Query Reformulation in Collaborative Information Retrieval

Armin Hust, Stefan Klink, Markus Junker, Andreas Dengel
German Research Center for Artificial Intelligence (DFKI GmbH)
P.O. Box 2080, 67608 Kaiserslautern, Germany
{Armin.Hust, Stefan.Klink, Markus.Junker, Andreas.Dengel}@dfki.de

ABSTRACT
Information retrieval (IR) systems utilize user feedback for generating optimal queries with respect to a particular information need. However the methods that have been developed in IR for generating these queries do not memorize information gathered from previous search processes, and hence can not use such information in new search processes. Thus each new search process does not know anything about previous search processes and can not profit from the results of the previous processes. We call systems which can consider results from previous search processes Collaborative Information Retrieval (CIR) systems. Improving retrieval quality in a CIR system should be possible, since the system can learn from many queries issued from various users. In this paper we present a new method for use in CIR. We are proposing to use previously learned queries and their relevant documents for improving overall retrieval quality. Based on the similarity of a new query to previously learned queries we are expanding the new query by extracting terms from documents which have been judged as relevant to these previously learned queries. Thus our method uses global feedback information for query expansion in contrast to local feedback information which has been used in previous work in query expansion methods.

KEY WORDS
Collaborative Information Retrieval, Query Expansion, Text Mining, Machine Learning

1 Introduction

Gathering information for fulfilling the information need of a user is an expensive operation in terms of time required and resources used. Queries may have to be reformulated manually by the user or automatically by the IR system several times until the user is satisfied. The same expensive operation has to be carried out, if another user has the same information need and thus initiates the same or a similar search process.
How users can improve the original query formulation by means of relevance feedback is an ongoing research activity in IR [1]. In our approach we are using global relevance feedback which has been learned from previous queries instead of local relevance feedback which is produced during execution of a individual query.
The motivation for our query expansion method is straightforward, especially in an environment where document collections are static:

- If documents are relevant to a query which has been issued previously by a user, then the same documents are relevant to the same query at a later time, when that query is re-issued by the same or by a different user. This is the trivial case, where similarities between the two different queries is the highest.
- In the non-trivial case a new query is similar to a previously issued query only to a certain degree. Then our assumption is that documents which are relevant to the previously issued query will be relevant to the new query only to a certain degree.

In this work we do not consider learning methods for user relevance feedback, instead we expect that relevance judgements are available for use. An IR system should be able to maintain information about previous search processes as well as information about relevance judgements (directly specified or derived from users actions). Then in processes called Collaborative Information Retrieval (CIR) the system may improve overall retrieval quality for all users, benefitting from previous search processes issued by different users.

2 Document Retrieval

The task of document retrieval is to retrieve documents relevant to a given query from a fixed set of documents. Documents as well as queries are represented in a common way using a set of index terms (called terms from now on). Terms are determined from words of the documents, usually during preprocessing phases where some noise reduction procedures are incorporated (e.g. stemming and stopword elimination). In the following a term is represented by $t_i,\ 1 \leq i \leq M$, where M is the number of terms in the document collection.

2.1 Vector Space Model

One of the simplest but most popular models used in IR is the vector space model (VSM) [1], [2]. A document in the VSM is represented as a M dimensional vector

$$d_j = (w_{1j}, w_{2j}, ..., w_{Mj})^T,\ \ 1 \leq j \leq N, \qquad (1)$$

where T indicates the transpose of the vector, w_{ij} is the weight of term t_i in document d_j and N is the number of

documents in the document collection. A query in the VSM is also represented as a M dimensional vector

$$q_k = (w_{1k}, w_{2k}, ..., w_{Mk})^T, \ \ 1 \leq k \leq L, \qquad (2)$$

where w_{ik} is the weight of term t_i in query q_k and L is the number of queries contained in the document collection.
The result of the execution of a query is a list of documents ranked according to their similarity to the given query. The similarity $sim(d_j, q_k)$ between a document d_j and a query q_k is measured by the cosine of the angle between these two M dimensional vectors:

$$sim(d_j, q_k) = \frac{d_j^T \cdot q_k}{\parallel d_j \parallel \cdot \parallel q_k \parallel}, \qquad (3)$$

where $\parallel \cdot \parallel$ is the Euclidean norm of a vector. In the case that the vectors are already normalized (and hence have a unit length) the similarity is simply the dot product between the two vectors d_j and q_k:

$$sim(d_j, q_k) = d_j^T \cdot q_k = \sum_{i=1}^{M} w_{ij} \cdot w_{ik} \qquad (4)$$

The VSM is one of the models we are using in this work for comparison to our new query expansion method.

