

Query Difficulty for Digital Libraries

Claudia Hauff

Human Media Interaction,
University of Twente, the Netherlands
c.hauff@ewi.utwente.nl

Abstract. Research in *query difficulty* deals with the task of determining how well or poorly a query is likely to perform given a search system and a collection of documents. If the performance of queries can be established in advance of or during the retrieval stage, specific steps can be taken to improve retrieval. In this report, a categorization of existing algorithms is introduced and various algorithms are briefly assessed. The research questions are outlined and where possible first results are reported.

1 Introduction

Research in *query difficulty* deals with the task of determining how well or poorly a query is likely to perform given a search system and a collection of documents. If the performance of queries can be established in advance of or during the retrieval stage, specific steps can be taken to improve retrieval. For instance, if a query is estimated to fail because of ambiguity [CTZC02] or non-specificity [HO04], the user can be offered a clarification form [TVFZ07] to pick the query's topic in order to aid the search system. Alternatively, the retrieval system can redirect the query to a different collection or utilize a different retrieval approach [WRBH08]. Moreover, predicted failures can be exploited by a search engine to discover missing content [YTFCD05] and subsequently to add the missing topical aspects to the collection (through focused crawling [CvdBD99] for example). In the opposite case, if the query is predicted to result in a satisfactory document ranking, query expansion [ACR04] can further improve the results. Apart from benefiting the search engine or collection keepers, precise query difficulty algorithms can also help the user to better understand how to find information in large scale collections such as the Web or distributed collections that have been made accessible via a domain-specific portal.

In recent years, a number of algorithms have been shown to perform well on established news report test collections such as TREC Volumes 4+5. However, the same algorithms applied to the more noisy Web test collections such as WT10g [Sob02] or GOV2 [CCS04] generally result in poor query difficulty estimations. Why this is the case is not yet well understood. Several factors influence the quality of query difficulty algorithms and as part of the planned research each of them will be explored:

- the collection (newspaper reports versus Web pages)

- the query set (long queries versus queries with 2-3 keywords)
- the query type (transactional, informational, navigational queries)
- the retrieval approach (vector space model versus language modeling).

The key contributions and research questions of the thesis which will be discussed in subsequent sections are aimed to be

- (a) an analysis of the strengths and weaknesses of state of the art query difficulty algorithms
- (b) a formalization of the heuristics that the algorithms are based on
- (c) a proposal of a novel query difficulty algorithm that is based on the results found in (a) and (b)
- (d) evaluation of the proposed novel algorithm
- (e) an analysis of the currently employed evaluation measures

One of the goals of this work is to employ query difficulty algorithms not just to established TREC test collections but to also apply the algorithms to areas related to Digital Libraries. One possibility are scholarly repositories such as Citeseer¹ and arXiv.org²: if a query is predicted to fail, it can indicate poor coverage of the topic by the repository or inadequate understanding of the topic by the searcher.

In a first step, existing algorithms are investigated. A categorization of query difficulty algorithms is introduced and each algorithm is theoretically and experimentally evaluated. This work should lead to several insights including formal explanations for the fact that several proposed algorithms are almost always guaranteed to have the same performance [HHdJ08]. Moreover, the influence of the retrieval approach on the performance of the algorithms is shown and explained. As part of the investigation, an improved version of *Query Clarity* [CTZC02], a state-of-the-art algorithm, is proposed which is less sensitive to the specific parameter settings and the retrieval approach than the original algorithm. A detailed description of the so-called *Improved Query Clarity* can be found in [HMBY08]. As part of the analysis, it is attempted to formalize the heuristics the query difficulty algorithms base their performance on. The aim is to be able to formally show to some extent if an algorithm is going to perform well or not. Lastly, the evaluation measures, namely correlation coefficients, will be explored and a number of shortcomings will be discussed.

Before elaborating on the proposed contributions to the area of query difficulty in the next sections, an overview of related work is given.

