Vector Space Basis Change in Information Retrieval

Rabeb Mbarek[1], Mohamed Tmar[1], and Hawete Hattab[2]

[1] Multimedia Information Systems and Advanced Computing Laboratory,
High Institute of Computer Science and Multimedia, University of Sfax, Sfax,
Tunisia

[2] Umm Al-Qura University, Makkah,
Saudi Arabia

rabeb.hattab@gmail.com, mohamedtmar@yahoo.fr, hattab.hawete@yahoo.fr

Abstract. The Vector Space Basis Change (VSBC) is an algebraic operator responsible for change of basis and it is parameterized by a transition matrix. If we change the vector space basis, then each vector component changes depending on this matrix. The strategy of VSBC has been shown to be effective in separating relevant documents and irrelevant ones. Recently, using this strategy, some feedback algorithms have been developed. To build a transition matrix some optimization methods have been used. In this paper, we propose to use a simple, convenient and direct method to build a transition matrix. Based on this method we develop a relevance feedback algorithm. Experimental results on a TREC collection show that our proposed method is effective and generally superior to known VSBC-based models. We also show that our proposed method gives a statistically significant improvement over these models.

Keywords. Vector space model, vector space basis change, VSBC-based model, relevance feedback.

1 Introduction

Information Retrieval (IR) is the activity of obtaining information resources relevant to an information need from a very large collection of documents [29, 2]. Most IR systems compute a numeric score which measures the relevance of an object with respect to a query, and rank the objects according to this value. Several IR models, including Vector Space Model (VSM) [29], probabilistic models [23] and language model[12], have been proposed to model this scoring function.

In the VSM, documents and queries are represented by vectors. Each component in a vector represents the weight of a term in the document and so the set of index terms (original vector space basis) generates documents and queries. The weight of the term depends on the well-known tf-idf weighting method.

The idea of Relevance Feedback (RF) is to take the results that are initially returned for a given query and to use information about whether or not these results are relevant to perform a new query. The most commonly used RF methods aim to rewrite the user query. In the VSM, RF is usually undertaken by re-weighting the query terms without any modification in the vector space basis. With respect to the initial vector space basis (index terms), relevant and irrelevant documents share some terms (at least the terms of the query which selected these documents). According to [15, 16], the technique of VSBC is effective in separating relevant documents and irrelevant ones.

The VSBC is an algebraic operator responsible for change of basis and it is parameterized by a transition matrix. By changing the basis, each vector component changes depending on this matrix. For example[1], let us consider a vector space basis $E = (e_1, e_2, e_3)$. Let v_1, v_2 and v_3 be three vectors as follows:

$$\begin{aligned} v_1 &= (1, 2, -1)_E ; \\ v_2 &= (2, 4, 1)_E ; \\ v_3 &= (2, 3, 1)_E . \end{aligned} \tag{1}$$

[1] The goal of this example is to show the impact of VSBC in similarity scores and also to help the reader to understand more about this strategy.

ISSN 1405-5546
DOI: 10.13053/CyS-18-3-2028

Equation 1 leads to the following:

$$\begin{aligned} v_1 =&\quad 1.e_1 + 2.e_2 - 1.e_3, \\ v_2 =&\quad 2.e_1 + 4.e_2 + 1.e_3, \\ v_3 =&\quad 2.e_1 + 3.e_2 + 1.e_3. \end{aligned}$$

Hence:

$$\begin{aligned} v_1 =&\quad (1, 2, -1)(e_1, e_2, e_3)^T, \\ v_2 =&\quad (2, 4, 1)(e_1, e_2, e_3)^T, \\ v_3 =&\quad (2, 3, 1)(e_1, e_2, e_3)^T. \end{aligned}$$

Now let us consider another vector space basis $F = (f_1, f_2, f_3)$ such as:

$$\begin{aligned} f_1 &= \big(0, 2, -2\big)_E; \\ f_2 &= \big(0, -1, 0\big)_E; \\ f_3 &= \big(-2, -3, -2\big)_E. \end{aligned} \tag{2}$$

Let v be a vector and $(\beta_1, \beta_2, \beta_3)$ its components with respect to the basis $E = (e_1, e_2, e_3)$. This vector has the new coordinates $(\lambda_1, \lambda_2, \lambda_3)$ with respect to the basis $F = (f_1, f_2, f_3)$. The relation between these components can be defined as follows:

$$\begin{aligned} v &= (\beta_1, \beta_2, \beta_3)(e_1, e_2, e_3)^T \\ &= (\lambda_1, \lambda_2, \lambda_3)(f_1, f_2, f_3)^T \\ &= (\lambda_1, \lambda_2, \lambda_3) \begin{pmatrix} 0 & 2 & -2 \\ 0 & -1 & 0 \\ -2 & -3 & -2 \end{pmatrix} (e_1, e_2, e_3)^T. \end{aligned}$$

Thus

$$(\lambda_1, \lambda_2, \lambda_3) \begin{pmatrix} 0 & 2 & -2 \\ 0 & -1 & 0 \\ -2 & -3 & -2 \end{pmatrix} = (\beta_1, \beta_2, \beta_3).$$

Let

$$M = \begin{pmatrix} 0 & 2 & -2 \\ 0 & -1 & 0 \\ -2 & -3 & -2 \end{pmatrix}.$$

Finally,

$$(\lambda_1, \lambda_2, \lambda_3) = (\beta_1, \beta_2, \beta_3)M^{-1}, \tag{3}$$

where

$$M^{-1} = \begin{pmatrix} 0.5 & 2.5 & -0.5 \\ 0 & -1 & 0 \\ -0.5 & -1 & 0 \end{pmatrix}.$$

The matrix M (resp. M^{-1}) is called the transition matrix from E to F (resp. from F to E).