2.2 Query Expansion

Usage of short queries in IR produces a shortcoming in the number of documents ranked according to their similarity to the query. The number of ranked documents is related to the number of appropriate query terms. The more query terms, the more documents are retrieved and ranked according to their similarity to the query [3]. Several methods, called query expansion methods, have been proposed to cope with this problem [1], [2]. These methods fall into three categories: usage of feedback information from the user, usage of information derived locally from the set of initially retrieved documents, and usage of information derived globally from the document collection.

2.2.1 Pseudo Relevance Feedback

The method called *pseudo feedback* (also called *pseudo relevance feedback*, PRF) works in three stages: First documents are ranked according to their similarity to the original query. Then highly ranked documents are assumed to be relevant and their terms (all of them or some highly weighted terms) are used for expanding the original query. Then documents are ranked again according to the similarity to the expanded query.
In this work we employ a simple variant of pseudo relevance feedback [4]. Let E be the set of document vectors given by

$$E = \left\{ d_j \middle| \frac{sim(d_j, q)}{\max_{1 \leq i \leq N}\{sim(d_i, q)\}} \geq \theta \right\} \qquad (5)$$

where q is the original query and θ is a threshold of the similarity. Then the sum D_q of the document vectors in E

$$D_q = \sum_{d_j \in E} d_j \qquad (6)$$

is used as expansion terms for the original query. The expanded query vector q' is obtained by

$$q' = q + \alpha \frac{D_q}{\parallel D_q \parallel}, \qquad (7)$$

where α is a parameter for weighting the expansion terms. Then the documents are ranked again according to their similarity $sim(d_j, q')$.
PRF is one of the models we are using in this work for comparison to our new query expansion method.

3 Query Linear Combination and Relevant Documents

In this paper we employ a query expansion method based on linear combinations of similar queries and their relevant documents (QLD). Our method uses feedback information and information globally available from previous queries. Feedback information in our experimental environment is available in the ground truth data provided by the test document collections. The ground truth provides relevance information, i.e. for each query there exists a list of relevant documents.

3.1 Query Expansion Method

We will first give a high level description of the method, then the detailed mathematical description is given. Query expansion works as follows:

- compute the similarities between the new query and each of the existing old queries
- select the old queries having a similarity to the new query which is greater than or equal to a given threshold
- use these selected old queries to rebuild the new query approximately as a linear combination of the old queries according to the least squares method and get the coefficients of the linear combination
- select these old queries, where the coefficient in the linear combination for the new query is greater than or equal to a given threshold
- from these selected old queries get the sets of relevant documents from the ground truth data
- from each set of relevant documents compute a new document vector
- use these document vectors and a weighting scheme to enrich the new query, where the coefficients in the linear combination for the new query are used as weighting parameters

The formal description is given here. The similarity $sim(q_k, q)$ between a query q_k and a new query q is measured by the cosine of the angle between these two M dimensional vectors:

$$sim(q_k, q) = \frac{q_k^T \cdot q}{\parallel q_k \parallel \cdot \parallel q \parallel}, \qquad (8)$$

or simply using the dot product between the two vectors q_k and q:

$$sim(q_k, q) = q_k^T \cdot q = \sum_{i=1}^{M} w_{ik} \cdot w_i \quad (9)$$

in the case that the vectors are already normalized (and hence have a unit length).

Let S be the set

$$S = \{q_k | sim(q_k, q) \geq \sigma,\ 1 \leq k \leq L\} \quad (10)$$

of existing old queries q_k having a similarity greater than or equal to a threshold σ to the new query q, let $|S|$ be the number of queries in S, and let T_k be the sets

$$T_k = \{d_j | q_k \in S \wedge d_j \text{ is relevant to } q_k\} \quad (11)$$

of all documents relevant to the queries q_k in S. Then the sums D_k of the document vectors in each T_k

$$D_k = \sum_{d_j \in T_k} d_j \quad (12)$$

are used as expansion terms for the original query.

The expanded query vector q' is obtained by

$$q' = q + \sum_{k=1}^{|S|} \tilde{\lambda_k} \frac{D_k}{\| D_k \|}, \quad (13)$$

where the $\tilde{\lambda_k}$ are parameter for weighting the expansion terms.

