17] serves for the retrieval method. Document d's retrieval score with respect to query q is: $\log \prod_{q_i \in q} p(q_i|d)$; $p(q_i|d)$ is the probability assigned to q_i by a Dirichlet smoothed unigram language model induced from d with the smoothing parameter set to 1000 [21]. The same smoothing approach was applied to the language models used by our Wikipedia-based predictors, except for the query language model which was a simple maximum likelihood estimate.

As we integrate the proposed predictors in a regression framework, we follow previous recommendations [10][2], and past practice [10, 7, 2], and measure prediction quality by the root mean squared error (RMSE) between the predicted average precision (AP@1000) and the ground truth AP@1000 determined by using TREC's relevance judgments. Statistically significant differences of prediction quality are determined using the paired t-test.

[1]www.lemurproject.org

[2]It was shown that measuring prediction quality using Pearson correlation between the predicted and the actual value of average precision might not result in reliable results when integrating predictors using regression.

Collection	Wikipedia	Corpus	Integrated
ROBUST	0.191	0.188^{w}	$\mathbf{0.186}^{w}$
WT10G	0.188	0.186^{W}	$\mathbf{0.175}^{W}_{c}$
GOV2	0.188	0.184^{W}	$\mathbf{0.176}^{W}_{C}$
ClueWeb	0.162	0.163	$\mathbf{0.155}^{w}_{c}$

Table 2: Prediction quality for language-model-based retrieval of the Wikipedia-based and corpus-based predictors and their integration. 'W' ('w') and 'C' ('c') mark statistically significant differences with the Wikipedia-based and corpus-based predictors, respectively, with $p \leq 0.05$ ($p \leq 0.1$). Boldface marks the best result in a row.

We compared the prediction quality of using three sets of predictors. The first is our (42) proposed Wikipedia-based corpus-independent predictors. We also use the query length as a predictor which results in 43 predictors.

The second set of predictors is composed of three families of state-of-the-art pre-retrieval corpus-based predictors [11] applied to the corpus at hand (D). These use the following statistics for a single query term: (i) inverse-document frequency (IDF) [8, 12, 11], (ii) variability of its tf-idf value in documents in the corpus in which it appears (VAR.TF.IDF) [22], and (iii) its similarity to the corpus (SQC) [22]. To aggregate the term-based statistics across the query terms, for each of the three families of predictors, we used the average, standard deviation and maximum functions. Overall, 9 predictors were created for this set.

The third set of predictors that we consider is a merge of the first two sets (i.e., our Wikipedia-based predictors and the previously proposed corpus-based predictors).

We integrate the predictors in each set using Lasso regression [9] so as to predict AP@1000. Leave-one-out cross validation is performed over the query set. The prediction quality numbers (RMSE) are presented in Table 2.

We see in Table 2 that the corpus-based predictors outperform, to a statistically significant degree, the Wikipedia-based predictors for three out of four collections. This finding should come as no surprise as the Wikipedia-based predictors do not analyze the corpus upon which search is performed. For ClueWeb, the Wikipedia-based predictors outperform the corpus-based predictors, but not to a statistically significant extent. (This could be the result of ClueWeb containing Wikipedia pages.) More importantly, the integrated set of Wikipedia-based and corpus-based predictors results in improved prediction quality. Specifically, for the Web collections, the integration of the two sets yields substantial and statistically significant improvements of prediction quality over using either set alone.

Analysis of the relative effectiveness of the Wikipedia predictors reveals interesting insights. The most effective predictor was $TNWP$, which was used by the regressor for all four collections. The other leading predictors were LBG, PDC and PMI, each used for three out of the four collections. We believe that the fact that all aspects of Wikipedia: page titles, links, categories and page text, were used for prediction attests to the merits of using Wikipedia for query-performance prediction.

track	run	Wikipedia	Corpus	Integrated
TREC07	median	0.179	$\mathbf{0.165}^{W}$	0.169
	top5	0.32	0.32	**0.315**
TREC08	median	0.171	0.164	**0.156**
	top5	**0.337**	**0.337**	**0.337**
TREC12	median	0.157	0.151	**0.144**
	top5	0.277	0.277	$\mathbf{0.258}^{W}_{C}$
Clue09	median	0.114	0.114	$\mathbf{0.104}^{W}_{C}$
	top5	0.173	$\mathbf{0.156}^{W}$	0.162^{W}
Clue10	median	0.103	**0.099**	0.1
	top5	**0.173**	0.182^{w}	$\mathbf{0.173}_{c}$
Clue11	median	0.115	0.141^{W}	$\mathbf{0.11}^{W}_{C}$
	top5	0.146	0.152	**0.145**

Table 3: Prediction quality for TREC runs. 'W' ('w') and 'C' ('c') mark statistically significant differences with the Wikipedia-based and corpus-based predictors, respectively, with $p \leq 0.05$ ($p \leq 0.1$). Boldface marks the best result in a row.

4.2 Predicting the performance of TREC runs

Insofar, we predicted the performance of language-model-based retrieval. To evaluate prediction quality for a variety of retrieval methods, we use TREC runs. Specifically, we perform the evaluation with runs submitted to "ROBUST-based" tracks (TREC7, TREC8, TREC12) and ClueWeb-based tracks from 2009-2011 (denoted Clue09, Clue10 and Clue11), as these represent two different types of corpora, namely, newswire and Web.

