# Evaluation of a Methodology for Modeling Term Relationship through Geometry: Experiments at TREC 2010 Relevance Feedback Track

Emanuele Di Buccio*, Massimo Melucci* and Jian-Yun Nie**

*Department of Information Engineering,
University of Padua, Italy
{dibuccio,melo}@dei.unipd.it

**Department of Computer Science and Operations Research,
University of Montreal, Canada
nie@iro.umontreal.ca

**Abstract.** The work reported in this paper is focused on the experimental evaluation of a methodology which models sources for feedback through a vector subspace formalism. This work considers a specific application of the methodology that exploits correlation among terms in documents judged as relevant to support feedback. Experiments were carried out during the participation to the TREC 2010 Relevance Feedback Track, thus investigating the effectiveness of the methodology application for modeling term correlation on a very large text corpus and when little evidence, namely one relevant document, is used as input for feedback.

## 1 Introduction

One issue when dealing with feedback strategies is to understand which factors affected the user perception of relevance of a document. These factors should be modeled and used to improve the information need description submitted to the system by the user. In this work it is suggested that this objective can be achieved by a methodology based on a geometric framework, where factors are uniformly modeled as vector subspaces.

The contribution of this paper is the experimental evaluation of an application of the methodology to Explicit Relevance Feedback (RF). The methodology application hypothesizes that correlation among terms in relevant documents is one of these factors and investigates this hypothesis on a very large text corpus and using little evidence as input for feedback.

The experiments were carried out during the participation to the TREC 2010 Relevance Feedback track. The scenario considered in the RF track supposes that the user submits a query, obtains a list of results and provides relevance judgment on a single relevant document. The Information Retrieval (IR) system then can perform feedback on the basis of the initial information need representation, i.e. the submitted query, and possible information extracted from the feedback document.

The specific approach adopted in this work for modeling term correlation is to obtain a vector subspace representation on the basis of local co-occurrence data on a subset of the terms appearing in the feedback document. The modeling approach was originally proposed in [1] and applied to Pseudo-relevance Feedback, then adopted in an Explicit RF scenario when up to five relevant documents are available [2]. It shares some intuitions proposed in previous works where term relationships are modeled within vector spaces. Similarly to the Hyperspace Analogues to Language (HAL) spaces [3], term-by-term correlation is obtained from the text corpus using a sliding window; all the terms co-occurring within the window are considered as related to each other. In HAL spaces the strength of the relationship is inversely proportional to the distance between the terms and in the original proposal was directional. In this work the decaying factor is not considered and the relationship is not directional, as in [4]. Another aspect that our approach shares with the work of Bai et al., spefically their "local" approach, is that term relationship are extracted from the feedback set only — in their case from the pseudo feedback set, in our case from the set of documents explicitly judged as relevant. The vector subspace representation of term correlation is computed by Singular Value Decomposition (SVD) of the matrix prepared with local co-occurrence data. SVD is, for instance, adopted in Latent Semantic Indexing (LSI) [5]. In LSI term relationship can be captured in the reduced space: in [6] the authors investigate the values of the term–term correlation reduced matrix and their relationship with high order term co-occurrence. For instance, a second order co-occurrence between two terms $a$ and $c$ exists when a term $a$ co-occurs with $b$ and $b$ co-occurs with $c$. A strong correlation was observed between second-order term co-occurrence and values obtained by SVD. A "local" version of LSI is proposed in [7] for the routing problem: LSI is applied on the matrix whose rows are relevant documents and whose columns are factors extracted by LSI on the original matrix. In our case the decomposition is applied to the local correlation matrix obtained from the feedback set.

This work extends that reported in [2] investigating the effectiveness of diverse implementation of the methodology steps and using only one relevant document as source for feedback. Section 2 describes a general methodology to support feedback that has been investigated for diverse sources of evidence and the specific application adopted for the explicit feedback scenario. Section 3 describes the experimental setting adopted, specifically the parsing and indexing procedure — Section 3.1, the baseline and how topics were parsed — Section 3.2, and the evaluation methodology for the adopted feedback strategy — Section 3.3. Results obtained from the experimental evaluation are reported in Section 4. Section 5 reports some concluding remarks.