2 Related Work

Query difficulty algorithms broadly fall into two categories: *pre-retrieval predictors* and *post-retrieval estimators*. Pre-retrieval predictors predict the performance of a query before the retrieval stage is reached and are, thus, independent

¹ <http://citeseer.ist.psu.edu/>

² <http://arxiv.org/>

of the ranked list of results; essentially, they are search-independent. Such predictors base their predictions solely on query terms, collection statistics and possibly external sources such as WordNet [Fel98], which provides information on the terms' semantic relationships, or Wikipedia³. Notably, ranked list independence is widely viewed as one of the weaknesses of the pre-retrieval paradigm, since the performance of the query depends on the particular retrieval approach chosen and the various parameter settings such as the amount of smoothing in the language modeling framework [PC98,ZL01]. On the other hand, post-retrieval estimators base their estimations on the ranked list of results, which in turn provides more information, making accurate query difficulty estimations easier to achieve. This benefit is somewhat offset by the added requirement of a retrieval stage, as, given a query is deemed to be difficult, another retrieval stage might be necessary. Note that if the ranked list of results is available, an algorithm *estimates* the quality of the list, whereas algorithms that do not rely on the ranked list have to *predict* its quality [VCMF08]. In practice however, prediction and estimation are often used interchangeably.

2.1 Pre-retrieval Algorithms

Pre-retrieval predictors can be divided into four different groups according to the heuristic(s) they exploit (Figure 1).

The *specificity* based predictors [HO04,MT05,SWT04,ZST08] predict a query to perform better with increased specificity. How the specificity is determined further divides these predictors into collection-statistics based and query based predictors. The average number of characters $AvQL$ in a query is the most basic predictor possible - the longer on average the query terms, the more specific the query is predicted to be. The average inverse document frequency $AvIDF$ determines the specificity by the document frequency of the query terms and thus it is a collection statistics based specificity predictor. Queries with low frequency terms are predicted to achieve a better performance than queries with high frequency terms. He et al. [HO04] evaluated a number of algorithms including *Query Scope* which bases the prediction on the number of documents in the collection that contain at least one of the query terms.

A variety of predictors exploits the query terms' *ambiguity* to predict the difficulty of the query [HLdR08,MT05]. In such a scheme, if a term always appears in the same or similar context, the term is considered to be unambiguous, if on the other hand the term appears in many different contexts it is considered to be ambiguous. For example, the term *tennis* will mainly appear in the context of sports, rarely will it be mentioned in documents about finances or politics. The term *field* however is more ambiguous and can easily occur in sports articles, agriculture articles or even politics (consider *field of Democratic candidates*). Ambiguity is somewhat related to specificity, as an ambiguous term can have a high document frequency, but there are exceptions; *tennis* might not be specific in a corpus containing a lot of sports related documents, but it is unambiguous

³ <http://www.wikipedia.org/>

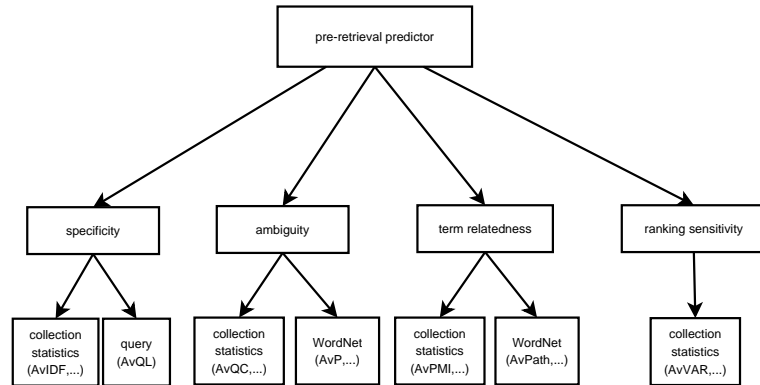


Fig. 1. Categories of pre-retrieval predictors.

