Automatically Building a Stopword List for an Information Retrieval System

Rachel Tsz-Wai Lo, Ben He, Iadh Ounis
Department of Computing Science
University of Glasgow
17 Lilybank Gardens
Glasgow, UK
lotr|ben|ounis@dcs.gla.ac.uk

ABSTRACT: *Words in a document that are frequently occurring but meaningless in terms of Information Retrieval (IR) are called stopwords. It is repeatedly claimed that stopwords do not contribute towards the context or information of the documents and they should be removed during indexing as well as before querying by an IR system. However, the use of a single fixed stopword list across different document collections could be detrimental to the retrieval effectiveness. This paper presents different methods in deriving a stopword list automatically for a given collection and evaluates the results using four different standard TREC collections. In particular, a new approach, called term-based random sampling, is introduced based on the Kullback-Leibler divergence measure. This approach determines how informative a term is and hence enables us to derive a stopword list automatically. This new approach is then compared to various classical approaches based on Zipf's law, which we used as our baselines here. Results show that the stopword lists derived by the methods inspired by Zipf's law are reliable but very expensive to carry out. On the other hand, the computational effort taken to derive the stopword lists using the new approach was minimal compared to the baseline approaches, while achieving a comparable performance. Finally, we show that a more effective stopword list can be obtained by merging the classical stopword list with the stopword lists generated by either the baselines or the new proposed approach.*

Keywords: Information Retrieval, information theory, stopword
Reviewed and accepted 15 Dec. 2004

1. INTRODUCTION

Two related facts were noticed in the early days of information retrieval by *Luhn* [13]. First of all, a relatively small number of words account for a very significant fraction of all text's size. Words like IT, AND and TO can be found in virtually every sentence in English-based documents. Secondly, these words make very poor index terms [3]. Users are indeed unlikely to ask for documents with these terms. Moreover, these words make up a large fraction of the text of most documents. According to Francis and Kucera [8], the ten most frequently occurring words in English typically account for 20 to 30 percent of the tokens in a document.

These words are said to have a very low discrimination value [16] when it comes to IR and they are known as stopwords or sometimes as noise words or the negative dictionary. In other words, the amount of information carried by these words is negligible. Consequently, it is usually worthwhile to ignore all stopword terms when indexing the documents and processing the queries.

By analysing the Brown corpus, Fox [7] derived a list of stopwords which contains all the obvious stopwords like THE, FOR, IS, AND, IT etc. This stopword list is referred to as the *classical stopword list* in this paper. However, each collection of documents is unique. It is therefore sensible to automatically fashion a different stopword list for different collections in order to maximise the performance of an IR system. Moreover, it is possible that the pattern of word occurrences has changed over the last 20 years, especially in the context of the Web. Therefore, the current classical stopword list might need to be updated.

In this paper, we introduce a new approach, called term-based random sampling, inspired by the query expansion technique [19]. Using the *Kullback-Leibler* divergence measure [5], the new approach determines the amount of information a word contains [15]. The less information a word has, the more likely it is going to be a stopword. We evaluate our new term-based random sampling approach using various TREC collections. In addition, we compare the new term-based random sampling approach to four approaches inspired by Zipf's law [20], which are used here as our baselines.

The remainder of this paper is organised as follows: Section 2 introduces the baseline approaches. Section 3 discusses the new proposed approach, term-based sampling approach, in details. Section 4 gives the experimental setup and the evaluation approach. Section 5 discusses and analyses the results. Finally, Section 6 concludes the work and provides some possible future work, based on the findings of our experiments.

2. OUR BASELINE APPROACHES

George Kingsley Zipf (1902-1950) observed that the term's *rank-frequency* distribution can be fitted very closely by the relation:

$$F(r) = \frac{C}{r^{\alpha}} \qquad (1)$$

where a ~ 1 and *C* is ~ 0.1. The above equation (1) is known as *Zipf's law* [20]. Our four baseline approaches are inspired by *Zipf*'s law. Each collection is indexed, stemmed but no tokens are removed. This allows us to determine the best possible stopword list for a given collection. We use the general method, given in Table 1, to derive a stopword list based on *Zipf*'s law.

