A Study of Collection-based Features for Adapting the Balance Parameter in Pseudo Relevance Feedback

Ye Meng[1], Peng Zhang[1], Dawei Song[1,2], and Yuexian Hou[1]

[1] Tianjin Key Laboratory of Cognitive Computing and Application, Tianjin University, China
[2]The Computing Department, The Open University, United Kingdom
{pzhang, dwsong, yxhou}@tju.edu.cn,
ye.meng04@gmail.com

Abstract. Pseudo-relevance feedback (PRF) is an effective technique to improve the ad-hoc retrieval performance. For PRF methods, how to optimize the balance parameter between the original query model and feedback model is an important but difficult problem. Traditionally, the balance parameter is often manually tested and set to a fixed value across collections and queries. However, due to the difference among collections and individual queries, this parameter should be tuned differently. Recent research has studied various query based and feedback documents based features to predict the optimal balance parameter for each query on a specific collection, through a learning approach based on logistic regression. In this paper, we hypothesize that characteristics of collections are also important for the prediction. We propose and systematically investigate a series of collection-based features for queries, feedback documents and candidate expansion terms. The experiments show that our method is competitive in improving retrieval performance and particularly for cross-collection prediction, in comparison with the state-of-the-art approaches.

Keywords: Information Retrieval; Pseudo-Relevance Feedback; Collection Characteristics

1 Introduction

Pseudo-relevance feedback (PRF) has been proven effective for improving retrieval performance. The basic idea is to assume a certain number of top-ranked documents as relevant and select expansion terms from these documents to refine the query representation [18]. A fundamental question is whether the feedback information are truly relevant to the query. Cao et al. [4] show that the expansion process indeed adds more bad terms than good ones, and the proportions of bad terms in different collections are different. This means that there is noise in the expansion terms and we can not always trust the expansion information. Thus, we need to carefully balance the original query model and the expansion model derived from the feedback documents. If we over-trust the feedback information,

the retrieval performance can be harmed due to the noise in the expansion model. If we under-trust it, we will not be able to take full advantage of the feedback information. Currently, the balance parameter is often manually tested and set to a fixed value across queries for a specific collection, to combine the original query and the expansion terms derived from the feedback documents. Due to the difference between different collections and different queries, this parameter should be set differently. Recently, Lv and Zhai [10] present a learning approach to adaptively predict the optimal weight of the original query model for different queries and collections. They explore a number of features and combine them using a regression approach for the prediction. The features they used are mostly based on the original query and feedback information, yet do not sufficiently consider features of the candidate expansion terms and the collection.

It has long been recognized in information retrieval that document collection has a great impact on the performance of a retrieval model [17]. In this paper, we propose and systematically investigate a set of collection-based features about queries, feedback documents and candidate terms, which are complementary to the features used in Lv and Zhai [10]. Specifically, three types of features are studied, including (1) Information amount of query: we suppose that a query is more reliable when it carries more information; (2) Reliability of feedback documents; (3) Reliability of candidate terms: We will trust the feedback documents and candidate terms only when they are highly reliable. The proposed features are feed into a logistic regression model to predict the feedback parameter.

2 Related Work

Pseudo-relevance feedback has been implemented in different retrieval models : e.g., vector space model, probabilistic model, and language model. In the vector space model [6], feedback is usually done by using the Rocchio algorithm, which forms a new query vector by maximizing its similarity to relevant documents and minimizing its similarity to non-relevant document. The feedback method in classical probabilistic models [3][16] is to select expanded terms primarily based on Robertson/Sparck-Jones weight. In the language modeling approaches [9][20], relevance feedback can be implemented through estimating a query language model or relevance model through exploiting a set of feedback documents. All those works used a fixed parameter to control the balance parameter between original query and feedback information.