and while specificity based predictors would classify it as being a hard query, ambiguity based predictors would not. The predictor *AvQC* [HLdR08] belongs to this category. It determines a query term’s ambiguity by calculating the similarity between all documents that contain the query term. The lower the ambiguity, the higher the similarity between all those documents. Instead of basing the predictors on the collection, ambiguity can also be calculated with an external source such as WordNet. WordNet [Fel98] is an online lexical database developed at Princeton University, inspired by psycholinguistic theories. WordNet’s building blocks are sets of synonymous terms⁴, called *synsets*, each representing a lexical concept and each connected to others through a range of semantic relationships. The number of synsets a query term belongs to, that is its *polysemy* value *AvP*, is also a valid ambiguity based predictor.

The disadvantage of predictors in the first two categories stems from their lack of consideration for the relationship between terms; the query *political field* is actually not ambiguous due to the relationship between the two terms, but an ambiguity based predictor is likely to predict a poor performance since *field* can appear in many contexts. Similarly for a specificity based predictor - *field* will be an often occurring term in a general corpus. The average pointwise mutual information *AvPMI* of a query predicts a better performance for queries whose terms exhibit a relationship. Again, not only collection statistics based predictors are possible, but also WordNet based predictors - the average path length *AvPath* between all query terms indicates the amount of relationship between them.

Finally, *ranking sensitivity* based predictors exploit the distribution of term weights across the collection to predict how difficult it will be to distinguish the documents from each other. If the term weights across all documents containing a particular query term are similar, there is little evidence how to rank those documents. Conversely, if the term weights differ widely across the collection,

⁴ A term can be a single word, a compound or a phrase.

ranking becomes easier. Since the established retrieval approaches such as the vector space model or language modeling rely on (inverse) term and document frequencies, in [ZST08] the term weight is based on TF.IDF and the averaged term weight variability $AvVAR$ is the normalized sum of the query term weight deviations.

The pre-retrieval predictors introduced in this section are only a subset of all proposed pre-retrieval predictors, but they emphasize our categorization well. In the next section, an overview of a number of post-retrieval predictor algorithms is given.

2.2 Post-retrieval Algorithms

Retrieval estimators relying on the initial ranked list of results can similarly be categorized (Figure 2). None of the existing post-retrieval estimators exploit external resources, all rely either on a comparison between the top ranked documents and the collection as a whole or on a comparison between the top ranked documents of two or more ranked lists of results.

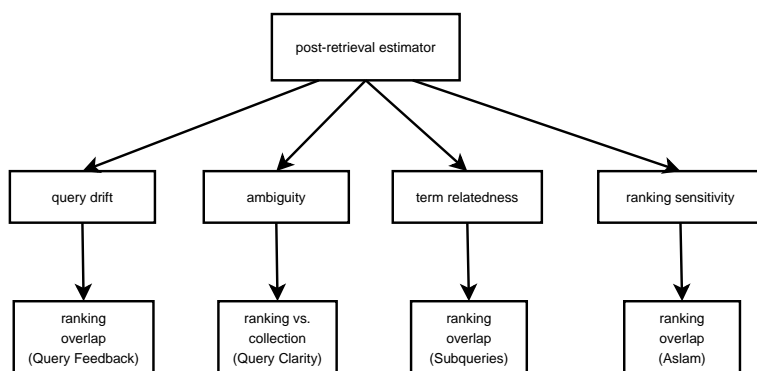


Fig. 2. Categories of post-retrieval estimators.

Query drift is the change of focus of a query due to faulty query expansion' [MSC98]. A direct application of that concept was introduced by Zhou and Croft [ZC07] who investigated two approaches to estimating query difficulty in Web search environments. *Query Feedback* frames query difficulty estimation as a communication channel problem. The input is query Q , the channel is the retrieval system and the ranked list L is the noisy output of the channel. From the ranked list L , a new query Q' is generated, a second ranking L' is retrieved with Q' as input and the overlap between L and L' is used as difficulty score. The lower the overlap between the two rankings, the higher the query drift and thus the more difficult the query. *Weighted Information Gain* measures "the change in information about the quality of retrieval from an imaginary state

that only an average document is retrieved [estimated by the collection model] to a posterior state that the actual search results are observed”.