The VSBC causes many vector behavior changes. Indeed, with respect to the basis E we have:

$$\cos(v_1, v_2) = \frac{v_1.v_2}{\|v_1\| \times \|v_2\|} = 0.8018,$$

$$\cos(v_1, v_3) = 0.7638,$$

which means that, with respect to the basis E, v_2 is more similar to v_1 than v_3.

Using Equation 3, v_1, v_2 and v_3 are rewritten in the basis F as:

$$v_1 = (1, 2, -1). \begin{pmatrix} 0.5 & 2.5 & -0.5 \\ 0 & -1 & 0 \\ -0.5 & -1 & 0 \end{pmatrix} = (1, 1.5, -0.5),$$

$$v_2 = (2, 4, 1). \begin{pmatrix} 0.5 & 2.5 & -0.5 \\ 0 & -1 & 0 \\ -0.5 & -1 & 0 \end{pmatrix} = (0.5, 0, -1),$$

$$v_3 = (2, 3, 1). \begin{pmatrix} 0.5 & 2.5 & -0.5 \\ 0 & -1 & 0 \\ -0.5 & -1 & 0 \end{pmatrix} = (0.5, 1, -1).$$

Thus

$$\cos(v_1, v_2) = 0.4781,$$

$$\cos(v_1, v_3) = 0.8909$$

We remark that contrary to E, with respect to the basis F, v_3 is more similar to v_1 than v_2.

We can conclude that the VSBC causes vector behavior changes.

The best framework that could bring the VSBC technique into application is RF: the user shows relevant and irrelevant documents in an initial ranking and instead of reformulating the query, we change the vector space basis in which it is written (as well as the documents). In [15, 16], Mbarek et al. built a basis which gathers the relevant documents, and the irrelevant ones are kept away from the relevant ones. These approaches were evaluated on a RF framework and on a Pseudo-Relevance Feedback (PRF) framework in [18]. Mbarek et al. used optimisation techniques to build a transition matrix (Supremum and Infimum of the function distance).

The main contribution of this paper is as follows. First, we propose to build a transition matrix using a simple, convenient and direct method based on an algebraic technique.

Second, we incorporate the VSBC into the classical Rocchio's agorithm and propose a new VSBC-based Rocchio model called VSBCRoc4. Finally, we compare our proposed model with all existing VSBC-based models. We show that our model has better performance over models of [8, 20, 15, 16, 18, 17] and the classic Rocchio's model combined with the $BM25$ baseline model; these improvements are statistically significant.

This paper is organized as follows. Section 2 presents the related work. Section 3 describes our algebraic VSBC-based approach. Section 4 shows the evaluation results obtained from a user study experiment. Conclusions and future work are presented in Section 5.

2 Related Work

In the VSM, a vector represents each document in a collection. Each component of a vector reflects a term associated with the document. The value assigned to that component reflects the importance of the term in representing the document. A query is also represented by a vector such that each component is the weight of a term.

A variety of models are available in the literature for weighting the document and query vector terms. A recent reconsideration of the geometry underlying IR, and indirectly of the VSM, was done in [33]. The VSM showed good feedback performance on most collections whereas the probabilistic model had problems with some collections [9].

2.1 Relevance Feedback

RF uses information provided by the user concerning whether or not the results that are initially returned for a given query are relevant to make a new query. The content of the selected documents is used to re-weight original terms and/or add new terms to the initial query [27]. The RF has been used in several IR models: the VSM [25], the probabilistic model [22, 5], the language model [6], and the bayesian network retrieval model [7]. RF is covered in several books (e.g., [14]) and surveys [26]. A dedicated track (i.e., the RF track) was run at TREC in 2008 and 2009. There exist two principal techniques of RF: a semi-automatic technique and an automatic technique.

The semi-automatic technique requires the intervention of the user who must identify and select the relevant and the irrelevant documents. The typical approach of this technique is called the Rocchio model [25], which is based on the VSM. The basic idea of this method is to add an average weight of each term within the set of relevant documents to the original query vector, and to subtract an average weight within the set of irrelevant ones from this vector. This hypothesis was followed by Ide in [10] who deduced from the formula of Rocchio a flexible one which enabled him not only to confirm the positive results obtained by Rocchio, but also to study three alternatives of this model [10].