## 2 Methodology

The specific methodology tested in the TREC 2010 Relevance Feedback Track is a specific application of that proposed in [8]. The general objective of the methodology can be explained considering the following scenario. Let us suppose a user interacts with an IR system and submits a first description of his information need, e.g. a query statement. He obtains a list of results and examines some of them. At this point a number of sources of evidence can be potentially adopted to support feedback, e.g. properties of the results and the corresponding documents or the behavior of the user when interacting with them. One of the main issues is how to obtain a uniform representation of the diverse source contributions and exploit them as new dimensions of the information need representation, thus helping refine the initial information need formulation. In [8] it is suggested that the vector subspace formalism proposed in [1] can be adopted to achieve this objective. The basic rationale of that framework is that, once a source contribution has been modeled as a vector space basis and a document as a vector, the subspace spanned by the basis vectors is adopted as a new dimension of the information need representation: the degree to which a document satisfies the dimension can be measured by the distance between the document vector and the dimension subspace. Documents can be re-ranked according to this measure.

The methodology is constituted by a set of steps. In the following they are discussed when applied to the specific case of term correlation in an explicit feedback scenario:

**Source Selection.** The first step is the selection of the source. This step requires the definition of the hypothesis on the possible factors that can affect the user perception of relevance when feedback is explicitly or implicitly provided by the user. Moreover, this step requires the selection of the source from which features are actually gathered. In this work it is hypothesized that correlation among terms modeled using local co-occurrence data in a set of feedback documents can provide the system with information on the user perception of relevance. The actual source adopted to distill evidence is a single relevant document.

**Evidence Collection.** The second step concerns the actual collection of the evidence, which is then adopted to model the dimension. Since in this work the dimension to be modeled is the correlation among terms in the relevant document, the evidence used is the terms appearing in the feedback document. The features extracted to model term correlation are statistical information on term occurrence in the document and in the collection, and term positional information to compute local co-occurrence in window of text of fixed size. Given a set of feedback documents, a possible solution is to adopt only terms appearing in the user provided description and exploit the evidence extracted from the feedback set to capture possible relationships among them. But usually textual queries are short, as in the test collection adopted in the RF track. Queries can

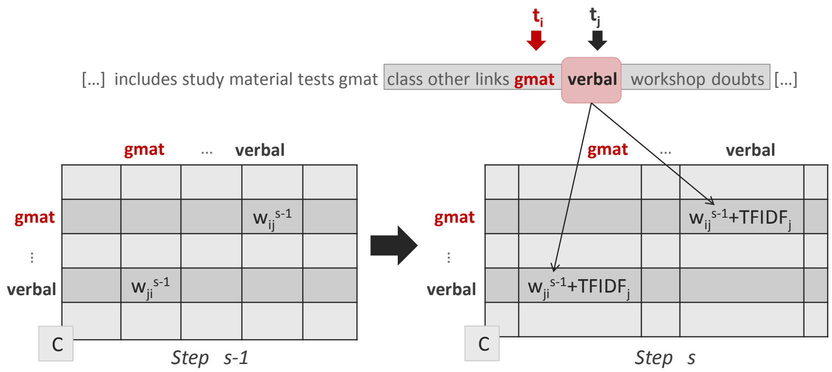


Fig. 1: Matrix preparation.

benefit from expansion based on other terms occurring in the document of the feedback set [19]. A possible approach is to consider all the terms in the feedback documents as good terms, but this could add too much noise. Therefore a term selection strategy should be adopted. The experiments reported in this work were performed selecting terms with highest IDF, i.e. hypothesizing that the most discriminative terms in the collection and present in the document are "good" expansion terms.