The $\tilde{\lambda_k}$ are computed as follows: in most cases we cannot represent the new query q exactly as a linear combination of the old queries $q_k \in S$, i.e.

$$q = \sum_{k=1}^{|S|} \lambda_k q_k, \quad q_k \in S. \quad (14)$$

will not have a solution for the coefficients λ_k. Equation 14 is equivalent to a system of linear equations

$$Q\lambda = q \quad (15)$$

where $Q \in \mathbb{R}^{M \times |S|} = (q_1, q_2, \ldots, q_{|S|})$ is a matrix of M rows and $|S|$ columns and $\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_{|S|})^T$ is a column vector consisting of $|S|$ elements.

In our case Q is normally singular ($M \gg |S|$) and there is no solution to the system. In this situation we find a vector $\hat{\lambda}$ so that it provides a closest fit to the equation in some sense. Our approach is to minimize the Euclidean norm of the vector $Q\lambda - q$, i.e we solve

$$\hat{\lambda} = argmin_\lambda \|Q\lambda - q\| \quad (16)$$

where $\hat{\lambda} = (\hat{\lambda_1}, \hat{\lambda_2}, \ldots, \hat{\lambda_{|S|}})^T$ is called the least squares solution for the system $Q\lambda = q$.

Then from our set of $|S|$ coefficients $\hat{\lambda_k}$ we select only these coefficients $\tilde{\lambda_k}$ which are greater than or equal to a predefined parameter β, i.e.

$$\tilde{\lambda_k} = \begin{cases} \hat{\lambda_k} & \text{if } |\hat{\lambda_k}| \geq \beta \\ 0 & \text{if } |\hat{\lambda_k}| < \beta \end{cases} \quad (17)$$

Notes:

- if σ in (10) is chosen to high the set S may be empty. Then the sets T_k will be empty and the document vectors D_k will be $(0, ..., 0)^T$. In this case the new query will not be expanded.
- even if a query q_k is in the set S, the corresponding set T_k may be empty (in case where no relevance judgements are contained in the ground truth data for query q_k). Then the corresponding document vector D_k will be $(0, ..., 0)^T$.
- parameters σ in (10) and β in (17) are the tuning parameters for method QLD.

4 Experimental Design

In this section we describe the test collections used in the experimental comparison. We use standard document test collections and standard queries and questions provided by [5] and [6]. On the one hand by utilizing these collections we take advantage of the ground truth data for performance evaluation. On the other hand we do not expect to have queries having highly correlated similarities as we would expect in a real world application. So it is a challenging task to show performance improvements for our method.
In our experiments we used the following eight collections:

- the CACM (titles and abstracts from the journal 'Communications of the ACM'), CISI (Institute of Scientific Information) and CRAN (aeronautics abstracts) collections are available at [5]. All collections are provided with queries and their ground truth.
- the CR (congressional record) collection. The CR collection is contained in the TREC test collections disk 4 [6], accompanied by the ground truth for 34 selected queries out of the TREC standard queries 251 - 300. We created three test cases for the CR collection, using the TREC queries of different length in order to investigate the influence of query length. The "CR-title" contains the "title" queries (the shortest query representation), the "CR-desc" contains the "description" queries (the medium length query representation), the "CR-narr" contains the "narrative" queries (the longest query representation).
- the FR (federal register) collection. The FR collection is contained in the TREC test collections disk 2, accompanied by the ground truth for 112 selected queries out of the TREC standard queries 51 - 300.
- the AP90 (associated press articles) collection contained in the TREC test collections disk 3. Originally the AP90 collection contains 78321 documents. From the TREC-9 Question Answering track (QA) we selected the question set 201-893. Questions 201-700 were created without reference to the document set. Then in a separate pass equivalent but re-worded questions (questions 701-893) were created from a subset of these 500 questions [7]. Because of the method used for the construction of this set (especially questions 701 - 893) we expected to get higher similarities between different questions. From the ground truth

data provided with the QA-track we selected only those questions having a relevant answer document in the AP90 document collection. Thus we reduced our test data to 723 documents and 353 questions.

Terms used for document and query representation were obtained by stemming and eliminating stopwords. Statistics about these collections before stemming and stopword elimination can be found in [2] and [4].