Later, many works on the RF semi-automatic method were enriched by the contribution of the probabilistic model. This technique was first implemented by Croft and Harper [5]. The probabilistic model is based on the probability that a document is relevant to the user for a given query. This model is related to the RF because its parameters are estimated by the presence/absence of terms in relevant and irrelevant documents. De Compos et al. uses the Bayesian network retrieval model in [7]. The inference relations are represented by the term-document relations or the term-term ones. The RF is based on the distribution of messages among documents and terms to express the term relevance and irrelevance relations.

Due to the sensitivity to the quality of selected documents and terms, in some cases, the RF process does not operate satisfactorily. To improve the robustness of RF, several approaches have been proposed as follows. Sakai et al. [28] proposed to select only a subset of feedback documents instead of using all the documents. Cao et al. [4] suggested to select a subset of important terms instead of using all the terms obtained through feedback for query refinement. Tao and Zhai [32] proposed to change the importance of each feedback document. Xu et al. [34] and Zhou et al. [36] suggested to use a large external collection like Wikipedia or the web as a source of expansion terms beside those obtained through feedback process. Lv and Zhai [13] proposed a positional relevance model where the terms in the document which are nearer to the query terms are assigned more weight. Recently, Zhou et al. [37] proposed a novel approach to PRF inspired by collaborative filtering.

According to Salton [30], in the environments where the technique of the automatic RF is implemented, a number of documents extracted by the initial query are considered relevant. The procedures and formulas used in the approach of the automatic RF are alternatives of the formulas of Rocchio and Ide which make it possible to abstract irrelevant documents.

2.2 Vector Space Basis Change

Latent Semantic Indexing (LSI) [8] is a variant of VSM which maps a high dimensional space into a low dimensional one. LSI tries to take advantage of the conceptual content of documents. Instead of searching on individual terms, a search is performed on concepts. This technique is based on the Singular Value Decomposition (SVD) aiming at decomposing the term-document frequency matrix and disclosing the principal components used to represent fewer independent concepts than many inter-dependent index terms. This method results in a new vector space basis with a lower dimension than the original one (all index terms), and in which each component is a linear combination of the indexing terms. LSI is a VSBC-based model. It is stated in IR literature that LSI model is 30% more effective than the classical VSM. However, LSI yields poor retrieval accuracy vs the Okapi $BM25$ model on TREC collections [1]. Atreya and Elkan showed that $BM25 + LSI$ improves the performance (s_1^+ model) [1]. If A is the term-document matrix and A_k is the closest approximation to A among all matrices of rank k, then

$$s_1^+(Q_{int}) = Q_{int}^T[\frac{\lambda A_k}{\sqrt{diag(A_k^T A_k)}} + \frac{(1-\lambda)A}{\sqrt{diag(A^T A)}}], \tag{4}$$

where Q_{int} is the initial query.

According to Melucci [20], a context is modeled by a vector space basis and its evolution is modeled by a VSBC. Melucci developed a new context-based model called IRiX: if B is a basis which describes a context, $L(B)$ is the event that a vector belongs to the subspace spanned by B and P_B is a projector to this subspace, then the probability that a vector y is in the context described by B is

$$Pr[L(B)|L(\{y\})] = y^T.P_B.y, \tag{5}$$

where y^T is the transpose of the vector y. IRiX is a VSBC-based model. Recently, Mbarek et al. computed a context which gives the best ranking [17].

Recently, Mbarek et al. [15, 16] developed RF algorithms based on a VSBC. These RF algorithms improve the results of known models (BM25 model, Rocchio model). They build a transition matrix which gives a better representation of documents. This transition matrix should minimize the sum (S_1) of squared distances between each relevant document and g_R (g_R is the centroïd of relevant documents) and should maximize the sum (S_2) of squared distances between each irrelevant document and g_R. According to [15] (IBM1 model), this transition matrix should minimize the quotient

$$\frac{S_1 + \gamma}{S_2 + \gamma}, \tag{6}$$

where γ is a real parameter close to zero.

And according to [16] (IBM2 model), this transition matrix should maximize the difference

$$S_2 - S_1. \tag{7}$$

The main problem with Rocchio's approach [25] is that relevant and irrelevant documents overlap in the vector space because they often share the same terms (at least those of the query). Therefore, with respect to the original basis, it is difficult to select terms that separate relevant and irrelevant documents. To avoid this problem, Mbarek et al. [18] incorporated the VSBC into the classic Rocchio's model and proposed VSBC-based Rocchio's model, called VSBVRoc model. Let Q_{new} be the reformulated query and M be a transition matrix. For the VSBVRoc model, the reformulated query is

$$Q_{new} = Q_{int} + \beta.\frac{1}{|R|}\sum_{d \in R} M.d. \qquad (8)$$

If there is no VSBC (M is the unit matrix[2]), then we obtain the classical Rocchio's formula

$$Q_{new} = Q_{int} + \beta.\frac{1}{|R|}\sum_{d \in R} d. \qquad (9)$$

In [19], Melucci has showed that the classical Rocchio's algorithm is a VSBC-based model: there exists a matrix M such that Equation 9 is equivalent to $Q_{new} = M.Q_{int}$.