Cronen-Townsend et al. [CTZC02] introduced Clarity Score which measures a query’s *ambiguity* towards a collection. The approach is based on the intuition that the top ranked results of an unambiguous query will be topically cohesive and terms particular to the topic will appear with high frequency. The term distribution of an ambiguous query on the other hand is assumed to be more similar to the collection distribution, as the top ranked documents cover a variety of topics. An extension of Clarity Score that takes into account the temporal profiles of the queries was proposed by Diaz et al. [DJ04].

Yom-Tov et al. [YTFCD05] presented an estimator based implicitly on the *relationship* between the query terms: they compared the ranked list of the original query with the ranked lists of the query’s constituent terms. The idea behind the approach is that for well performing queries the result list does not change considerably if only a subset of query terms is used. They applied machine learning approaches, exploiting several features, among others the overlap in the top ranked documents between the original query and the subqueries, the score of the top ranked document and the number of query terms.

A very different idea is the utilization of the *sensitivity* of the query to the retrieval approach. Aslam et al. [AP07] consider a query to be difficult if different ranking functions retrieve diverse ranked lists. If the overlap between the top ranked documents is large across all ranked lists, the query is deemed to be easy.

3 Overview of Preliminary Results

3.1 The Explanatory Power of Correlation as an Evaluation Measure

Test collections such as WT10g or the GOV2 come with a set of queries and relevance judgments, that is, for each query the set of relevant documents in the collection are known. A retrieval approach is then evaluated by calculating the *average precision* of the ranked document list returned for each query. If a retrieval approach returns a ranked list for a query that has a low average precision, a good query difficulty algorithm is expected to predict or estimate poor performance. Thus, the better a query difficulty algorithm is, the more closely its predictions or estimations resemble the average precision values. The similarity of the two lists - query difficulty scores and average precision scores - is determined by their *correlation coefficient* which is an indicator of the degree of relationship between them.

A number of correlation measures exist. In the evaluation of query difficulty algorithms three different correlation coefficients are typically employed: *Pearson’s* product-moment correlation coefficient (also known as linear correlation coefficient), *Spearman’s* rank correlation coefficient and *Kendall’s* tau rank correlation coefficient. The latter two are *rank correlations* which means they are calculated on the ranks of raw scores instead directly on the raw scores.

Pearson's correlation coefficient on the other hand assumes a linear relationship between the scores of the two lists. This difference in calculation is not only superficial. Pearson's correlation coefficient reaches 1 (ideal value), if the prediction scores Y_i equate to the average precision scores X_i up to constant and a multiplier: $Y_i = a \times X_i + b$. Therefore, the prediction scores can be viewed as *predicted/estimated average precision* scores. Rank correlations make no assumptions about the type of relationship between the lists of scores. Both lists are converted to ranks and the correlation of the ranks is measured. In this case, the ranks give an indication of each query's effectiveness relative to the other queries in the list but no quantitative prediction is made about the retrieval score of the query. Which correlation measure to utilize, depends on the task at hand. If one is interested in recognizing poorly performing queries from a set of given queries for instance, a rank correlation based evaluation is recommendable. However, if one query is submitted to several collections and one is interested in the question which of the returned ranked lists has the highest retrieval performance, the linear correlation coefficient would be a better evaluator. When no particular task is considered, usually both types of correlation are reported.

Knowing the correlation coefficient of an algorithm however does not directly translate to knowing how it will influence the performance of the retrieval system when applied. Possible questions to ask include

- Is it worth (=does the system improve?) running a time consuming query difficulty algorithm that has a reported correlation coefficient of 0.4?
- If an algorithm improves the correlation coefficient by 0.1, by how much does the system's retrieval performance increase?
- At what threshold is a correlation coefficient high enough such that the first/second/third quartile of all cases improve over the baseline?