The best known term weighting schemes use weights according to the so-called *tf–idf* schemes. In our experiments we employ a standard scheme as follows: For document vectors the weights w_{ij} are calculated as

$$w_{ij} = tf_{ij} \cdot idf_i, \tag{18}$$

where tf_{ij} is a weight computed from the raw frequency f_{ij} of a term t_i (the number of occurrences of term t_i in document d_j)

$$tf_{ij} = \sqrt{f_{ij}}, \tag{19}$$

and idf_i is the inverse document frequency of term t_i given by

$$idf_i = \log \frac{N}{n_i}, \tag{20}$$

where n_i is the number of documents containing term t_i. For query vectors the weights w_{ik} are calculated as

$$w_{ik} = \sqrt{f_{ik}}, \tag{21}$$

where f_{ik} is the raw frequency of a term t_i in a query q_k (the number of occurrences of term t_i in query q_k).

5 Experimental Results

In this section the results of the experiments are presented. Results were evaluated using the average precision over all queries. Recall/precision graphs were generated according to [2]. Then significance tests were applied to the results.

5.1 Results

The methods VSM (vector space model), PRF (pseudo relevance feedback) and QLD (query linear combination and relevant documents) were applied. Best parameter value settings for parameters α and θ for method PRF have been obtained by experiment. Parameters α and θ are chosen such that average precision is highest (see table 1 and [4]).

Method QLD has been evaluated using different settings for parameters σ and β. From the set of queries contained in each collection we selected each query one after the other and treated it as a new query $q_l, 1 \leq l \leq L$. Then for each fixed query q_l we computed the similarity $\sigma_k := sim(q_k, q_l)$ for all queries $q_k, 1 \leq k \leq L, k \neq l$ according to equations (8) and (9). Then σ in equation (10) has been varied from 0.0 up to 1.0 in steps of 0.01, and β in equation (17) has been varied from 0.0 up to the maximum value $\hat{\lambda_k}$ computed by equation (16) in steps of 0.01.

Table 1. Best parameter values for methods PRF and QLD

		CACM	CISI	CRAN	CR-desc
PRF	α	1.7	0.7	1.3	0.5
	θ	0.35	0.7	0.9	0.85
QLD	σ	0.22	0.25	0.37	0.36
	β	0.16	0.23	0.41	0.37

		CR-narr	CR-title	FR	AP90
PRF	α	0.4	0.6	0.6	0.2
	θ	0.95	0.75	0.55	0.75
QLD	σ	0.17	0.41	0.37	0.68
	β	0.20	0.43	0.13	0.48

Finally we got our new query vector according to equation (13) and issued the query. Best parameters values for σ and β are reported in table 1.

Table 2. Average precision obtained in different methods

	CACM	CISI	CRAN	CR-desc
VSM	0.130	0.120	0.384	0.175
PRF	0.199	0.129	0.435	*0.204*
QLD	0.227	*0.171*	0.436	0.175
QLDPRF	*0.273*	**0.173**	*0.453*	*0.204*
PRFQLD	**0.275**	0.169	**0.470**	**0.208**

	CR-narr	CR-title	FR	AP90
VSM	0.173	0.135	0.085	0.745
PRF	*0.192*	0.169	0.113	0.757
QLD	0.175	0.164	0.108	0.812
QLDPRF	*0.192*	*0.184*	**0.161**	**0.815**
PRFQLD	**0.193**	**0.190**	*0.144*	*0.814*

In the next steps we combined two methods of query expansion in this ways: First, after having expanded the new query using the QLD method, we applied the PRF method against the expanded query. This method is reported as the QLDPRF method. Second, after having expanded the new query using the PRF method, we applied the QLD method against the expanded query. This method is reported as the PRFQLD method. Best parameter value settings have again been obtained by experiment and are chosen such that average precision is highest.

Table 2 shows the average precision obtained by using the best parameter values for different methods. For each collection the best value of average precision is indicated by bold font, the second best value is indicated by italic font.

Figures 1 and 2 show the recall/precision graphs for two of the test collections. Each figure contains the graphs for methods VSM, PRF, QLD, QLDPRF and PRFQLD.