**Dimension Modeling.** Term correlation can be modeled using local co-occurrence data as follows:

- **Matrix Preparation**. Let $\mathcal{T}$ be the set of terms selected from the source and let $C \in \mathbb{R}^{|\mathcal{T}|\times|\mathcal{T}|}$ be a matrix whose elements are initially set to zero, namely $c_{ij} = 0$ for $1 \leq i, j \leq |\mathcal{T}|$. For each term $t_i \in \mathcal{T}$ a window of text centered around each occurrence of $t_i$ is considered; if a term $t_j \neq t_i \in \mathcal{T}$ appears in the window of text, statistical information about $t_j$, e.g. its total frequency in the collection, or a weight derived from such information, e.g. the TF·IDF, is added both to $c_{ij}$ and $c_{ji}$. A pictorial description is reported in Figure 1. The text window is centered on the word $t_i =$"gmat". Since the term $t_j =$"verbal" belongs to the terms selected to expand the query, the weight of "verbal" is added to both the entries $c_{ij}$ and $c_{ji}$ of the matrix $C$, namely those referring to the correlation between $t_i$ and $t_j$. The value $w_{ij}^{s-1}$ and $w_{ji}^{s-1}$ refer respectively to the weight in $c_{ij}$ and $c_{ji}$ at the step $s - 1$.
- **Matrix Decomposition and Basis Vectors Selection.** A possible solution to obtain a vector space basis from the matrix $C$ is SVD. The matrix is decomposed as $C = U\Sigma V^T$, where $\Sigma \in \mathbb{R}^{n\times n}$ and $U, V \in \mathbb{R}^{|\mathcal{T}|\times n}$, with $U = V$ since $C$ is symmetric; the columns of $U$ constitute an orthonormal vector space basis. A subset of the basis vectors is adopted to model the dimension. Therefore, if $U = [\mathbf{b}_1, \ldots, \mathbf{b}_{\mathcal{T}}]$ and a subset $\{\mathbf{b}_q, \ldots, \mathbf{b}_{q+r}\}$ is

selected, the subspace $L(R_F) = span(\{\mathbf{b}_q, \ldots, \mathbf{b}_{q+r}\})$ is adopted as model of the dimension.

**Document Representation and Prediction.** A document in the framework is represented as a vector, whose entries can be, for instance, term weights. The basic rationale underlying prediction is to measure the degree to which a document representation satisfied the new dimension of the information need representation modeled on the basis of the evidence gathered from the source. Once a subspace representation for the dimension and a vector representation for the document have been obtained, prediction can be performed by measuring the distance between the document vector $\mathbf{y}$ and the dimension subspace $L(R_F)$; in particular,

$$\mathrm{m}_{L(R_F)}(\mathbf{y}) = \mathbf{y}^T \cdot \mathrm{P}_{L(R_F)} \cdot \mathbf{y}. \tag{1}$$

where $\mathrm{P}_{L(R_F)}$ is the projector onto the subspace $L(R_F)$. This is a trace-based function. The idea of using trace in IR, and in particular the density operators, was originally introduced in [11], and one of its important consequence was to "establish a link between geometry and probability in vector spaces" [11]. The specific function described by Equation 1 is discussed in [1].

## 3 Experiments

The IR system adopted in the experiments exploits the functionalities provided by Apache Lucene [12] for indexing and retrieval[1]. The specific choices made in regard to parsing, indexing and retrieval are described in the remainder of this section. The experiments were carried out on the ClueWeb09 Category B corpus, constituted by 50,220,423 English web pages.

### 3.1 Parsing and Indexing

Each web-page of the ClueWeb09 Category B was parsed and the following information was extracted from each record in Web ARChive (WARC) format: the TREC-ID, the URI and the content. Each of them was stored in a distinct `Field` of a Lucene `Document`[2]. All the content of the document was processed during indexing except for the text contained inside the `<script></script>` and the `<style></style>` tags. When parsing, the title of the document was extracted and considered as the beginning of the document content.

Stop words were removed during indexing[3]. No stemming was adopted. During indexing not only statistical information about the occurrence of the terms in the documents, namely their frequency, was stored, but also information about

[1] The specific version adopted in the experiments was Apache Lucene 2.4.1

[2] "A `Document` represents a collection of fields [...] Each field corresponds to a piece of data that is either queried against or retrieved from the index during search" [13]

[3] The stop words list is that available at the url http://ir.dcs.gla.ac.uk/resources/linguistic_utils/stop_words

the positions where terms occurred[4]. The information about the position of the terms was used to implement the methodology described in Section 2 and exploited for feedback as described in Section 3.3.