The specificity of our work consists of building a transition matrix using an algebraic method and comparing our proposed approach with all existing VSBC-based models.

3 Vector Space Basis Change based on an Algebraic Technique

Let M be a transition matrix from the original vector space basis (set of index terms) to a new basis B. If d is the vector of the document d with respect to the original basis, then $M.d$ is the vector of the same document d with respect to the basis B. With respect to the original vector space basis, relevant and irrelevant documents share some terms (at least the terms of the query which selected these documents). To avoid this problem, it suffices to generate each document by phrases. And so, this representation can optimally separate relevant and irrelevant documents. To model this approach, it suffices to remark that each phrase is a combination of index terms. Let us define the following matrix: each column is generated by a phrase, that is each column contains the combination coefficients of this phrase with respect to index terms. This matrix is the transition matrix from the original basis (index terms) to a basis composed by phrases.

3.1 Properties of the Transition Matrix

The transition matrix gives a new representation that keeps the relevant documents gathered to their centroïd and the irrelevant ones far from it. Each document d$_i$ is represented in a vector space by a vector $d_i = (w_{i1}, w_{i2}, ...w_{iN})$ where w_{ij} is the weight of the term t_j in the document d_i and N is the number of indexing terms. Note that our approach is independent of the term weighting method. The Euclidian distance between two documents d$_i$ and d$_j$ is given by

$$\begin{aligned} dist(d_i, d_j) &= \sqrt{\sum_{k=1}^{N}(w_{ik} - w_{jk})^2} \\ &= \sqrt{(d_i - d_j)^T.(d_i - d_j)}. \end{aligned} \qquad (10)$$

Let $d_i^* = M.d_i$ and $d_j^* = M.d_j$ be the vectors of the documents d$_i$ and d$_j$ respectively with respect to the new basis B. The distance between d_i^* and d_j^* is given by

$$\begin{aligned} dist(d_i^*, d_j^*) &= dist(M.d_i, M.d_j) \\ &= \sqrt{(d_i - d_j)^T.M^TM.(d_i - d_j)}. \end{aligned} \qquad (11)$$

The transition matrix M puts the relevant documents gathered to their centroïd g_R and the irrelevant documents far from it. g_R is done by

$$g_R = \frac{1}{|R|}\sum_{d \in R} d, \qquad (12)$$

where R is the set of relevant documents.

The transition matrix M should minimize the sum of squared distances between each relevant document d and g_R, i.e., the transition matrix M should contract the vector $d - g_R$ which implies that there exists a real parameter $0 < \alpha < 1$ such that

$$M(d - g_R) = \alpha(d - g_R). \qquad (13)$$

[2] A unit matrix of size n is the $n \times n$ square matrix with ones in the main diagonal and zeros elsewhere.

The transition matrix M also should maximize the sum of squared distances of each irrelevant document d and g_R, i.e., the transition matrix M should dilate the vector $d - g_R$, which implies that

$$M(d - g_R) = (1 + \alpha)(d - g_R). \tag{14}$$

3.2 Identification of the Transition Matrix

Let S be the set of irrelevant documents. The union $R \cup S$ is the initial set of ranked documents. Since there is no common documents between R and S, the union $R \cup S$ is a direct sum of R and S, i.e., a basis of $R \cup S$ is the union of a basis of R and a basis of S. Let $(e_1, ..., e_p)$ be a basis of R and $(e_{p+1}, ..., e_N)$ be a basis of S. $(e_1, ..., e_N)$ is a basis of the initial set of ranked documents.

According to Equations 13, 14 we have the following:

— If $d \in R$, then $d - g_R$ is an eigenvector of the matrix M associated to the eigenvalue α.

— If $d \in S$, then $d - g_R$ is an eigenvector of the matrix M associated to the eigenvalue $1 + \alpha$.

Then M is a diagonalized matrix (similar to a diagonal matrix) having two eigenvalues α and $1+\alpha$. Therefore

$$M = V.D.V^{-1}, \tag{15}$$

where D is a diagonal matrix formed by the eigenvalues[3] of M and the columns of V are the corresponding eigenvectors of M, i.e., the i-th column of V corresponds to the vector $e_i - g_R$.

3.3 Vector Space Basis Change and Relevance Feedback

In the VSM, the score of a document d vs. a query Q_{int} is often expressed by the inner product: $RSV(d, Q_{int}) = d^T.Q_{int}$.

If now the document and the query are generated by the basis B which is parameterized by the transition matrix M, this score becomes

$$RSV(M.d, M.Q_{int}) = d^T.M^T.M.Q_{int}.$$

This score represents the score of the document d, in the original basis, vs. the query $Q_{new} = M^T.M.Q_{int}$. Hence the VSBC has an effect of query reformulation: Q_{new} is the reformulated query.

4 Experiments

In this section we present experiments and results obtained to evaluate our approach.

4.1 Test Collection

The test collection Disk4&5 is used in this study. The Disk4&5 collection contains newswire articles from various sources, such as Association Press, Wall Street Journal, Financial Times, etc., which are usually considered as high-quality text data with little noise.