Recently, Lv and Zhai [10] present a learning approach to adaptively predict the optimal balance parameter for each query and each collection. They leverage state-of-the-art language models for ranking documents and use logistic regression to optimize an important parameter inside the language modeling framework. Three heuristics to characterize feedback balance parameter are used, including the discrimination of query, discrimination of feedback documents and divergence between query and feedback documents. These three heuristics are then taken as a road map to explore a number of features and combined them using the logistic regression model to predict the balance parameter. The experiments show that the proposed adaptive relevance feedback is more robust

and effective than the regular fixed-parameter feedback. Nevertheless there is still room to explore when the training and testing sets are different. Our work uses a similar method, but adds features based on characteristic of collection. The experiments show that our method is competitive in improving retrieval performance, in comparison with their approaches.

3 Basic Formulation

The Relevance Model (RM) [21] is a representative and state-of-the-art approach for re-estimating query language models based on PRF [9]. We will carry out our study in the RM framework.

For a given query $Q = (q_1, q_2, ..., q_m)$, based on the corresponding PRF document set F ($|F| = n$), RM estimates an expanded query model [22]:

$$P(w|\theta_F) \propto \sum_{D \in F} P(w|\theta_D) P(\theta_D) \prod_{i=1}^{m} P(q_i|\theta_D) \tag{1}$$

where $P(\theta_D)$ is a prior on documents and is often assumed to be uniform without any additional prior knowledge about the document D. Thus, the estimated relevance model is essentially a weighted combination of individual feedback document models with the query likelihood score of each document as the weight.

The estimated relevance model, $P(w|\theta_F)$, can then be interpolated with the original query model θ_Q to improve performance:

$$P(w|\theta'_Q) = \lambda P(w|\theta_Q) + (1 - \lambda) P(w|\theta_F) \tag{2}$$

where λ is a balance parameter to control the weight of the feedback information. The model in Eq.(2) is often referred to as RM3 [9]. When $\lambda = 1$, we only use the original query model (i.e., no feedback). If $\lambda = 0$, we ignore the original query and rely only on the feedback model.

4 The Proposed Collection-based Features

As aforementioned, due to the difference of collections in document type, size and other characteristics, and the difference of query difficulties, the expansion terms selected from the feedback documents are not always good terms [4]. Accordingly, the balance parameter should be set differently for different collections and queries. In this section, we investigate three types of collection-based features about query, feedback documents and candidate terms, for adaptive setting of the balance parameter.

4.1 Information Amount of Query

Intuitively, if a query contains a sufficient amount of information about the search topic, then the expansion terms may be less important and thus more weight

should be given to the original query. As the query performance is largely related to the information amount of the query, it is natural to borrow some features that have been used in query performance prediction [5]. As a step further, we also propose to look at two extra features, namely the mutual information and information entropy.

4.1.1. The distribution of information amount in the query terms

In general, each query term t can be associated with an inverse document frequency ($idf(t)$) describing the information amount that the term carries. According to Pirkola and Järvelin [13], the difference in the discriminative power of query terms, which is reflected by the $idf(t)$ values, could affect the retrieval effectiveness. Therefore, the distribution of the $idf(t)$ over query terms, denoted DI, might be an intrinsic feature that affects the selection of balance parameter. DI is represented as:

$$DI = \sigma_{idf} \tag{3}$$

where σ_{idf} is the standard deviation of the idf values of the terms in Q. In our study, idf is defined as follows:

$$idf(t) = \frac{\log \frac{(N+0.5)}{N_t}}{\log(N+1)} \tag{4}$$

where N_t is the number of documents containing the query term t, and N is the number of documents in the collection. The higher DI score, the more dispersive the query's information amount distribution is. Then we would need to bring in more precise information from the expansion terms, and thus give more weight to the feedback/expansion model.