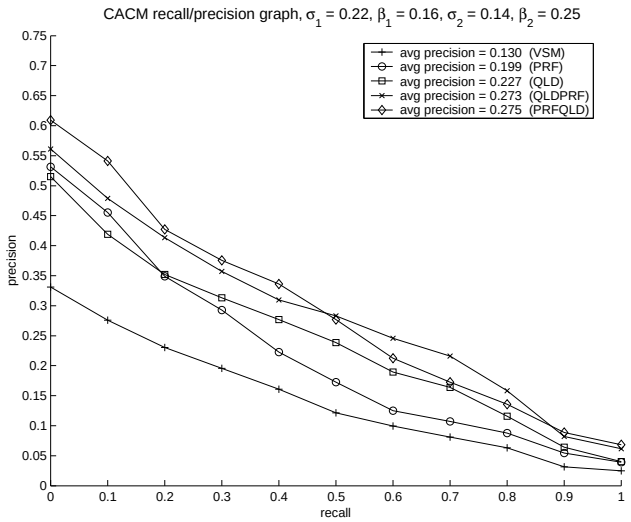

Figure 1. Recall/precision graphs for the CACM collection. σ_1 and β_1 are tuning parameter for the QLD method, σ_2 and β_2 denote tuning parameter for the PRFQLD method.

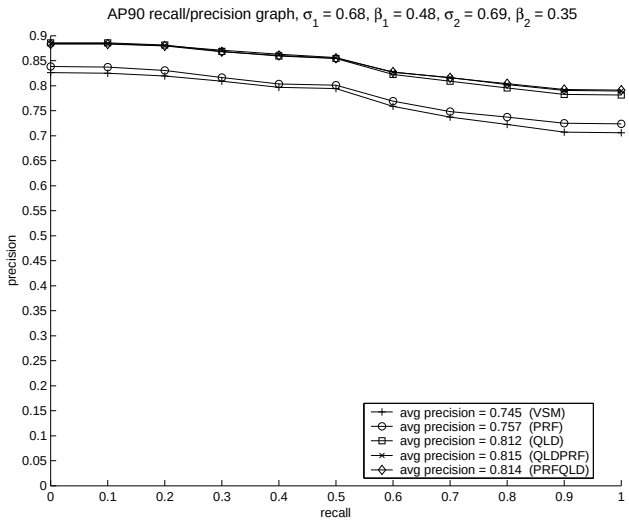

Figure 2. Recall/precision graphs for the AP90 collection.

5.2 Significance Testing

The next step for the evaluation is the comparison of the values of the average precision obtained by different methods. Statistical tests provide information about wether observed differences in different methods are really significant or just by chance. Several statistical tests have been used in IR [8], [9]. We employ the "paired t-test" described in [8]: Let x_l and y_l be the scores of retrieval methods X and Y for query $q_l, 1 \leq l \leq L$ and define $d_l = x_l - y_l$. The assumption is that the model is additive, i.e. $d_l = \mu + \varepsilon_l$, where μ is the mean value and the errors ε_l are independent and normally distributed. The null hypothesis H_0 is that $\mu = 0$, i.e. method X performs as well as method Y. The alternative hypothesis H_1 is that $\mu > 0$, i.e. method X performs better than method Y in terms of average precision.

The t-test

$$t = \frac{\overline{d}}{s}\sqrt{n}, \tag{22}$$

where

$$\overline{d} = \frac{1}{n}\sum_{i=1}^{n} d_i, \quad s^2 = \frac{1}{n-1}\sum_{i=1}^{n}(d_i - \overline{d})^2 \tag{23}$$

follows the t-distribution with a degree of freedom of $n-1$, where n is the number of samples, $\overline{d}$ is the sample mean and s^2 is the sample variance.

Given the value of t we obtain the p-value, i.e. the probability of observing the sample results d_l under the assumption that H_0 is true. Comparing the p-value to a given significance level α, we can decide whether the null hypothesis H_0 should be rejected or not.

The results are shown in table 3. Each row contains the results of two tests, i.e. test method X against method Y and vice versa. We tested each method X against each other method Y and vice versa. We used significance levels $\alpha = 0.05$ and $\alpha = 0.01$. If there are differences showing up from different significance levels we indicated it in the table cells as follows:

- An entry of $++$ in a table cell indicates that the null hypothesis is rejected for testing X against Y at significance level $\alpha = 0.01$. This means that method X is almost guaranteed to perform better than method Y.
- An entry of $+$ in a table cell indicates that the null hypothesis is rejected for testing X against Y at significance level $\alpha = 0.05$, but can not be rejected at significance level $\alpha = 0.01$. This means that method X is likely to perform better than method Y.
- An entry of o in a table cell indicates that the null hypothesis can not be rejected in both test. This means that there is low probability that one of the methods is performing better than the other method.
- An entry of $-$ in a table cell indicates that the null hypothesis is rejected for testing Y against X at significance level $\alpha = 0.05$, but can not be rejected at significance level $\alpha = 0.01$. This means that method Y is likely to perform better than method X.
- An entry of $--$ in a table cell indicates that the null hypothesis is rejected for testing Y against X at significance level $\alpha = 0.01$. This means that method Y is almost guaranteed to perform better than method X.