Using the algorithm in Table 1, by substituting 'term frequency' with one of the following four refinements, different sets of stopword lists can be computed. Four refinements are used for this experiment because we do not know for certain which refinement would produce a better set of stopwords.

- Term frequency (TF) of the terms in the corpus: In other words, the number of times a certain term appears throughout a specific collection.
- Normalised term frequency: Normalising the term frequency (TF) by the total number of tokens in the collection (i.e. the size of the lexicon file). The calculation is straightforward and can be achieved using the following formula:

$$TF_{Norm} = -log\left(\frac{TF}{v}\right) \qquad (2)$$

where *TF* is the term frequency of a particular term and *v* is

the total number of tokens in the lexicon file.

- Inverse Document Frequency (IDF) [12]: Using the term frequency distribution in the collection itself where the IDF value of a given term k is given by:

$$idf_k = log\left(\frac{NDoc}{D_k}\right) \qquad (3)$$

where *NDoc* is the total number of documents in the corpus and *Dk* is the number of documents containing term *k*.

In other words, infrequently occurring terms have a greater probability of occurring in relevant documents and should be considered as more informative and therefore of more importance in these documents.

- Normalised IDF: The most common form of IDF weighting is the one used by *Robertson* and *Sparck-Jones* [14], which normalises with respect to the number of documents not containing the term *(NDoc — Dk)* and adds a constant of 0.5 to both numerator and denominator to moderate extreme values:

$$idf_{k\ Norm} = log\left(\frac{(NDoc - D_k) + 0.5}{D_k + 0.5}\right) \qquad (4)$$

where *NDoc* is the total number of documents in the collection and *Dk* is the number of documents containing term *k*.

As shown in Table 1, a threshold needs to be determined. The aim is to find a threshold which allows generating a set of stopword list that would produce the best average precision. Choosing the threshold at random is not appropriate and is, therefore, not recommended. To choose a threshold, it is essential to investigate the frequencies difference between two consecutive ranks, namely *F(r)* and *F(r* + 1). This is because if the difference between *F(r)* and *F(r* + 1) is very high i.e. the term with frequency *F(r)* occurs more often than the term with frequency *F(r* + 1), we could then choose that value as a threshold. In other words, any terms with frequencies ≥ *F(r)* could be used as a stopword.

The choice of the most appropriate threshold for a given collection is very important. If one too many words is considered to be a stopword, then there is a possibility that a relatively informative word has been omitted from the retrieval process, resulting in a lower retrieval effectiveness. On the other hand, if not enough words are considered to be stopwords, large amount of documents that are less specific to a given query would be retrieved and the average precision can also decrease. Thus, for a given collection, the general method of Table 1 requires finding the optimal associated threshold. In Section 4, we show how the thresholds could be set.

3. THE TERM-BASED RANDOM SAMPLING APPROACH

We introduce the term-based random sampling approach, which is based on how informative a particular term is. We could determine whether a specific term is a stopword based on its importance i.e. the less important a term is, the more likely it is a stopword. The importance of a term can be assessed using the *Kullback-Leibler divergence measure* [5].

Intuitively, the proposed approach is similar to the idea of query expansion [19] in IR in which we expand the given query based on a particular query term. The idea behind query expansion is to find terms that complement the initially chosen query term (or terms) motivated by the realisation that one term per concept might sometimes be inadequate to express a concept accurately - so we add terms (original query term(s) plus expanded terms). Our approach is based on query expansion with one difference. Instead of finding terms that have the same or similar meaning to a given term, we find all the documents containing this term and we use these documents as our sample. Then, we extract the least informative terms from the sample by measuring the divergence

Generate a list of term frequencies vs terms based on the corpus.
Sort the term frequencies listings in descending order, i.e. the terms with the highest term frequencies will be at the top.
Rank the terms according to their term frequencies. The one with the highest term frequencies would have rank = 1 and the next most frequent term would have rank = 2, etc.
Draw a graph of term frequencies vs ranks. This should obey Zipf's Law.
Choose a threshold and any words that appear above the particular threshold are treated as stopwords.
Run the querying with the above said stopword list, all the stopwords in the queries would be removed.
Evaluate the system after querying and note down the average precision.