4.1.2. Query scope

The notion of query scope characterizes the generality of a query. For example, the query "Chinese food" is more general than "Chinese dumplings", as the latter is about a particular Chinese food. The query scope was originally studied in [14], defined as a decay function of the number of documents containing at least one query term, and has been shown to be an important property of the query. Similarly, in this paper, we define the query scope as follows:

$$QS = -\log(\frac{n_Q}{N}) \tag{5}$$

where n_Q is the number of documents containing at least one of the query terms, and N is the number of documents in the whole collection. A larger n_Q value will result in a lower query scope. The higher QS value means clearer information contained by the query, then we should give more weight to the original query.

4.1.3. Average inverse collection term frequency

According to Kwok [8], the inverse collection term frequency ($ICTF$) can be seen as an alternative of idf and is correlated with the quality of a query term. The average $ICTF$ ($AvICTF$) is given by:

$$AvICTF = \frac{log\Pi_{q \in Q} \frac{|C|}{tf_q}}{|Q|} \tag{6}$$

where tf_q is the occurrence frequency of a query term in the collection; $|C|$ is the number of tokens in the collection; and $|Q|$ is the query length. $AvICTF$ measures the overall discriminative power of query terms. The higher $AvICTF$ of the query indicates that more weight may be needed for the original query while the expansion terms may not bring much extra benefit.

4.1.4. Mutual information among query terms

Mutual information (MI) [12] is used to quantify how the terms in a query are associated to each other. The MI is a quantity that measures the mutual dependence of the two discrete random variables X and Y, defined as follows:

$$MI(Q) = I(X;Y) = \sum_{y \in Y} \sum_{x \in X} P(x,y) \log \frac{P(x,y)}{P(x)P(y)} \tag{7}$$

where $P(x,y)$ is the joint probability distribution function of X and Y, $P(x)$ and $P(y)$ are the marginal probability distribution functions of X and Y respectively. In our study, they can be defined as follows:

$$\begin{array}{l} P(x,y) = df_{xy}/N \\ P(x) = df_x/N \\ P(y) = df_y/N \end{array} \tag{8}$$

where x and y are two original query terms; df_{xy} is the document frequency where terms x and y co-occur; N is the number of documents in the whole collection; df_x and df_y are document frequency of the query term x and y respectively. The higher MI score means a high correlation among query terms, and thus more coherent information is carried by the original query. In turn, less weight can be given to candidate expansion terms.

4.1.5. Information entropy of query

We propose to analyze the term distribution in a query using information entropy [2]. In information theory, entropy measures the average amount of information contained in a message received, thus characterizing the uncertainty of information. For a random variable X with n outcomes $\{x_1, ..., x_n\}$, the widely used Shannon entropy (denoted by $H(X)$), is defined as follows:

$$IE(Q) = H(x) = -\sum_{i=1}^{n} P(x_i) \log P(x_i) \tag{9}$$

where $P(x_i)$ is the probability mass function of outcome x_i. In this study, it is calculated as follows:

$$P(x_i) = tf_i/Ntf \tag{10}$$

where tf_i is the frequency of query term x_i, Ntf is the sum of the all tfs in the collection. The high IE score means less certainty of the query, then more weight should be given to candidate expansion terms.

4.2 Reliability of feedback documents

We expect that the more reliable the feedback documents are, the more weight should be given to the expansion model derived from these documents.

4.2.1. Clarity of feedback documents

The clarity of feedback documents is defined as follows, as also used in [10].

$$CFD = \sum_{\omega \in F} p(\omega|\theta_F) \log \frac{p(\omega|\theta_F)}{p(\omega|C)} \tag{11}$$

where F is the set of feedback documents, $p(\omega|C)$ is the collection language model, and $p(\omega|\theta_F)$ is estimated as $p(\omega|\theta_F) = \frac{c(\omega,F)}{\sum_\omega c(\omega,F)}$. The higher CFD value, the more reliable the feedback documents tend to be.