Furthermore, the question remains, if correlation coefficients are the best option to evaluate the algorithms. For example, a very important evaluation metric is precision at 10 ($P@10$) or 20 documents, as the user is most interested in the top ranked documents and in a real search engine setting, rarely looks beyond the first page of results. Correlation measures fail here: $P@10$ is limited to 10 different scores - if none of the top 10 ranked documents is relevant, $P@10 = 0$, if all 10 documents are relevant $P@10 = 1$. The lack of discrimination between the scores makes correlation coefficients unusable. This restriction leads to the idea of developing an alternative evaluation measure that does not depend on correlation coefficients.

3.2 Assessment and Comparison of Pre-Retrieval Predictors

A number of pre-retrieval predictors have been evaluated in some depth to offer insight into the performance of each, when considered both in isolation and in view of the performance of others [HHdJ08]. Upon initial inspection, a number of predictors exhibit substantial similarities which are highlighted both mathematically and experimentally. As a first contribution and in order to present a more

abstract view of the existing predictors, categories of pre-retrieval predictors were identified as already presented in the previous section. The pre-retrieval predictors were evaluated on three widely different TREC collections: TREC Volumes 4+5 (a collection of news reports), WT10g [Sob02] (a collection of Web pages) and GOV2 [CCS04] (a collection of Web pages from the .gov domain). TREC Volumes 4+5 proved to be the easiest collection in the sense that the predictors achieved the best performance. WT10g on the other hand is the most difficult to predict the performance for, none of the pre-retrieval predictors achieved significant results. Among all predictors, ranking sensitivity based and specificity based predictors exploiting collection statistics were the most robust, ambiguity and term relatedness played a smaller role. Furthermore, WordNet based and query based predictors showed only significant results for TREC Volumes 4+5, on the two larger collections they failed.

3.3 Improved Query Clarity

In the work presented in [HMBY08], we evaluated two state-of-the art post-retrieval estimators, namely Query Clarity [CTZC02] and Query Feedback [ZC07]. Subsequently, based on our observations, two improvements were suggested for Query Clarity. A major issue of both algorithms is their sensitivity to the particular retrieval approach as well as to their own parameter settings. An example of our implemented version⁵ of Query Feedback is shown in Figure 3. Here, μ represents different parameter settings of the retrieval approach (language modeling with Dirichlet smoothing), while the parameter pair (s, t) on the x-axis is Query Feedback's parameter setting. It is evident that a thorough search through the parameter space is necessary to find the optimal setting for the algorithm.

Query Clarity has a single free parameter (the number of top ranked documents n to use) and exhibits comparable performance variations depending on n , the query set and the collection. We proposed two changes to Query Clarity which set n automatically in a query dependent fashion and exclude high frequency terms to avoid unnecessary addition of noise. These adaptations have been tested on the three TREC collections introduced earlier. Apart from one set of queries, *Improved Clarity* outperformed the baselines in all cases, in some instances by a large margin. Furthermore, the gap between the highest and lowest correlation scores for different retrieval runs is decreased. While a difference remains between the performance of query difficulty algorithms on WT10g and the two corpora TREC Volumes 4+5 and GOV2, we were able to improve the correlation significantly.

3.4 Formalization of Prediction Heuristics

In tandem with the work in [FTZ04] an attempt will be made to formulate and formalize a number of constraints that a prediction algorithm f should satisfy. An

⁵ Query Feedback and Improved Query Clarity were implemented on top of the Lemur Toolkit (<http://www.lemurproject.org/>). The original Query Clarity algorithm already exists in the toolkit.