### 3.2 Retrieval: Baseline

**Query Parsing.** The topics adopted in the experiments were the first even one hundred topics from the TREC 2009 Million Query track. Each of the one hundred topics was automatically parsed. Two different topic parsing approaches were adopted, depending on whether or not the topic was expressed using a specific query syntax, e.g. the presence of quotation marks or the plus sign.

If no special syntax was adopted in the topic string, the constituting terms were extracted; no stemming was adopted on the obtained terms. For each term $q_i$ in a topic we constructed a Lucene `TermQuery` for the content field, that is a query to retrieve all the documents where the term $q_i$ appears in their content field. The `TermQuery`'s constructed for the terms $q_i$'s in a topic were combined in a Lucene `BooleanQuery`: each `TermQuery` was considered as an optional clause, that is `TermQuery`'s were combined by logical `OR`[5].

In the event of topics with special syntax, i.e. topic 20152 ("pectin+rich+fruit") and topic 20962 ("diy+audio"), the punctuation was removed and the query was automatically translated in the Lucene query syntax, then transformed by the Lucene `QueryParser`. For instance, the topic string *"pectin+rich+fruit"* was translated in *+pectin +rich +fruit*, whose meaning is that the terms "pectin", "rich" and "fruit" must appear in the searched document field[6].

**Weighting Scheme.** The weighting scheme adopted was the BM25 [15], particularly exploiting the implementation for Lucene made available in [14] where the normalization constant $(k_1 + 1)$ is not adopted. A brief description is reported in the following. Let $V_D$ be the set of terms appearing in document $D$; the weight $w_i$ assigned to the term $t_i \in V_D$ is

$$w_i = \frac{tf'_i}{k_1 + tf'_i} \log \frac{N - n_i + 0.5}{n_i + 0.5}$$

where $N$ is the total number of document in the collection, $n_i$ is the number of documents in the collection where the term $t_i$ appears, and $k_1$ is a parameter

[4] In Lucene information about the unique terms in a field, their counts, their positions and their offsets can be stored at indexing time and then accessed by using `TermVectors`. The specific `TermVector` option chosen for the Lucene `Field` used for the "content" was `TermVector.WITH_POSITIONS`

[5] The specific boolean operator adopted for the Lucene `BooleanQuery` was `BooleanClause.Occur.SHOULD`

[6] Apache Lucene - Query Parser Syntax description can be found at: http://lucene.apache.org/java/2_4_1/queryparsersyntax.html

which was heuristically set to $k_1 = 2$ in the experiments. The quantity $tf'_i$ is defined as $tf'_i = tf_i/B$, where $tf_i$ is the term frequency of $t_i$, and

$$B = (1 - b) + b\frac{dl}{avdl}$$

where $dl = \sum_{t_i \in V_D} tf_i$ is the document length[7], and $avdl$ is the average document length in the collection. The value of $b$ adopted in the experiments was $b = 0.75$.

### 3.3 Retrieval: Feedback

**Test Collection.** The test collection adopted in the TREC 2010 Relevance Feedback track was constituted by a subset of the ClueWeb09 web corpus — Category A or Category B, one hundred topics and two groups of feedback sets. Each group was constituted by five sets of relevant documents, for a total of ten possible feedback sets $\mathcal{S} = \{10\text{-}1, \ldots, 10\text{-}10\}$. Documents in a set $10\text{-}i \in \mathcal{S}$ were selected according to specific criteria briefly reported in the following:

1. a randomly chosen document from among the topic's known relevant documents;
2. the most commonly returned relevant document in TREC 2009;
3. the least commonly returned relevant document in TREC 2009;
4. the longest relevant document;
5. the shortest relevant document;
6. another random relevant document;
7. the most spammy relevant document, determined by the approach proposed in [18];
8. the least spammy relevant document;
9. a random highly relevant document;
10. the most commonly returned non-relevant document.