4.2.2. The content of least frequent terms in feedback documents

The least frequent terms (LFT) are terms appearing less than a certain number of times (e.g., 3 in our experiments) in the collection and containing non-alphabetical characters, such as "00", "1", "2d". These terms usually have little practical significance. The content of LFT in feedback documents is defined as:

$$LFT_F = \frac{N(LFT)}{|F|} \tag{12}$$

where $N(LFT)$ is the number of LFT terms in the feedback documents, and $|F|$ is the total number of terms in the feedback documents. The higher LFT_F, the less reliable the feedback documents tend to be.

4.3 Reliability of Candidate Terms

We expect that the higher reliability of candidate expansion terms, the more weight should be given to them when combining with the original query model.

4.3.1. Mutual information between candidate expansion terms and query

The definition of $MI(C)$ is the same as $MI(Q)$ roughly, except the different meaning of the variables in Equations (7) and (8). For $MI(Q)$, the X and Y represent the original query terms, but for $MI(C)$, they represent the original query and candidate terms respectively.

4.3.2. Information entropy of candidate expansion terms

Similar to the definition of $IE(Q)$, the $IE(C)$ can be calculated using Equations (9) and (10), with x_i representing candidate terms.

4.3.3. The content of LFT in candidate expansion terms

This can be measured in the same way as for the feedback documents in Equation (12), and we defined it as $LFT_C = \frac{N(LFT)}{|C|}$. For candidate expansion terms, $N(LFT)$ is the number of LFT terms in the candidate terms, and $|C|$ is the total number of candidate terms.

5 The Logistic Regression Model

Logistic regression is widely used in data mining and machine learning. We use a logistic regression model to combine our features and generate a score for predicting the balance parameter, whereas the output is confined to values between 0 and 1. The method is the same as the one used in [10], defined as follows:

$$f(X) = \frac{1}{1 + exp(-X)} \tag{13}$$

where the variable $X = \overline{\omega} * \overline{x}$ represents the set of features. Specifically, $\overline{x}$ is a vector of numeric values representing the features and $\overline{\omega}$ represents a set of weights, which indicates the relative weights for each features. $f(X)$ represents the probability of a particular outcome given the set of features.

6 Experiments and Results

6.1 Experimental Setup

We used five standard benchmarking collections in our experiments: AP8890 (AP), WSJ8792 (WSJ), ROBUST2004 (ROBUST), WT10G and SJM, which are different in size and genre. The WSJ, AP and SJM collections are relatively small and consist of news articles, science and technology reports and government documents, whereas WT10G is a larger Web collection. The details of these collections are shown in Table 1.

Table 1. Information of Collections

collection	Size	#doc(K)	Queries	#qry	Avg_dl	Dev(dl)
SJM	286 MB	90K	101-150	46	218	364
AP	728 MB	243K	151-200	50	244	244
WSJ	508 MB	173K	151-200	50	247	455
ROBUST	1.85 GB	528K	601-700	99	254	869
WT10G	10.2 GB	1692K	501-550	50	379	2941

In all the experiments, we only used the title field of the TREC queries for retrieval, because it is closer to the actual queries used in the real web search applications and relevance feedback is expected to be the most useful for short queries [19].

First, we used Indri which is part of the Lemur Toolkit [11] to index document collection. In the indexing process, all terms were stemmed using Porter's English stemmer [15], and stopwords from the standard InQuery stoplist [1] were removed. Then, we initially retrieved a document list for each query using language model with the Dirichlet prior (takes a hyper-parameter of μ applied

to smooth the document language model which is better than other smoothing methods for title query.) and fixed the smoothing parameter to 1500 for all queries. This is our baseline for all pseudo relevance models in our experiments denoted as LM. After that, for each query, we used the expanded query model (Eq.(1)) to get the candidate expansion terms. In this part, we fixed the number of feedback documents to top 30, and the number of candidate expansion terms to 100 according to the settings in existing work.

To train the proposed adaptive relevance feedback, we needed to obtain the training data first. Considering the reliability and authority of training data, 90% of the queries were selected randomly for training, resulting in a total of 41 out of queries 101-150, 45 out of queries 151-200, 89 out of queries 601-700, and 45 out of query 501-550, and the rest were taken as testing queries. In this way, we aimed to make the training data more diversified and the test results more general and reliable. It turned out that 262 queries were taken as training data of different types and 33 queries were taken as testing data.