We focus on two "pseudo" runs: (i) a median run, where the AP@1000 value per query is the median of the AP@1000 values of all runs for that query; b) a top5 run, where the AP@1000 per query is the average AP@1000 values for that query posted by the five best (MAP) performing runs in a track. Thus, the prediction quality for these two pseudo runs reflects the ability to predict the performance of median and highly effective retrieval methods.

To predict the performance of the runs in each track, we trained the Lasso regressor over the queries in the other two tracks that use the same collection; e.g., to predict the performance for TREC7, we trained over the TREC8 and TREC12 queries. For the training phase, we used the query-likelihood retrieval model as was the case in Section 4.1.

Table 3 presents the prediction quality numbers. Overall, the results are consistent with those presented for the language-model-based retrieval. For the ROBUST-based tracks, the integration of Wikipedia-based and corpus-based predictors can be of merit, although the improvements are rarely statistically significant. For the ClueWeb tracks, the integration often posts statistically significant improvements. Furthermore, in some cases for ClueWeb, the Wikipedia-based predictors outperform the corpus-based predictors.

5. CONCLUSIONS AND FUTURE WORK

We presented a novel corpus-independent approach to pre-retrieval prediction of query performance. That is, the effectiveness of retrieval performed in response to a query is predicted without analyzing the corpus upon which search is performed. Our approach is based on quantifying properties of the query using Wikipedia. Empirical evaluation demonstrated the merits of our approach.

For future work, we plan to integrate our prediction methods with state-of-the-art post-retrieval predictors [14]. In addition, we intend to evaluate the quality of our predictors in a cross-corpus experimental setting as in [6].

6. ACKNOWLEDGMENTS

We thank the reviewers for their comments. This paper is based on work supported in part by the Israel Science Foundation under grant no. 433/12, by Google's and Yahoo!'s faculty research awards, and by an IBM Ph.D. fellowship.

7. REFERENCES

[1] J. Arguello, J. L. Elsas, J. Callan, and J. G. Carbonell. Document representation and query expansion models for blog recommendation. In *Proceedings of ICWSM*, 2008.

[2] N. Balasubramanian, G. Kumaran, and V. R. Carvalho. Predicting query performance on the web. In *Proceedings of SIGIR*, pages 785–786, 2010.

[3] K. Balog, M. Bron, and M. De Rijke. Category-based query modeling for entity search. In *Advances in Information Retrieval*, pages 319–331. Springer, 2010.

[4] J. Callan. Distributed information retrieval. In W. Croft, editor, *Advances in information retrieval*, chapter 5, pages 127–150. Kluwer Academic Publishers, 2000.

[5] D. Carmel and E. Yom-Tov. *Estimating the Query Difficulty for Information Retrieval*. Synthesis Lectures on Information Concepts, Retrieval, and Services. Morgan & Claypool Publishers, 2010.

[6] D. Carmel, E. Yom-Tov, A. Darlow, and D. Pelleg. What makes a query difficult? In *Proceedings of SIGIR*, pages 390–397, 2006.

[7] K. Collins-Thompson and P. N. Bennett. Predicting query performance via classification. In *Proceedings of ECIR*, pages 140–152, 2010.

[8] S. Cronen-Townsend, Y. Zhou, and W. B. Croft. Predicting query performance. In *Proceedings of SIGIR*, pages 299–306, 2002.

[9] C. Hans. Bayesian lasso regression. *Biometrika*, 96(4):835–845, 2009.

[10] C. Hauff, L. Azzopardi, and D. Hiemstra. The combination and evaluation of query performance prediction methods. In *Proceedings of ECIR*, pages 301–312, 2009.

[11] C. Hauff, D. Hiemstra, and F. de Jong. A survey of pre-retrieval query performance predictors. In *Proceedings of CIKM*, pages 1419–1420, 2008.

[12] B. He and I. Ounis. Inferring query performance using pre-retrieval predictors. In *Proceedings of SPIRE*, pages 43–54, 2004.

[13] E. Hoque, G. Strong, O. Hoeber, and M. Gong. Conceptual query expansion and visual search results exploration for web image retrieval. In *Advances in Intelligent Web Mastering–3*, pages 73–82. Springer, 2011.

[14] O. Kurland, A. Shtok, S. Hummel, F. Raiber, D. Carmel, and O. Rom. Back to the roots: a probabilistic framework for query-performance prediction. In *Proceedings of CIKM*, pages 823–832, 2012.

[15] J. Mothe and L. Tanguy. Linguistic features to predict query difficulty. In *ACM SIGIR 2005 Workshop on Predicting Query Difficulty - Methods and Applications*, 2005.

[16] F. Scholer, H. E. Williams, and A. Turpin. Query association surrogates for web search. *JASIST*, 55(7):637–650, 2004.

[17] F. Song and W. B. Croft. A general language model for information retrieval (poster abstract). In *Proceedings of SIGIR*, pages 279–280, 1999.

[18] A.-M. Vercoustre, J. Pehcevski, and V. Naumovski. Topic difficulty prediction in entity ranking. In *Proceedings of INEX*, pages 280–291, 2009.

[19] Y. Xu, G. J. Jones, and B. Wang. Query dependent pseudo-relevance feedback based on wikipedia. In *Proceedings of SIGIR*, pages 59–66, 2009.

[20] E. Yom-Tov, S. Fine, D. Carmel, and A. Darlow. Learning to estimate query difficulty: including applications to missing content detection and distributed information retrieval. In *Proceedings of SIGIR*, pages 512–519, 2005.

[21] C. Zhai and J. D. Lafferty. A study of smoothing methods for language models applied to ad hoc information retrieval. In *Proceedings of SIGIR*, pages 334–342, 2001.

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