**Methodology.** Given a set of relevant documents $10\text{-}i \in \mathcal{S}$, for each topic the participants should perform feedback using the relevant document corresponding to such topic in the set $10\text{-}i$. The runs using the feedback sets 10-1–10-5 were mandatory, while those based on 10-6–10-10 were optional; we submitted both mandatory and optional runs. The specific evaluation methodology based on the approach described in Section 2, can be summarized by the following steps performed for each topic, once a document set was selected:

1. **Evidence Collection**: Extract the top $k = 10$ terms among those appearing in the documents with highest IDF and expand the query, constituted by the topic keywords, with the selected terms. The expanded query has $k + h$ terms where $h$ is the number of terms in the initial query.

[7] Document lengths are stored in `norm` in Lucene and the encoding/decoding procedure can cause a loss of precision — see http://lucene.apache.org/java/2_4_1/scoring.html We have not investigated the impact of this loss of precision on the retrieval effectiveness.

2. **Dimension Modeling**:
   a– Computation of the local co-occurrence matrix $C$ by windows of text in the considered relevant document. In particular a window of text of size 7 is centered around each occurrence of a keyword $t_i \in \mathcal{T}$. If a keyword $t_j \in \mathcal{T}$ appears in the window of text centered around $t_i$, the TF·IDF weight of $t_j$ is added to the elements $c_{ij}$ and $c_{ji}$ of $C$.

   b– Decomposition of the matrix $C$ by SVD[8].

   c– Selection of the first eigenvector $\mathbf{b}$ and adoption of the subspace $L(R_F) = L(\{\mathbf{b}\})$ as model of the dimension.
3. **Document Representation**: Represent each document as a vector $\mathbf{y} \in \mathbb{R}^{k+h}$, where $y_i$ is the BM25 weight of the term $t_i$ in the considered document.
4. **Prediction**: Re-ranking of the top $m = 2500$ results retrieved by the baseline according to the distance between the vector representation of the document and the computed subspace; the specific function adopted is Eq. 1, that is $\mathbf{y}^T \cdot \mathrm{P}_{L(R_F)} \cdot \mathbf{y} = (\mathbf{b}^T\mathbf{y})^2$, where $\mathbf{y}$ is the document vector and $\mathrm{P}_{L(R_F)}$ the projector onto the subspace $L(R_F) = L(\{\mathbf{b}\})$.

## 4 Results

The results for the diverse feedback sets are reported in Table 1. Since TREC 2010 RF Track submissions were pooled to depth 10, results on the effectiveness of the methodology application are reported using a measure that considers the top ten documents, specifically P@10. Results show that the most effective re-ranking in terms of P@10 is obtained when the longest relevant document (rf10-4) is adopted for feedback. But the paired two-tailed t-test (95% confidence interval) shows that the differences among the diverse feedback sets are not significant. When compared with the baseline, i.e. the first stage prediction with no feedback, dimension-based re-ranking negatively affected the baseline ranking for all the feedback sets. Results for some of the feedback sets (those marked by stars in Table 1) should be considered preliminary since some of the documents in the top 10 for topic 20696 were in the pool but they were omitted from judging.

In order to gain insights into the effectiveness of the methodology application when a recall oriented measure is adopted, we performed the evaluation using the prels of the TREC 2009 Million Query Track; the adopted measure was the statMAP. Results are reported in Table 2 and show that the most effective re-ranking is obtained when the shortest relevant document (rf10-5) is adopted for a source for term relationship. Also in this case the differences among the diverse feedback sets were not significant according to the paired two-tailed t-test. One result worth noting is that re-ranking based on the most commonly returned non relevant document (rf10-10) was the second best performing feedback set. When compared with the baseline, also in this case dimension-based re-ranking negatively affected the baseline ranking for all the feedback sets.