For traditional RM3 model, as we have discussed in Section 3, the balance parameter λ (Eq.(2)) changed from 0 to 1 (0.1, 0.2,....., 1) on five collections to find the optimal λ (this parameter is fixed for all queries in the same collection) and we called it RM3-Manual. For our adaptive relevance feedback, we chosen the optimal λ for each query. All the above processes were the same for our training and testing data.

Finally, the effectiveness of the IR models on each collection was measured by the Mean Average Precision (MAP) [7] at the top 1000 retrieved documents.

6.2 Sensitivity of Balance Parameter

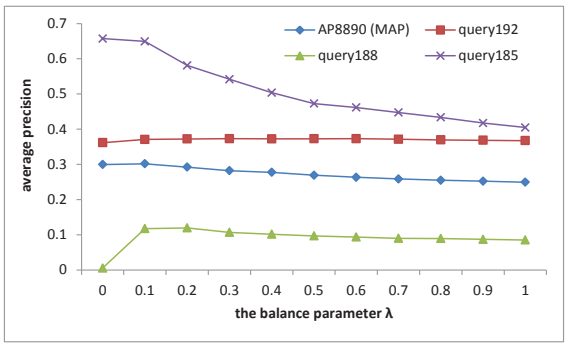

Fig. 1. Sensitivity of the balance parameter (λ) for different queries on AP8890 collection

We investigated the sensitivity of balance parameter on AP8890 collection and some queries of AP8890 in relevance feedback experiments by varying λ form 0 to 1, as it is showed in Figure 1. We could observe that the setting of λ

could affect the retrieval performance significantly, and the optimal parameter for different queries on the same collection could be quite different.

6.3 Correlation between features and the optimal balance parameter

We measured the correlation between features and the optimal balance parameter for each query in the training data using Pearson and Spearman methods which are common. Based on the section 4, we could obtain a matrix[262,10] of query-features, each query had its own 10 feature values and optimal λ which were the base of analysis. As showing in Table 2, DI, QS, $IE(Q)$ and LFT_F are more correlated with the optimal feedback coefficient than other features. It may mean that the information of query plays an important role in predicting the balance parameter.

Table 2. Pearson and Spearman correlation coefficients between features and the optimal λ on training data

Features	Pearson	Spearman
DI	**-0.0453**	**-0.1495**
QS	**-0.133**	**-0.1022**
AvICTF	0.0531	0.0892
MI(Q)	0.0028	-0.0262
IE(Q)	**-0.0832**	**-0.1347**
CFD	0.0751	0.05
LFT_F	**0.1158**	**0.1037**
MI(C)	-0.0509	-0.0702
IE(C)	-0.0662	-0.0559
LFT_C	-0.0189	-0.0178

6.4 Prediction models and the Results

In this part, we trained three prediction models on our training data by using three different sets of features respectively and the assessment of fit is based on significance tests for the balance parameter.: (1) all of the proposed features (ten); (2) only DI, QS, $IE(Q)$ and LFT_F; (3) five important features proposed in [10], including clarity of queries, feedback length, clarity of feedback documents and the absolute divergence between queries and feedback documents (see [10] for more details). They are called as "RM3-A", "RM3-A2" and "RM3-B" respectively. Given a new query, we could predict its feedback balance parameter directly using the formula:$f(X) = \frac{1}{1+exp(-X)}$ which was introduced in section 5, and X for all ten features ($X1$) and four important features ($X2$) are showed

below:

$$X1 = -0.2444 - 3.6127 * DI - 0.4249 * QS - 0.0214 * AvICTF \\ +184.2403 * MI(Q) - 44.8057 * IE(Q) + 3.1588 * CFD \\ +0.6834 * LFT_F - 0.7406 * MI(C) + 0.5376 * IE(C) - 4.7051 * LFT_C \quad (14)$$

$$X2 = -0.5594 - 5.5303 * DI - 0.3347 * QS - 43.2822 * IE(Q) + 0.418 * LFT_F \quad (15)$$