Table 1. The TREC task and topic numbers associated with Disk4&5 collection

Task	Queries	Docs
TREC 2004, Robust	301-450	528,155

Since the actual queries used in a real application and feedback is expected to be most useful for short queries [35], in all experiments, we only use the title field of the TREC queries for retrieval. In the process of indexing and querying, each term is stemmed using Porter's English stemmer [21] and stopwords from InQuery's standard stoplist [11].

The most common performance measure in the TREC community is Mean Average Precision (MAP) which provides a single-figure measure of quality across recall levels. Among evaluation measures, MAP has been shown to have especially good discrimination and stability. For a single information need, Average Precision is the average of the precision values obtained for the set of top documents existing after each relevant document is retrieved, and this value is then averaged over information needs. The MAP performance measure for the top 1000 documents is used as evaluation metric.

[3] The first p elements of the diagonal are equal to α and the $N - p$ other elements are equal to $\alpha + 1$.

Table 2. Retrieval performance comparison

nb of terms	Rocchio+$BM25$	IRiX	s_1^+	IBM1	IBM2	Our model
10	0.2465	0.2312	0.2279	0.2513	0.2545	0.2601
20	0.2504	0.2374	0.2301	0.2575	0.2611	0.2674
30	0.2545	0.2411	0.2322	0.2603	0.2673	0.2712
50	0.2533	0.2424	0.2387	0.2631	0.2689	0.2757
Average	0.2511	0.2380	0.2322	0.2581	0.2630	0.2686

4.2 Baseline Models and Parameter Settings

In our experiments, we compare our model with the traditional combination of $BM25$ and Rocchio's feedback model, the combination of $BM25$ and LSI (s_1^+ [1]) and the based context model (IRiX [20]). In addition, we also compare our proposed model with the models IBM1 and IBM2 proposed in [15, 16], respectively.

In the initial ranking, the documents were weighted by the $BM25$ formula proposed in [24].

For the IBM1, IBM2 models and our model, the reformulated query is

$$Q_{new} = M^T.M.Q_{int}, \quad (16)$$

where M is the transition matrix computed from Equations 6, 7 and 15, respectively.

We incorporate the VSBC into the Rocchio's model and we propose a new VSBC-based model called VSBCRoc4.

We compare VSBCRoc4 with the VSBCRoc2, VSBCRoc3 models proposed in [18]. Note that VSBCRoc1 is the classic Rocchio model (there is no VSBC).

— The initial query Q_{int} is made from a short topic description, and using it, the top 1000 documents are retrieved from the collections.

— R is the set of top ranking p documents assumed to be relevant.

— S is the set of retrieved documents $501-1000$, assumed to be irrelevant. This strategy is widely used in IR [24, 3] and it is based on plausible heuristics rather than a theory.

For all the VSBC-based models in our experiments, there are several controlling parameters to tune. In order to find the optimal parameter setting for fair comparisons, we use a training method for both the baselines and our approaches. In particular, first, we sweep the values of b, α, β and γ, λ for $BM25$, our approach, Rocchio's formula, IBM1 model and s_1^+ model respectively from 0 to 1.0 with an interval of 0.1. Second, for parameters in RF models, the number of relevant documents $p \in \{1, 2, 3, 4, 5\}$ and the number of irrelevant documents is $N-p \in \{N-1, N-2, N-3, N-4, N-5\}$, where N is the number of expansion terms and it can have a value from $\{10, 20, 30, 50\}$. Note that, the selected $N-p$ irrelevant documents generate the set S, i.e., each irrelevant document is a linear combination of the $N-p$ selected documents. Finally, we vary the dimensionality LSI parameter k from 10 to 300, in steps of 10.

For our approach and the baseline models, the retrieved documents are ranked by the inner product[4] calculated as

$$RSV(Q_{new}, d) = Q_{new}^T.d. \quad (17)$$

4.3 Comparison with VSBC-based Models

From Table 2, we can clearly see that the classic Rocchio's model achieves improvements of 13.31%, 5.50% and 8.14% over $BM25$, IRiX and s_1^+, respectively, on the Disk$4\&5$ collection, while IBM1 and IBM2 obtain significant improvements over the classic Rocchio's model (2.79%, 4.74%, respectively).

In general, our proposed model obtains more improvements over the Rocchio's model and the

[4] Among a variety of similarity measures, inner product similarity is commonly used [2, 30, 31]