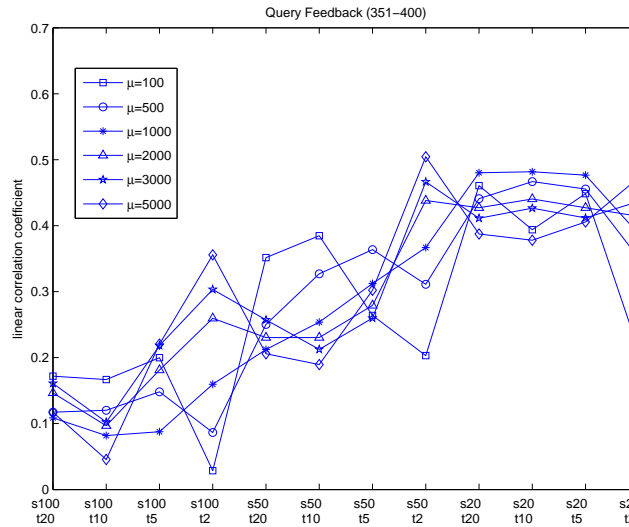


Fig. 3. Parameter sensitivity of Query Feedback, evaluated on TREC Volumes 4+5 with title topics 351-400. The documents were stemmed and stopwords were removed. The retrieval approach utilized for the experiments was language modeling with Dirichlet smoothing (μ is the smoothing parameter).

example of the constraints envisioned for the case of specificity is the following: \mathbf{q}_1 and \mathbf{q}_2 are queries; q_i , q_j and q_k are terms. The document frequency $df(t)$ of term t is the number of documents t occurs in. It is assumed that the larger the predictor score the more specific the query is (and therefore the higher its performance is predicted to be).

1. Given $\mathbf{q}_1 = q_i$ and $\mathbf{q}_2 = q_j$ with $df(q_i) < df(q_j)$ a prediction algorithm should return $f(\mathbf{q}_1) > f(\mathbf{q}_2)$.
2. Given $\mathbf{q}_1 = \{q_i, q_j\}$ and $\mathbf{q}_2 = \{q_i, q_k\}$ with $df(q_k) > df(q_j)$ a prediction algorithm should return $f(\mathbf{q}_1) > f(\mathbf{q}_2)$; that is, if two two-term queries have one term in common, the prediction algorithm should give a higher score to the query containing the more specific term.

A predictor that violates these constraints is expected to perform less well than a predictor that fulfills them. For the case of pre-retrieval specificity based predictors it could be shown that *AvIDF* fulfills both constraints and outperforms all other specificity based pre-retrieval predictors which fulfill the above constraints only conditionally or not at all, further supporting the choice of constraints. This analysis is still in a preliminary stage though, and no conclusive results can be reported yet.

4 Outlook

During my previous research I have focused on the assessment and evaluation of more than 15 pre-retrieval prediction algorithms as well as an in depth investigation of 2 post-retrieval predictors, namely Query Feedback and Query Clarity. In the course of that work, an improvement to Query Clarity was proposed and insights about the pre-retrieval predictors performance over various TREC test collections were gained. Currently, I am working on the categorization of the post-retrieval predictors and the formalization of constraints that prediction algorithms should fulfill. Building on the insights gained, a novel algorithm shall be developed.

During the doctoral consortium I would like to discuss among others what the added value could be of applying query difficulty algorithms to the field of Digital Libraries (e.g. discovering a missed topical aspect of the repository), what class of Digital Libraries would be a good focus and what this focus would imply for my research agenda.