Table 1 Algorithm for baseline approaches

of a given term distribution within the sampled document set from its distribution in the collection background. We then use the *Kullback-Leibler* (KL) divergence measure in order to determine each term's importance.

Similar to the baseline approaches, once stemmed, all tokens are kept in the lexicon file. This would once again enable us to deduce all possible stopwords for a given collection. Using the KL measure, the weight of a term *t* in the sampled document set is given by:

$$w(t) = P_x \cdot log_2 \frac{P_x}{P_c} \qquad (5)$$

In the above formula, $P_x = tfx/lx$ and $P_c = \frac{F}{token_c}$ where tf_x is the frequency of the query term in the sampled document set, l_x is the sum of the length of the sampled document set, *F* is the term frequency of the query term in the whole collection and $token_c$ is the total number of tokens in the whole collection. The steps for the term-based random sampling approach is given in Table 2.

Selecting a random term in the lexicon file has the possibility of finding only one document containing that term which would result in a relatively small sample. This problem can be overcome by repeating the selection step *Y* times. Theoretically, if we run the selection step repeatedly, a better sample can be achieved, creating a better overview of the terms distribution and their importance. Note that this approach is simpler to implement than the baseline approaches, introduced in Section 2, even though the algorithm looks quite complex. This is because everything can be done automatically and therefore is less expensive to carry out. For the baseline approaches, we need to go through the *F(r) — F(r* + 1) listing one by one, and we also need to fit the *tf/idf* vs rank graph to *Zipf*'s law. Furthermore, there is no need to monitor the process unlike the baseline approaches, where the thresholds are required to be selected by going through the *F(r) — F(r + 1)* listing.

The term-based random sampling approach is an alternate approach in determining stopwords. This approach is based on how informative a term is and since stopwords are terms that contain no information, this approach could enable us to determine a different set of stopwords, compared to the baseline approaches. Moreover, the term-based random sampling approach selects its first term randomly, hence enables us to have a better coverage of the whole collection. Since we have no idea what term it would select as its first term, for all we know, it could be a pre-defined stopword or it could be a term that is truly informative, for example, a term that occurs very infrequently in the collection.

4. EXPERIMENTAL SETUP

We experiment with four TREC collections[1], as shown in Table 3. Each TREC collection comes with a set of queries (See Table 4).

[1]See http://trec.nist.gov/data/doc_eng.html for further information on the TREC collections

Each query consists of three fields, namely Title, Description and Narrative. For all reported experiments, we used all three fields. The reason is that this would maximise our chances of using the generated stopwords, and hence enable us to assess the effectiveness of both the baselines and the proposed approaches.

Terrier[2] retrieval platform was used to conduct all of our experiments. Terrier is based on a divergence from randomness (DFR) framework for deriving parameter-free probabilistic models for IR. The PL2 model was used for all of the experiments. PL2 is one of the DFR document weighting models [1]. Using the PL2 model, the relevance score of a document *d* for query *Q* is given by:

Repeat Y times, where Y is a parameter:

Randomly choose a term in the lexicon file, we shall call it ω_{random}

Retrieve all the documents in the corpus that contains ω_{random}

Use the refined Kullback-Leibler divergence measure to assign a weight to every term in the retrieved documents. The assigned weight will give us some indication of how important the term is.

Divide each term's weight by the maximum weight of all terms. As a result, all the weights are controlled within [0,1]. In other words, normalise each weighted term by the maximum weight.

Rank the weighted terms by their associated weight in ascending order. Since the less informative a term is, the less useful a term is and hence, the more likely it is a stopword.