From the above formula, we can see that, for the distribution of information amount in queries, DI and $IE(Q)$ are correlated negatively to the λ and $MI(Q)$ shows a positive correlation. This is consistent with our expectation that more weight should be given to the original query when the query has more information. For the reliability of feedback documents and expansion terms, LFT_F, $EI(C)$ and LFT_C should be positively to the λ, while CFD and $MI(C)$ should show a negative correlation. That means high weight $(1 - \lambda)$ should be given to candidate terms when the feedback information is more reliable. However, the different behaviors of LFT_F and LFT_C in Eq.(14) could be explained as a trade-off between "credibility" and "quantity of information".

Table 3. Performance comparison of RM3-A and RM3-B on all testing data.

	SJM	AP8890	WSJ8792	ROBUST	WT10G
LM	0.2461	0.3279	0.3321	0.2258	0.2840
RM3-A	**0.3105**	0.3249	**0.3360**	**0.2597**	**0.3061**
RM3-B	0.3081	0.3309	0.3357	0.2495	0.3009
RM3-A2	0.3104	**0.3363**	0.3342	0.2485	0.2999
RM3-Manual	0.3175	0.3361	0.3293	0.2555	0.3044

The performance (MAP) on baseline, RM3-A, RM3-A2, RM3-B and RM3-Manual are demonstrated in Table 3.It shows that RM3-A, RM3-A2 and RM3-B all outperform the LM, but comparing with RM3-Manual, there is still room improvement by further optimizing the feedback parameter. RM3-A is better than RM3-B on SJM, WSJ8792, ROBUST2004 and WT10G, and only fails on AP8890. RM3-A2 is better than RM3-B on AP8890. The result is encouraging, our method which explicitly takes into account the collection-based features and the multiple types of our training data make the result more robust when predicting for different type of collections. As for RM3-A and RM3-A2, the results indicate that the performance of using all features is better than some important features in generally.

Further more, we show the performance of baseline, RM3-A, RM3-B, RM3-A2 and RM3-Manual on ROBUST2004 for each testing query in table 4. The results show that our method (RM3-A) is really effectiveness for per query when comparing with RM3-B.

Table 4. Performance comparison of RM3-A and RM3-B on ROBUST2004.

ROBUST2004	LM	RM3-A	RM3-B	RM3-A2	RM3-Manual
query609	0.0280	**0.0404**	0.0308	0.0359	0.0443
query621	0.0812	0.0634	0.0868	0.0671	0.0634
query635	0.5471	**0.6152**	0.6067	0.6168	0.6026
query642	0.3503	**0.4014**	0.4014	0.3882	0.4014
query651	0.0220	**0.0786**	0.0175	0.0186	0.0164
query666	0.7081	**0.6815**	0.6727	0.6815	0.6935
query678	0.1509	**0.1961**	0.1933	0.1933	0.213
query683	0.0976	**0.2677**	0.2227	0.2227	0.2677
query691	0.0125	**0.0144**	0.0138	0.0144	0.0139
query700	0.2600	0.2384	0.2503	0.2469	0.2384

7 Conclusions and Future Work

In this paper, we propose a series of collection-based features about query, feedback documents and candidate expansion terms, then combine them using a logistic regression model to adapt the balance parameter of PRF for different queries and collections (RM3-A). The experiments show that our method outperforms a state-of-art method (RM3-B) when the training and test data are of very different types. This verifies our hypothesis on incorporating collection-sensitive features will help improve the retrieval performance. On the other hand, there is still a room for further improvement when comparing with the manual setting of optimal balancing parameter. We will keep improving our work in the future by investigating other features about collection and analyzing the relationship between different features. We will also evaluate our method on different PRF methods and using different training and test data.