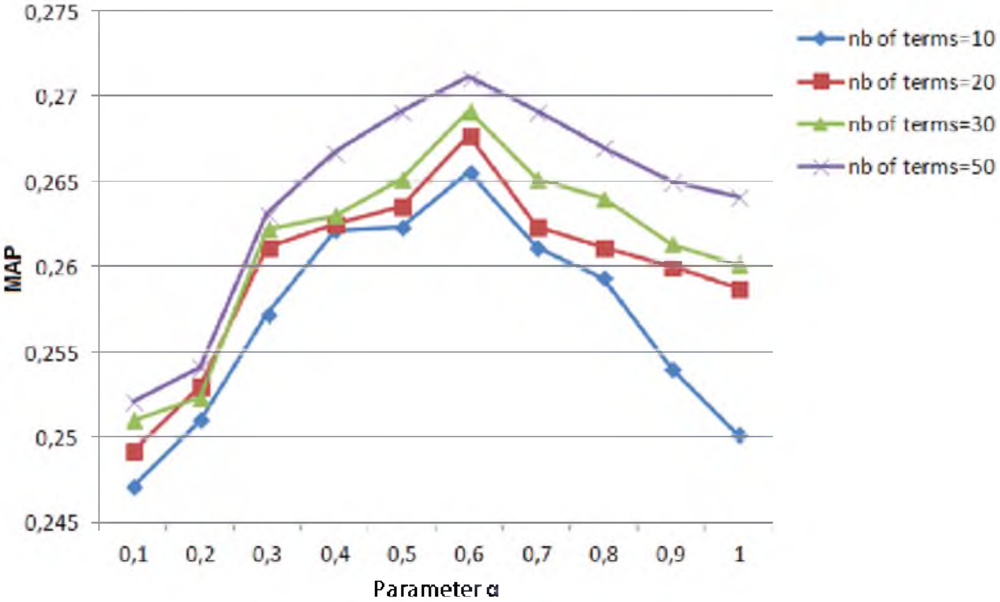

Fig. 1. Our model over disk$4\&5$ with 10, 20, 30 and 50 expansion terms, by α

models proposed in [15, 16]. Specifically, from Table 2, we observe that our model outperforms the classic Rocchio's model (6.97%), surpasses the IBM1 model (4.07%) and exceeds the IBM2 model (2.13%) significantly[5], which demonstrates the effectiveness of our proposed model.

4.4 Impact of Parameters

In our proposed model, there are two important parameters: (1) α controls, first, how to contract the difference between a relevant document and the centroïd of relevant documents, and second, how to dilate the difference between an irrelevant document and the centroïid of relevant documents and (2) p the number of relevant documents. The parameter p also generates the number of irrelevant documents.

The parameter α is a key parameter because it determines a new representation of documents such that relevant documents are gathered and the irrelevant documents are kept away from the relevant ones. Then, in our experiments we attempt to obtain the optimal value of α which gives the better improvement. From Figure 1, we show how the performance of our model changes with the value of α. We investigate a range of α from 0.1 to 1, and the numbers of expansion terms are 10, 20, 30 and 50. The best MAP value is 0.2715 when the value of α is 0.6 an the number of expansion terms is 50.

The parameter p is also a key parameter which generates two entities (R and S). In Figure 2, we show how the performance of our model changes with the value of p. We investigate a range of p from 1 to 10, and the numbers of expansion terms are 10, 20, 30 and 50. The best MAP value is 0.2711 when the value of p is 3 and the number of expansion terms is 50.

4.5 Comparison of VSBCRoc4 with VSBCRoc1, VSBCRoc2 and VSBCRoc3

The VSBC was incorporated into the Rocchio's model by Mbarek et al. in [18]. In this paper Mbarek et al. proposed two VSBC-based Rocchio's models called VSBCRoc2 and VSBCRoc3. Note that VSBCRoc1 is the classical Rocchio's model (there is no VSBC). If we incorporate our VSBC technique into the classical Rocchio's

[5] Statistically significant improvement over Rocchio's model, IBM1 and IBM2 models according to the Student t-test at the 0.05 level.

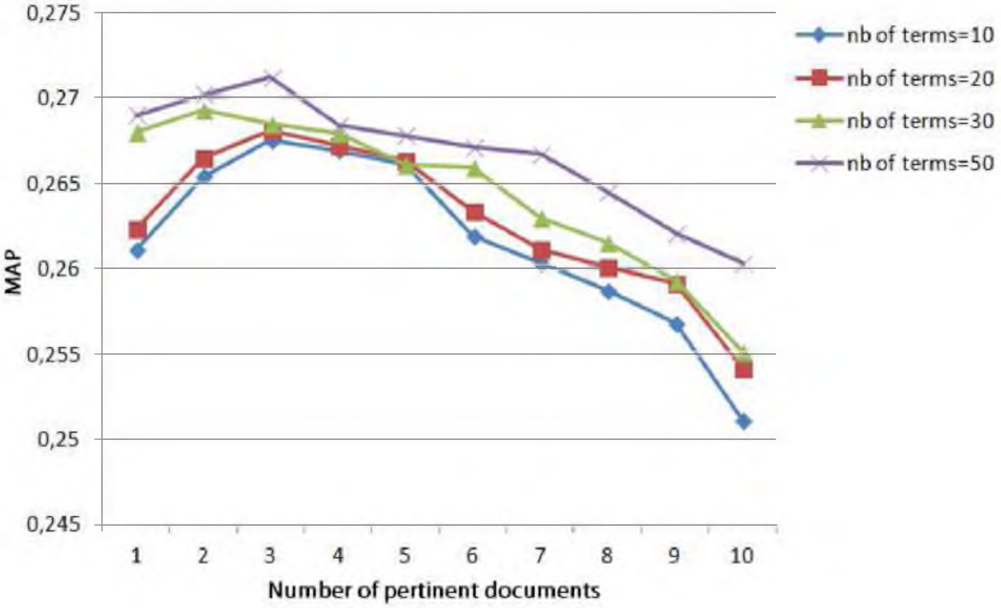

Fig. 2. Our model over disk$4\&5$ with 10, 20, 30 and 50 expansion terms, by the number of documents

model, we obtain a new VSBC-based Rocchio's models called VSBCRoc4. In this section we compare VSBCRoc4 with VSBCRoc$i$ $(1 \leq i \leq 3)$.