Extract the top X top-ranked (i.e. least weighted), where X is a parameter.

You now have an array of length $X * Y$. Each element in the array is associated to a weight.

Shrink the array by merging the elements containing the same term and take the average of the term's associated weights. For example, if the term *"retrieval"* appears three times in the array and its weights are 0.5, 0.4 and 0.3 respectively, we merge these three elements together into one single one and the weight of the term *"retrieval"* will become

$$\frac{(0.5 + 0.4 + 0.3)}{3} = 0.4$$

Rank the shrunk array in increasing order depending on the term's weight. In other words, sort the array in ascending order.

Extract the L top-ranked terms as stopword list for the collection. L is a parameter. Therefore, it is often a good idea to use trial and error.

Table 2 Algorithm for Term-based sampling approach

Collection	Size	Number of documents	Number of terms in the lexicon	c value
disk45 [17]	2GB	528155	801397	2.13
WT2G [10]	2GB	247491	1020277	2.75
WT10G [9]	10GB	1692096	3206346	2.43
DOTGOV [18]	18GB uncompressed	1247753	2821821	2.00

Table 3 Collections used for the experiment

Collection	Query Sets	Number of queries
disk45	TREC7 and TREC8 of ad-hoc tasks	100
WT2G	TREC8	50
WT10G	TREC10	50
DOTGOV	TREC11 and TREC12 merged	100

Table 4 Query sets used for the experiment

[2]Terrier is developed by the Information Retrieval Group at the University of Glasgow. Further information about Terrier can be found at: http://ir.dcs.gla.a.uk/terrier/

$$score(d,Q) = \sum_{t \in Q} w(t,d)$$
$$= \sum_{t \in Q} \frac{1}{tfn+1}\big(tfn \cdot \log_2 \frac{tfn}{\lambda} + (\lambda + \frac{1}{12 \cdot tfn} - tfn) \cdot \log_2 e + 0.5 \cdot \log_2(2\pi \cdot tfn)\big) \quad (6)$$

where λ is the mean and variance of a Poisson distribution. *w(t,d)* is the weight of document *d* for query term *t.*

The normalised term frequency *tfn* is given by the *normalisation 2 [2]:*

$$tfn = tf \cdot \log_2(1 + c \cdot \frac{avg\_l}{l}), (c > 0) \quad (7)$$

where *l* is the document length and *avg_J* is the average document length in the whole collection, *tf* is the original term frequency, *c* is the free parameter of the normalisation method, which can be different for different collections and is automatically estimated [11]. The c value used for each collection can be found in Table 3.

The control factor for this experiment is Fox*'s* classical stopword list. Using both the baselines and the term-based sampling approach, we compared the derived stopword list to the classical stopword list, using average precision. For the baseline approaches, we varied the threshold values, hoping to find one particular set of stopword list that would produce a better average precision than the classical stopword list. Over 50 different sets of stopword lists were generated for each of the variants.

Since the term-based random sampling approach takes into account how informative a term is, the proposed algorithm is able to automatically formulate a list of stopwords that are likely to increase the average precision. The parameter *X* was set to 200, *Y* was set to 1000 while *L* was set to 400. Empirically, this setting was shown to achieve the optimal results, while the computational effort was kept within a reasonable limit.

Finally, we produced another new stopword list by merging Fox's classical stopword list with the best stopword list generated using either the baselines or the term-based random sampling approach. The merging process consists of adding the newly defined stopwords to the existing classical stopword list, removing any duplicates in order to ensure each term will occur once and once only in the new merged stopword list. The idea is that this merged stopword list might even be stronger in terms of effectiveness. This idea of merging follows directly from a classical IR technique of combining evidences, as discussed in [6]. In the next section, we compare the term-based random sampling approach to our baselines, and evaluate the effectiveness of the merged stopword list.