[8] In the experiments the JAMA package [16] was used to implement all the functionalities for constructing and manipulating matrices.

| Feedback Set | P@10 B | P@10 FB | $\Delta_{\texttt{FB}-\texttt{B}}$ (%) |
|---|---|---|---|
| rf10-1 | 0.469 | 0.382 | -18.55 |
| rf10-2 | 0.470 | 0.394 | -16.17 |
| rf10-3 | 0.473 | $0.411^{\star}$ | -13.11 |
| rf10-4 | 0.470 | 0.418 | -11.06 |
| rf10-5 | 0.470 | $0.369^{\star}$ | -21.49 |
| rf10-6 | 0.473 | $0.401^{\star\star}$ | -15.22 |
| rf10-7 | 0.474 | $0.406^{\star}$ | -14.35 |
| rf10-8 | 0.471 | $0.389^{\star}$ | -17.41 |
| rf10-9 | 0.476 | $0.388^{\star\star\star}$ | -18.49 |
| rf10-10 | 0.469 | $0.388^{\star}$ | -17.27 |

Table 1: P@10 reported for the baseline (B) and dimension-based re-ranked documents (FB) for the diverse feedback sets. Last column reports the percentage difference between the baseline and dimension-based re-ranking. Results are obtained using residual collections. Results marked by stars are those for which at least one document in the top 10 for topic 20696 was in the pool but was not judged (the number of stars denotes the number of unjudged documents).

| Feedback Set | statMAP B | statMAP FB | $\Delta_{\texttt{FB}-\texttt{B}}$ (%) |
|---|---|---|---|
| rf10-1 | 0.227 | 0.129 | -43.05 |
| rf10-2 | 0.226 | 0.123 | -45.48 |
| rf10-3 | 0.226 | 0.126 | -44.29 |
| rf10-4 | 0.229 | 0.127 | -44.68 |
| rf10-5 | 0.226 | 0.138 | -39.13 |
| rf10-6 | 0.229 | 0.134 | -41.39 |
| rf10-7 | 0.231 | 0.130 | -43.68 |
| rf10-8 | 0.229 | 0.133 | -42.19 |
| rf10-9 | 0.230 | 0.130 | -43.61 |
| rf10-10 | 0.231 | 0.137 | -40.67 |

Table 2: statMAP reported for the baseline (B) and dimension-based re-ranked documents (FB) for the diverse feedback sets. Last column reports the percentage difference between the baseline and dimension-based re-ranking. Results are obtained using residual collections.

## 5 Concluding Remarks

The results of the methodology evaluation reported in this work show that the current implementation of the methodology application for term relationship is not effective to support document re-ranking when a single relevant document is adopted for feedback. Alternative implementations of the methodology steps should be investigated. However, the methodology provides a principled approach for performing this investigation in a single framework and for evaluating the effectiveness of alternative implementations. Indeed, the methodology considered in this work aims at being general and highly modular. For instance, once hypothesized that term correlation is an effective source for feedback when only little evidence is available, one can implement the diverse methodology steps using diverse strategies. For instance, once the modeling step fixed we can investigate the way diverse term selection strategies can affect the overall methodology effectiveness, e.g. using diverse features or diverse combination of features to select terms exploited by unsupervised [9] or supervised [10] techniques. The extraction at query time of features that require collection-wide statistics (as for some of the features adopted in [10] or inter document term ranking functions [20]) from large text corpora can be quite slow: this issue should be addressed, e.g. using a sampling strategy as in [21].

As for the dimension modeling step a possible issue to investigate is the way a greater number of eigenvectors can affect the retrieval effectiveness. Moreover, examining the matrices $C$ in several cases term self-correlation is less than the correlation with other terms of the expanded query. A further research question is to investigate if that can negatively affect retrieval effectiveness as in LSI [17].

## Acknowledgment

The research leading to these results has received funding from the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement N. 247590. The authors thank the members of the RALI Lab of the University of Montreal for the fruitful discussions and the members of the technical support team of the Department IRO of the University of Montreal for the valuable support on the work reported in this paper.