From Table 3, we can clearly see that our proposed model obtains more improvements over the classic Rocchio's model and the models proposed in [18]. Specifically, in Table 3, we observe that our model outperforms the classic Rocchio's model (7.48%) and surpasses the models of [18] (3.09%-6.93%) significantly[6], which demonstrates the effectiveness of our proposed model.

5 Conclusion

This paper proposes an RF algorithm based on a VSBC. The VSBC consists of using a transition matrix: by changing the basis, each vector component changes depending on this matrix. In this paper, a transition matrix, that puts the relevant documents gathered to their centroïd (g_R) and the irrelevant documents far from it, is computed to guide the RF process.

In this work, an algorithm for RF to compute the transition matrix is devised.

The starting idea is to build a transition matrix which contracts the difference between relevant documents and g_R and dilates the difference between irrelevant documents and g_R. And so each vector difference is an eigenvector of this transition matrix. Using the decomposition of diagonalized matrix (product of the transpose of the eigenvectors matrix, a diagonal matrix and the eigenvectors matrix), we obtain our transition matrix. When the transition matrix is built, we incorporate the VSBC in the classical Rocchio's algorithm and we obtain a new model called VSBCRoc4.

What makes our approach different from previous works is the assumption that we use an algebraic method to compute the transition matrix.

The proposed model based on VSBC is evaluated on a standard TREC collection. We showed that our approach is very effective and outperforms the VSBC-based models in different frameworks and the improvements are statistically significant. Additionally, we analyze the influence of the parameters α and p in the performance of our model. We

[6] Statistically significant improvement over VSBCRoc1, VSBCRoc2 and VSBCRoc3 according to the Student t-test at the 0.05 level.

Table 3. Comparison of retrieval performance

♯ of terms	VSBCRoc1= Rocchio	VSBCRoc2	VSBCRoc3	VSBCRoc4
10	0.2465	0.2512	0.2538	0.2607
20	0.2504	0.2567	0.2591	0.2691
30	0.2545	0.2641	0.2651	0.2713
50	0.2533	0.2674	0.2695	0.2783
Average	0.2511	0.2524	0.2618	0.2699

intend to apply other algebraic operator (like vector product) to build a geometric RF algorithm.

References

1. **Atreya, A. & Elkan, C.** (**2010**). Latent semantic indexing (LSI) fails for TREC collections. *SIGKDD Explorations*, 12(Issue 2), 5–10.
2. **Baeza-Yates, R. & Ribeiro-Neto, B.** (**1999**). *Modern Information Retrieval*. ACM Press, Addison-Wesley.
3. **Basile, P., Caputo, A., & Semeraro, G.** (**2011**). Negation for document re-ranking in ad-hoc retrieval. In *ICTIR*. 285–296.
4. **Cao, G., Nie, J.-Y., Gao, J., & Robertson, S.** (**2008**). Selecting good expansion terms for pseudo-relevance feedback. In *SIGIR*. 243–250.
5. **Croft, B. W. & Harper, D. J.** (**1979**). Using probabilistic models of information without relevance information. *Journal of Documentation*, 35(4), 285–295.
6. **Croft, W. B., Cronen-Townsend, S., & Lavrenko, V.** (**2001**). Relevance feedback and personalization: A language modelling perspective. In *DELOS Workshop*. 49–54.
7. **de Campos, L. M., Fernández-Luna, J. M., & Huete, J. F.** (**2001**). Relevance feedback in the Bayesian network retrieval model: An approach based on term instantiation. In *IDA*. 13–23.
8. **Deerwester, S. C., Dumais, S. T., Landauer, T. K., Furnas, G. W., & Harshman, R. A.** (**1990**). Indexing by latent semantic analysis. *Journal of the ASIS*, 41(6), 391–407.
9. **Harman, D.** (**1992**). Relevance feedback revisited. In *SIGIR*. 21–24.
10. **Ide, E.** (**1971**). New experiments in relevance feedback. In *SMART*. 337–354.
11. **James, A., Connell, M., Croft, W. B., Feng, F., Fisher, D., & Li, X.** (**2000**). INQUERY and TREC-9. In *TREC*.
12. **Jay, M. P. & Croft, W. B.** (**1968**). A language modeling approach to information retrieval. In *SIGIR*. 275–281.
13. **Lv, Y. & Zhai, C.** (**2010**). Positional relevance model for pseudo-relevance feedback. In *SIGIR*. 579–586.
14. **Manning, C. D., Raghavan, P., & Schütze, H.** (**2008**). *Introduction to Information Retrieval*. Cambridge University Press, UK.
15. **Mbarek, R. & Tmar, M.** (**2012**). Relevance feedback method based on vector space basis change. In *SPIRE*. 342–347.
16. **Mbarek, R., Tmar, M., & Hattab, H.** (**2014**). A new relevance feedback algorithm based on vector space basis change. In **Gelbukh, A.**, editor, *Computational Linguistics and Intelligent Text Processing. Proceedings of CICLing 2014, 15th International Conference on Intelligent Text Processing and Computational Linguistics, Kathmandu, Nepal*, volume 8404 of *Lecture Notes in Computer Science*. 355–366.
17. **Mbarek, R., Tmar, M., & Hattab, H.** (**2014**). An optimal context for information retrieval. In *AAIM*. 323–330.
18. **Mbarek, R., Tmar, M., & Hattab, H.** (**2014**). Rocchio model based on vector space basis change for pseudo relevance feedback. In *SLATE*. 215–224.
19. **Melucci, M.** (**2005**). Context modeling and discovery using vector space bases. In *CIKM*. 808–815.
20. **Melucci, M.** (**2008**). A basis for information retrieval in context. *ACM Trans. Inf. Syst.l*, 26(3), 1–41.
21. **Porter, M.** (**1980**). An algorithm for suffix stripping. *Program*, 14, 130–137.
22. **Robertson, S. & Spärck-Jones, J.** (**1976**). Relevance weighting of search terms. *Journal of the American Society for Information Science*, 27(3), 129–146.
23. **Robertson, S. E. & Walker, S.** (**1994**). Some simple effective approximations to the 2-Poisson model for probabilistic weighted retrieval. In *SIGIR*.