5. RESULTS AND ANALYSIS

In the following sections, we present the findings of our experiments. Section *5.1* discusses the obtained results using the variant baseline approaches. Section *5.2* presents the results using the new proposed approach. Finally, we present a further method for generating a more effective stopword list in Section 5.3. All of the statistical testing were done using the Wilcoxon Matched-Pairs Signed-Ranks Test.

5.1 Baseline Approaches - Overall Results

We generated over 50 stopword lists for each of the variants described in Section 2. The best obtained results are shown in Table 5. As indicated in this table, *normalised IDF* is the best variant of the approaches inspired by *Zipf*'s law. This was expected, since IDF is more reliable than TF alone and normalised IDF is a refinement to IDF.

Table 5 also shows that in the case of the WT2G collection, the results obtained are the best, with the average precision improving by about 5% using normalised IDF. This is perhaps due to the fact that this collection is the smallest out of the four that we have used and the number of tokens in this collection is relatively small compared to the other collections. Results obtained using WT10G collection were also good. In WT10G, using the normalised IDF method results in an improvement of 4% average precision. The average precision for the DOTGOV and disk45 collection did very marginally improve using normalised IDF. Finally, it is worth mentioning some of the newly generated stopwords found using the baseline approaches for each collection (See Table 6). Notice that the terms *'http', 'html', 'web'* appear in every Web collection. As a result, we have successfully updated the stopword list with these terms.

5.2 Term-Based Random Sampling Approach- Overall Results

The overall results obtained using the proposed term-based random sampling approach can be seen in Table 7. The results obtained were not as positive as the baseline approaches nor as positive as we have hoped, especially with the DOTGOV and the WT10G collections. However, the computational effort is reduced significantly when compared to the baseline approaches. After investigation, we noticed that very few of the newly generated stopwords (the ones that were not already contained in the classical stopword list) occurred in the queries. However, as we will show in Section 5.3.2, if we merge the new generated stopword list with the classical stopword list, we can produce a more efficient stopword list for the system. Selected terms generated using the new proposed approach can be found in Table 8.

5.3 Refinement - Merging

We have shown that the classical stopword list was indeed very effective on its own, despite its old age. Moreover, most terms in the newly defined stopword lists were also in the classical one. Therefore, one can assume that if we merge the two lists together, then this should provide us with an even more effective stopword list for a given collection. In Section 5.3.1 we merge the classical stopword list with the stopword lists produced using normalised IDF while in Section 5.3.2, we merge the classical stopword list with the best stopword list produced using the new approach, the term-

Collection	Classical	TF	Normalised TF	IDF	Normalised IDF	p-value
disk45	0.2123	0.2130	0.2123	0.2113	**0.2130**	0.8845
WT2G	0.2569	0.2650	0.2676	0.2682	**0.2700**	0.001508*
WT10G	0.2000	0.2049	0.2076	0.2079	**0.2079**	0.1231
DOTGOV	0.1223	0.1212	0.1208	0.1227	**0.1227**	0.55255

Table 5 Best average precision produced by each method for different collection, using baseline approaches, where Classical is Fox's stopword list. * indicates a significant difference between the stopword lists generated by normalised IDF and Fox's classical stopword list at 0.05 level.

disk45	WT2G	WT10G	DOTGOV
financial	html	able	content
company	http	copyright	gov
president	htm	ok	define
people	internet	http	year
market	web	html	administrate
london	today	january	http
national	policy	history	web
structure	content	facil	economic
january	document	html	year

Table 6 Selected terms derived using the baseline approaches for each collection used

Collection	Classical	Best Obtained	p-value
disk45	**0.2123**	**0.2129**	0.868
WT2G	**0.2569**	**0.2668**	0.07544
WT10G	**0.2000**	**0.1900**	0.4493
DOTGOV	**0.1223**	**0.1180**	0.002555*

Table 7: Average Precision for best results produced using term-based random sampling approach, where Classical is Fox's stopword list and Best Obtained is the best obtained stopword list using the parameters mentioned in Section 4. * indicates a significant difference between the stopword lists generated by the new proposed term-based random sampling approach and Fox's classical stopword list at 0.05 level.