## References

1. M. Melucci. A basis for information retrieval in context. *ACM Transactions on Information Systems*, 26(3):1–41, 2008.
2. E. Di Buccio and M. Melucci. University of Padua at TREC2009: Relevance Feedback Track. In *Proceedings of the Eighteenth Text REtrieval Conference (TREC 2009)*, Gaithersburg, Maryland, November 17-20, 2009.
3. K. Lund and C. Burgess. Producing high-dimensional semantic spaces from lexical co-occurrence. *Behavior Research Methods, Instruments & Computers*, 28(2):203–208, 1996.

4. J. Bai, D. Song, P. Bruza, J.-Y. Nie and G. Cao. Query expansion using term relationships in language models for information retrieval. In *Proceedings of the 14th ACM international conference on Information and knowledge management (CIKM'05)*, pages 688–695, Bremen, Germany, 2005. ACM.
5. S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41:391–407, 1990.
6. A. Kontostathis and W. M. Pottenger. A framework for understanding latent semantic indexing (lsi) performance. *Information Processing and Management*, 42(1):56–73, 2006.
7. D. Hull. Improving text retrieval for the routing problem using latent semantic indexing. In *Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval (SIGIR '94)*, pages 282–291, Dublin, Ireland. Springer-Verlag New York, Inc.
8. E. Di Buccio, M. Lalmas and M. Melucci. From Entities to Geometry: Towards exploiting Multiple Sources to Predict Relevance. In *Proceedings of the First Italian Information Retrieval Workshop (IIR-2010)*, Padua, Italy, January 27-28, 2010, CEUR-WS.org, ISSN 1613-0073, Vol. 560, urn:nbn:de:0074-560-7.
9. J. Xu and W. B. Croft. Improving the effectiveness of information retrieval with local context analysis. *ACM TOIS*, 18(1):79–112, 2000.
10. G. Cao, J.-Y. Nie, J. Gao, and S. Robertson. Selecting good expansion terms for pseudo-relevance feedback. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval (SIGIR'08)*, pages 243–250, Singapore, Singapore, 2008. ACM.
11. C.J. van Rijsbergen. *The Geometry of Information Retrieval*. Cambridge University Press, New York, NY, USA, 2004.
12. Apache Lucene - Overview. http://lucene.apache.org/java. Last Published: 19 June 19, 2010
13. O. Gospodnetić and E. Hatcher. *Lucene in Action*. Manning Publications Co., first edition, 2004.
14. J. Pérez-Iglesias, J. R. Pérez-Agüera, V. Fresno and Y. Z. Feinstein. Integrating the Probabilistic Models BM25/BM25F into Lucene. CoRR. 2009. http://arxiv.org/abs/0911.5046
15. S. E. Robertson and H. Zaragoza. The Probabilistic Relevance Framework: BM25 and Beyond. *Foundations and Trends in Information Retrieval*, 3(4):333–389, 2009.
16. JAMA : A Java Matrix Package. http://math.nist.gov/javanumerics/jama/. Last update: July 13, 2005.
17. L. A. F. Park and K. Ramamohanarao. An analysis of latent semantic term self-correlation. *ACM Transactions on Information Systems*, 27(2):1–35, 2009.
18. G. V. Cormack, M. D. Smucker and C. L. A. Clarke Efficient and effective spam filtering and re-ranking for large web datasets. *CoRR*, abs/1004.5168, 2010.
19. Harman, D. Relevance feedback revisited. In *Proceedings of the 15th annual international ACM SIGIR conference on Research and development in information retrieval SIGIR '92)*, pages 1–10, New York, NY, USA, 1992. ACM.
20. W. S. Wong, R. W. P. Luk, H. V. Leong, K. S. Ho and D. L. Lee. Re-examining the effects of adding relevance information in a relevance feedback environment. *Information Processing and Management*, 44:1086–1116, 2008.
21. M.-A. Cartright, J. Seo and M. Lease. UMass Amherst and UT Austin @ The TREC 2009 Relevance Feedback Track. In Ellen M. Voorhees, L. P. B., editor, *Proceedings of the Eighteenth Text REtrieval Conference (TREC 2009)*, NIST Special Publication: SP 500-278, Gaithersburg, Maryland, USA.