24. **Robertson, S. E., Walker, S., Hancock-Beaulieu, M., Gull, A., & Lau, M.** (**1992**). Okapi at TREC. In *TREC*. 21–30.

25. **Rocchio, J.** (**1972**). Relevance feedback in information retrieval. In *The SMART retrieval system-experiments in automatic document processing*. 313–323.

26. **Ruthven, I. & Lalmas, M.** (**2003**). A survey on the use of relevance feedback for information access systems. *The Knowledge Engineering Review*, 18(2), 95–145.

27. **Ruthven, I., Lalmas, M., & Rijsbergen, K.** (**2002**). Ranking expansion terms with partial and ostensive evidence. In *Fourth international conference on conceptions of library and information science: emerging frameworks and methods*. 199–219.

28. **Sakai, T., Manabe, T., & Koyama, M.** (**2005**). Flexible pseudo-relevance feedback via selective sampling. *ACM Transactions on Asian Language Information Processinge*, 4(2), 111–135.

29. **Salton, G.** (**1968**). *Automatic Information Organization and retrieval*. McGraw-Hill, New-York.

30. **Salton, G.** (**1989**). *Automatic Text Processing: The Transformation, Analysis, and Retrieval of Information by Computer*. Addison-Wesley.

31. **Salton, W., Wong, S., & Yang, C.** (**1975**). A vector space model for automatic indexing. *Communications of the ACM*, 18(11), 613–620.

32. **Tao, T. & Zhai, C.** (**2006**). Regularized estimation of mixture models for robust pseudo-relevance feedback. In *SIGIR*. 162–169.

33. **van Rijsbergen, C.** (**2004**). *The Geometry of Information Retrieval*. Cambridge University Press, Cambridge.

34. **Xu, Y., Jones, G. J., & Wang, B.** (**2009**). Query dependent pseudo-relevance feedback based on Wikipedia. In *SIGIR*. 59–66.

35. **Zhai, C. & Lafierty, J.** (**2001**). Model-based feedback in the language modeling approach to information retrieval. In *CIKM*. 403–410.

36. **Zhou, D., Lawless, S., & Wade, V.** (**2012**). Improving search via personalized query expansion using social media. *Information Retrieval*, 15, 218–242.

37. **Zhou, D., Truran, M., Liu, J., & Zhang, S.** (**2013**). Collaborative pseudo-relevance feedback. *Expert Systems with Applications*, 40, 6805–6812.

Rabeb Mbarek received a Master degree in Computer Science from the High Institute of Computer Science and Multimedia, University of Sfax, Tunisia. She is a member of Multimedia Information systems and Advanced Computing Laboratory. Currently she is a Ph.D student. Her research interests are information retrieval, query optimization and language modeling.

Mohamed Tmar holds Ph.D. in Computer Science, University of Paul Sabatier, Toulouse, France (2002). He is a member of Multimedia Information systems and Advanced Computing Laboratory, Sfax, Tunisia. His research interests are information retrieval, information filtering, XML and multimedia retrieval, query optimization and language modeling.

Hawete Hattab has Ph.D. in Mathematics from the Sfax University (2004). He is an Associate Professor at Umm Al-Qura University, Department of Mathematics. He is a member of Dynamical Systems and Combinatory Laboratory, Sfax, Tunisia. His main research interests are dynamical systems and information systems.

Article received on 07/01/2014; accepted on 30/01/2014.