disk45	WT2G	WT10G	DOTGOV
column	advance	copyright	server
general	beach	friend	modify
califonia	company	memory	length
industry	environment	mountain	content
month	garden	problem	accept
director	industry	science	inform
desk	material	special	connect
economic	pollution	internet	gov
business	school	document	byte

Table 8 Selected terms generated for each collection using the new proposed term-based random sampling approach

based random sampling approach. As mentioned in Section 4, the merged stopword list was produced by adding all the terms from the newly defined stopword lists onto the classical stopword list and then removing all the duplicate terms so that each term appears once in the newly merged stopword list.

5.3.1 Baseline Approaches

A shown in Table 9, by merging the best obtained stopword lists with the classical stopword list, we have managed to provide the system with an even more effective set of stopwords, hence improving the average precision. In particular, we have managed to improve the system by almost 6% in the WT2G collection, 5.45% for the WT10G collection, 1.5% for the DOTGOV collection, respectively.

5.3.2 Term-Based Random Sampling Approach

Table 10 shows that by merging the best obtained stopword list with the classical stopword list, we also obtain some reasonable improvement. Comparing the results obtained using both the normalised IDF baseline and the new proposed approach, the results were more desirable when using our baseline approaches. Looking at the TREC queries, we noticed that these queries were relatively conservative and as a result, not all the stopwords found using the new proposed approach were in the queries. However, the computation effort required to obtain the stopword list using the proposed approach was minimal compared to our baseline approaches, suggesting that it is more easily used in an operational setting. Its balance between cost and effectiveness seems to be reasonably good compared to the baselines.

6. CONCLUSIONS AND FUTURE WORK

In this paper, we proposed a new method for automatically generating a stopword list for a given collection. The approach, called term-based random sampling approach, is based on how informative a given term is. We investigated the effectiveness and the robustness of this new approach using various standard collections. The new approach was compared to four variant baselines approaches inspired by *Zipf*'s law. The results show that the proposed novel approach achieves a comparable performance to the baseline approaches, while having a lower computational overhead. In addition, using the proposed approach, the optimal threshold setting is easier to obtain. Moreover, the experimental results demonstrate that a more effective stopword list could be derived by merging Fox's classical stopword list with the stopword list produced by either the baselines or the proposed approach.

The KL divergence measure has been used to assess the informative amount of a given term in this paper. We plan to investigate other metrics such as the *Poisson*-based approach in [15], where documents are assumed to be independent rather than disjoint, or the divergence from randomness approaches proposed in [1]. Another area that requires further investigation is whether a verb or a noun are equally informative [4]. For example, consider the phrase '*I can open a can of tuna with a can opener',* there are three oc- currences of the term *'can'.* However, the first *'can'* is a verb whereas the latter two are nouns. Should we remove all the frequently occurring nouns and verbs as stopwords, the precision of the IR system would decrease considerably. The ideal Information Retrieval system stopword list would be able to take into accounts both verb and noun and their appropriate usage during indexing a collection.

Collection	Classical	Normalised IDF	Merged Stopword List	p-value
disk45	0.2123	0.2130	**0.2130**	0.8845
WT2G	0.2569	0.2700	**0.2712**	0.000746*
WT10G	0.2000	0.2079	**0.2109**	0.03854*
DOTGOV	0.1223	0.1227	**0.1241**	0.6775

Table 9 Average Precision for Merged Stopword List using baseline approaches, where Classical is Fox's stopword list* indicates a significant difference between the stopword lists generated by the merged stopword list using normalised IDF and Fox's classical stopword list at 0.05 level.