Acknowledgments. This work is supported in part by Chinese National Program on Key Basic Research Project (973 Program, grant No.2013CB329304, 2014CB744604), the Chinese 863 Program (grant No. 2015AA015403), the Natural Science Foundation of China (grant No. 61272265, 61402324), and the Research Fund for the Doctoral Program of Higher Education of China (grant No. 20130032120044).

References

1. J. Allan, M. E. Connell, W. B. Croft, F.-F. Feng, D. Fisher, and X. Li. Inquery and trec-9. Technical report, DTIC Document, 2000.
2. I. Bia00ynicki-Birula and J. Mycielski. Uncertainty relations for information entropy in wave mechanics. *Communications in Mathematical Physics*, 44(2):129–132, 1975.
3. C. Buckley and S. Robertson. Relevance feedback track overview: Trec 2008. In *In Proceedings of TREC 2008*, 2008.

4. G. Cao, J.-Y. Nie, J. Gao, and S. Robertson. Selecting good expansion terms for pseudo-relevance feedback. In *SIGIR*, pages 243–250. ACM, 2008.
5. B. He and I. Ounis. Query performance prediction. *Information Systems*, 31(7):585–594, 2006.
6. K. S. Jones. Experiments in relevance weighting of search terms. *Information Processing & Management*, 15(79):133144, 1979.
7. K. Kishida. Property of mean average precision as performance measure in retrieval experiment. *Ipsj Sig Notes*, 2001:97–104, 2001.
8. K.-L. Kwok. A new method of weighting query terms for ad-hoc retrieval. In *Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 187–195. ACM, 1996.
9. V. Lavrenko and W. B. Croft. Relevance based language models. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 120–127. ACM, 2001.
10. Y. Lv and C. Zhai. Adaptive relevance feedback in information retrieval. In *Proceedings of the 18th ACM conference on Information and knowledge management*, pages 255–264. ACM, 2009.
11. P. Ogilvie and J. P. Callan. Experiments using the lemur toolkit. In *TREC*, volume 10, pages 103–108, 2001.
12. H. Peng, F. Long, and C. Ding. Feature selection based on mutual information: Criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 27(8):1226–1238, 2005.
13. A. Pirkola and K. Järvelin. Employing the resolution power of search keys. *Journal of the American Society for Information Science and Technology*, 52(7):575–583, 2001.
14. V. Plachouras, I. Ounis, C. J. van Rijsbergen, and F. Cacheda. University of glasgow at the web track: Dynamic application of hyperlink analysis using the query scope. In *TREC*, volume 3, pages 636–642, 2003.
15. M. F. Porter. An algorithm for suffix stripping. *Program*, 14(3):130–137, 1980.
16. G. Salton. Improving retrieval performance by relevance feedback. *Journal of the American Society for Information Science*, 41(4):288–297, 1990.
17. M. Sanderson, A. Turpin, Y. Zhang, and F. Scholer. Differences in effectiveness across sub-collections. *Cikm Acm Conference on Information & Knowledge Management*, pages 1965–1969, 2012.
18. J. Xu and W. B. Croft. Query expansion using local and global document analysis. In *Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 4–11. ACM, 1996.
19. Z. Ye and J. X. Huang. A simple term frequency transformation model for effective pseudo relevance feedback. In *Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval*, pages 323–332. ACM, 2014.
20. C. Zhai and J. Lafferty. Model-based feedback in the language modeling approach to information retrieval. *Proceedings of Tenth International Conference on Information & Knowledge Management*, pages 403–410, 2001.
21. C. Zhai and J. Lafferty. A study of smoothing methods for language models applied to ad hoc information retrieval. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 334–342. ACM, 2001.
22. P. Zhang, D. Song, X. Zhao, and Y. Hou. A study of document weight smoothness in pseudo relevance feedback. In *Information Retrieval Technology*, pages 527–538. Springer, 2010.