From table 3 we can see that QLD performs better than VSM at the 0.05 (0.01) level in 5 (4) of 8 cases. Improvements of method QLD over PRF can be seen in 2 (2) of 8 cases, and PRF performs better than QLD in 1 (1) of 8 cases.

Additionally we can see that the combined method QLDPRF outperforms VSM in 8 (7) of 8 cases, and outperforms PRF in 4 (2) of 8 cases. Method PRFQLD out-

Table 3. Paired t-test results for $\alpha = 0.05$ and $\alpha = 0.01$

methods X	methods Y	CACM	CISI	CRAN	CR-desc
PRF	VSM	++	++	++	++
QLD	VSM	++	++	++	o
QLD	PRF	o	++	o	—
QLDPRF	VSM	++	++	++	++
QLDPRF	PRF	+	++	o	o
QLDPRF	QLD	++	+	++	++
PRFQLD	VSM	++	++	++	++
PRFQLD	PRF	++	++	++	o
PRFQLD	QLD	+	o	++	++
PRFQLD	QLDPRF	o	o	o	o

methods X	methods Y	CR-narr	CR-title	FR	AP90
PRF	VSM	+	+	+	+
QLD	VSM	o	o	+	++
QLD	PRF	o	o	o	++
QLDPRF	VSM	+	++	++	++
QLDPRF	PRF	o	o	+	++
QLDPRF	QLD	o	o	++	o
PRFQLD	VSM	+	++	++	++
PRFQLD	PRF	o	o	+	++
PRFQLD	QLD	o	o	++	o
PRFQLD	QLDPRF	o	o	o	o

performs VSM in 8 (7) of 8 cases, and outperforms PRF in 5 (4) of 8 cases.

Also in 5 (4) and 4 (3) of 8 cases the combined methods QLDPRF and PRFQLD outperform the QLD method.

Only for the AP90 collection, where QLD outperforms the VSM and PRF methods, the combined methods cannot outperform QLD. This seems to be a result of the construction of the queries in this collection (see section 4), where queries specifying the same information need in different wordings exist. These queries have a high similarity against each other.

6 Conclusions

We have experimentally compared a new query expansion method based on query similarities and document relevance with two conventional information retrieval methods. From the results gathered from eight static test collections we have only one clear indication that the QLD method is superior to the conventional PRF method. But in contrast we also have only one clear indication that the conventional PRF method is superior to the QLD method.
From our results we think that we can combine this new method with the conventional PRF method. No performance degradation has been observed for this combination of the two methods. The results that have been obtained by combining the new QLD method with the conventional PRF method are promising.
Due to the construction method for the queries in the AP90 test collection where QLD significantly performs better than the other methods we think that we could utilize this new method in cases

- where old queries and their corresponding relevance information has been learned previously and
- where new queries have high similarities to old existing queries.

7 Acknowledgements

This work was supported by the German Ministry for Education and Research, bmb+f (Grant: 01 IN 902 B8).

References

[1] C. Manning and H. Schütze. *Foundations of Natural Language Processing*. MIT Press, 1999.

[2] R. Baeza-Yates and B. Ribeiro-Neto. *Modern Information Retrieval*. Addison-Wesley Publishing Company, 1999.

[3] Y. Qiu and H.-P. Frei. Concept-based query expansion. In *Proceedings of SIGIR-93, 16th ACM International Conference on Research and Development in Information Retrieval*, pages 160–169, Pittsburgh, US, 1993.

[4] K. Kise, M. Junker, A. Dengel, and K. Matsumoto. Experimental evaluation of passage-based document retrieval. In *Proceedings of the Sixth International Conference on Document Analysis and Recognition (ICDAR'01)*, 2001.

[5] `ftp://ftp.cs.cornell.edu/pub/smart`.

[6] `http://trec.nist.gov`.

[7] E. M. Voorhees and D. Harman. Overview of the ninth text retrieval conference (trec-9). In *Proceedings of the Ninth Text Retrieval Conference (TREC-9)*, 2001.

[8] D. Hull. Using statistical testing in the evaluation of retrieval experiments. In *Proceedings of SIGIR-93*, pages 329–338, 1993.

[9] Y. Yang and X. Liu. A re-examination of text categorization methods. In *Proceedings of SIGIR-99*, pages 42–49, 1999.