References

- [ACR04] Giambattista Amati, Claudio Carpineto, and Giovanni Romano. Query difficulty, robustness and selective application of query expansion. In *ECIR 2004*, 2004.
- [AP07] Javed A. Aslam and Virgil Pavlu. Query hardness estimation using Jensen-Shannon divergence among multiple scoring functions. In *ECIR 2007*, pages 198–209, 2007.
- [CCS04] Charles Clarke, Nick Craswell, and Ian Soboroff. Overview of the trec 2004 terabyte track. In *Proceedings of the Thirteenth Text REtrieval Conference*, 2004.
- [CTZC02] Steve Cronen-Townsend, Yun Zhou, and W. Bruce Croft. Predicting query performance. In *SIGIR 2002*, pages 299–306, 2002.
- [CvdBD99] Soumen Chakrabarti, Martin van den Berg, and Byron Dom. Focused crawling: a new approach to topic-specific web resource discovery. *Comput. Netw.*, 31(11-16):1623–1640, 1999.
- [DJ04] Fernando Diaz and Rosie Jones. Using temporal profiles of queries for precision prediction. In *SIGIR 2004*, pages 18–24, 2004.
- [Fel98] *WordNet - An Electronic Lexical Database*. The MIT Press, 1998.
- [FTZ04] Hui Fang, Tao Tao, and ChengXiang Zhai. A formal study of information retrieval heuristics. In *SIGIR 2004*, pages 49–56, 2004.
- [HHdJ08] Claudia Hauff, Djoerd Hiemstra, and Franciska de Jong. A survey of pre-retrieval query performance predictors. In *CIKM 2008*, pages 1419–1420, 2008.
- [HLdR08] Jiyin He, Martha Larson, and Maarten de Rijke. Using coherence-based measures to predict query difficulty. In *ECIR 2008*, pages 689–694, 2008.
- [HMBY08] Claudia Hauff, Vanessa Murdock, and Ricardo Baeza-Yates. Improved query difficulty prediction for the web. In *CIKM 2008*, pages 439–448, 2008.
- [HO04] B. He and I. Ounis. Inferring query performance using pre-retrieval predictors. In *The Eleventh Symposium on String Processing and Information Retrieval (SPIRE)*, pages 43–54, 2004.

- [MSC98] Mandar Mitra, Amit Singhal, and Chris. Improving automatic query expansion. In *SIGIR 1998*, pages 206–214, 1998.
- [MT05] Josiane Mothe and Ludovic Tanguy. Linguistic features to predict query difficulty - a case study on previous trec campaigns. In *ACM SIGIR'05 Query Prediction Workshop*, 2005.
- [PC98] J. Ponte and W. Bruce Croft. A language modeling approach to information retrieval. In *SIGIR 1998*, pages 275–281, 1998.
- [Sob02] Ian Soboroff. Does WT10g look like the web. In *SIGIR 2002*, 2002.
- [SWT04] F. Scholer, H.E. 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.
- [TVFZ07] Bin Tan, Atulya Velivelli, Hui Fang, and ChengXiang Zhai. Term feedback for information retrieval with language models. In *SIGIR 2007*, pages 263–270, 2007.
- [VCMF08] Vishwa Vinay, Ingemar Cox, and Natasa Milic-Frayling. Estimating retrieval effectiveness using rank distributions. In *CIKM 2008*, pages 1425–1426, 2008.
- [WRBH08] Ryen W. White, Matthew Richardson, Mikhail Bilenko, and Allison P. Heath. Enhancing web search by promoting multiple search engine use. In *SIGIR 2008*, pages 43–50, 2008.
- [YTFCD05] Elad Yom-Tov, Shai Fine, David Carmel, and Adam Darlow. Learning to estimate query difficulty: including applications to missing content detection and distributed information retrieval. In *SIGIR 2005*, pages 512–519, 2005.
- [ZC07] Yun Zhou and W. Bruce Croft. Query performance prediction in web search environments. In *SIGIR 2007*, pages 543–550, 2007.
- [ZL01] Chengxiang Zhai and John Lafferty. A study of smoothing methods for language models applied to ad hoc information retrieval. In *SIGIR 2001*, pages 334–342, 2001.
- [ZST08] Ying Zhao, Falk Scholer, and Yohannes Tsegay. Effective pre-retrieval query performance prediction using similarity and variability evidence. In *ECIR 2008*, pages 52–64, 2008.