Collection	Classical	Best Obtained	Merged Stopword List	p-value
disk45	0.2123	0.2129	**0.2129**	0.868
WT2G	0.2569	0.2668	**0.2703**	0.008547*
WT10G	0.2000	0.1900	**0.2066**	0.4451
DOTGOV	0.1223	0.1180	**0.1228**	0.5085

Table 10 Average Precision for Merged Stopword List using term-based random sampling approach, where Classical is Fox's stopword list and Best Obtained is the best obtained stopword list using the parameters mentioned in Section 4. * indicates a significant difference between the stopword lists generated by the merged stopword list using term-based random sampling approach and Fox's classical stopword list at 0.05 level.

7. ACKNOWLEDGEMENTS

This work is funded by a UK Engineering and Physical Sciences Research Council (EPSRC) project grant, number GR/R90543/01. The project funds the development of the Terrier IR framework (http://ir.dcs.gla.ac.uk/terrier).

References

[1] G. Amati (2003). *Probabilistic Models for Information Retrieval based on Divergence from Randomness.* PhD thesis, Department of Computing Science, University of Glasgow.

[2] G. Amati, C. J. van Rijsbergen (2002) . Probabilistic models of information retrieval based on measuring the divergence from randomness. *ACM Transactions on Information Systems (TOIS),* 20(4) 357-389.

[3] R. K. Belew (2000). *Finding Out About. Cambridge*: University Press.

[4] S. Charkrabarti (2003). *Mining the Web: Discovering knowledge from hypertext.* Morgan Kaufmann.

[5] T. M. Cover, J. A. Thomas (1991). *Elements of Information Theory.* John Wiley.

[6] W. B. Croft (2000). Combining approaches to information retrieval. *In: Advances in Information Retrieval: Recent Research from the Center for Intelligent Information,* p. 1-28. Kluwer

Academic Publishers.

[7] C. Fox (1992). Lexical analysis and stoplists. *In: Information Retrieval - Data Structures & Algorithms,* p. 102-130. Prentice-Hall.

[8] W. Francis (1982). *Frequency Analysis of English Usage: Lexicon and Grammar.* Houghton Mifflin.

[9] D. Hawking (2000). Overview of the TREC2002. *In: Proceedings of the Nineth Text REtrieval Conference (TREC 9),* p. 87-94, Gaithersburg, MD.

[10] D. Hawking, E. Voorhees, N. Craswell, P. Bailey (1999). Overview of the TREC-8 Web Track. In *Proceedings of the Eighth Text REtrieval Conference (TBEC-8),* p. 131-150, Gaithersburg, MD.

[11] B. He, I. Ounis (2003). A study of parameter tuning for term frequency normalization. *In: Proceedings of the Twelfth ACM CIKM International Conference on Information and Knowledge Management,* p. 10-16, New Orleans, LA, USA.

[12] K. Sparck-Jones (1972). A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation,* 28(1) 11-21.

[13] H. P. Luhn (1957). A statistical approach to mechanized encoding and searching of literary information. *IBM Journal of Research and Development,* 1(4) 309-317.

[14] S. E. Robertson, K. S. Jones (1976). Relevance weighting of search terms. *Journal of the American Society for Information Science,* 27(3)129-146.

[15] T. Roelleke (2003). A frequency-based and a Poisson-based definition of the probability of being informative. In *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval,* p. 227-234, Toronto, Canada.

[16] C. J. van Rijsbergen (1979). *Information Retrieval, 2nd edition.* Butterworth-Heinemann.

[17] E. Voorhees, D. Harman (1999). Overview of the Eighth Text REtrieval Conference (TREC-8). In: *Proceedings of the Eighth Text REtrieval Conference (TREC-8),* p. 1-23, Gaithersburg, MD,.

[18] E. M. Voorhees (2002). Overview of TREC2002. *In*: *Proceedings of the Eleventh Text REtrieval Conference (TREC2002),* p. 1-16, Gaithersburg, MD.

[19] J. Xu, W. B. Croft (1996). 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,* p. 4-11, Zurich, Switzerland.

[20] H. Zipf (1949). *Human Behaviours and the Principle of Least Effort.* Addison-Wesley, Cambridge, MA.