UAM at INEX 2012 Relevance Feedback Track: Using a Probabilistic Method for Ranking Refinement[*]

Esaú Villatoro-Tello, Christian Sánchez-Sánchez, Héctor Jiménez-Salazar, Wulfrano A. Luna-Ramírez, and Carlos Rodríguez-Lucatero

Language and Reasoning Group, Information Technologies Dept., Universidad Autónoma Metropolitana (UAM), Unidad Cuajimalpa, Mexico City. {evillatoro,csanchez,hjimenez,wluna,crodriguez}@correo.cua.uam.mx

Abstract. This paper describes the system developed by the Language and Reasoning Group of UAM for the Relevance Feedback track of INEX 2012. The presented system focuses on the problem of ranking documents in accordance to their relevance. It is mainly based on the following hypotheses: *(i)* current IR machines are able to retrieve relevant documents for most of general queries, but they can not generate a pertinent ranking; and *(ii)* focused relevance feedback could provide more and better elements for the ranking process than isolated query terms. Based on these hypotheses, our participation at INEX 2012 aimed to demonstrate that using some query-related relevance feedback it is possible to improve the final ranking of the retrieved documents.

1 Introduction

Information Retrieval (IR) deals with the representation, storage, organization, and access to information items[1] [1]. Given some query, formulated in natural language by a user, the IR system is suppose to retrieve and sort according to their relevance degree documents satisfying user's information needs [4].

The word *relevant* means that retrieved documents should be semantically related to the user information need. Hence, one main problem of IR is determining which documents are, and which are not relevant. In practice this problem is usually regarded as a *ranking* problem, whose goal is to define an ordered list of documents such that documents similar to the query occur at the very first positions.

Over the past years, IR Models, such as: Boolean, Vectorial, Probabilistic and Language models have represented a document as a set of representative keywords (i.e., index terms) and defined a *ranking* function (or retrieval function)

---

[1] Depending on the context, items may refer to text documents, images, audio or video sequences.

to associate a relevance degree for each document with its respective query [1, 4]. In general, these models have shown to be quite effective over several tasks in different evaluation forums (CLEF[2] and TREC[3]). Nevertheless, these retrieval systems still fail at retrieving most of the relevant documents to a given query in the first positions. The latter is due to the fact that modelling user intentions from queries is, in general, a highly subjective and difficult task, hence, post-processing and ranking refinement strategies have been adopted [12–15].

Post-retrieval techniques aim at refining retrieval results by means of feature re-weighting, query modification, document re-ranking and relevance feedback. The common idea is to interact with the user in order to learn or to improve a model of the underlying user's information need. Acceptable results have been obtained with such methods, however, they still have several limitations, including: *i)* the need of extensive user interaction[4]; *ii)* multiple execution of retrieval models; *iii)* the on-line construction of classification methods; *iv)* the lack of contextual information in the post-retrieval processing, which may be helpful for better modelling users' information needs; and *v)* the computational cost that involves processing the entire collection of documents each feedback iteration.

Document re-ranking or ranking refinement in information retrieval has been a widely research topic during the last fifthteen years. There are two main approaches for this task: *i)* indirect re-ranking via some query expansion strategy, and *ii)* direct re-ranking on initial retrieved documents [15]. Normally, query expansion strategies assume that top ranked documents are more likely to be relevant, the terms contained within these documents can be used to augment the original query and then a better ranking can be expected via a second retrieval process. In contrast, direct re-ranking strategies try to improve the ranking of the initial set of retrieved documents by directly adjusting their positions without the need of performing a second retrieval process, normally, this type of strategy use the information contained within the retrieved documents (*e.g.,* inter-document similarities) to generate a better ranking of them. The generated output (*i.e.,* a list of ranked documents) by any of this two strategies would be of obvious benefit to users, for example, direct ranking refinement can be used to improve automatic query expansion since a better ranking in the top retrieved documents can be expected.

1.1 Our approach

Our participation at the INEX 2012 Relevance feedback track proposes using an alternative post-retrieval technique that aims at improving the results provided

---

[2] Cross Language Evaluation Forum (http://www.clef-campaign.org/).

[3] Text Retrieval Conference (http://trec.nist.gov/).

[4] It is worth mentioning that if available, user interaction should be included in post-retrieval techniques as it is evident that information provided by the user is much more reliable than that obtained by a fully automatic process. Hence, the goal of post-processing techniques should be minimizing users' interaction instead of completely eliminate it from the process.

by a document retrieval system and that overcomes some of the limitations of current post-retrieval methods. Our work classifies as a direct document ranking refinement strategy. In particular we face the problem of re-ranking[5] a list of documents retrieved by some information retrieval system. This problem is motivated by the availability of retrieval systems that present high-recall and low-precision performance, which evidences that the corresponding retrieval system is in fact able to retrieve many relevant documents but has severe difficulties to generate a pertinent ranking of them. Hence, given a list of ranked documents, the problem we approach consists of moving relevant documents to the first positions and displacing irrelevant ones to the final positions in the list.

We propose a solution to the ranking refinement problem based on a Markov Random Field (MRF) [5, 9, 6, 16] that aims at classifying the ranked documents as relevant or irrelevant. Each document in the retrieved list is associated to a binary random variable in the MRF (*i.e.,* a node), the value of each random variable indicates whether a document is considered relevant (when its value is 1) or not (when its value is 0). The MRF considers several aspects: 1) the information provided by the base information retrieval system, 2) similarities among retrieved documents in the list, and 3) information obtained through a relevance feedback process. Accordingly, we reduce the problem of ranking refinement to that of minimizing an energy function that represents a trade-off between document relevance and inter-document similarity. The information provided by the information retrieval system is the base of our method, which is further enriched with contextual and relevance feedback information.

Our motivation for considering context information is that relevant documents to a query will be similar to each other and to its respective query, to some extent; whereas irrelevant documents will be different among them and not as similar to the query as the relevant documents[6]. Relevance feedback information has two main purposes: *i)* to work as a seed generation mechanism for propagating the relevancy/irrelevancy status of nodes (documents) in the MRF, and *ii)* to denote the users' search intention by working as *example texts*.

At this point it is important to mention that, traditionally a relevance feedback process takes as input a set of n documents (tentatively relevant) and generates as output a set of k isolated terms (tentatively relevant to the query) which are further employed for a query expansion process. For our purposes we will employ all the information contained in the feedback (called *example texts*) since by doing this we have showed [12–14] that it is possible to make a more accurate approximation of the users' search intention (*i.e.,* to become into a more explicit representation the implicit information contained in the query).

The proposed MRF does not require of multiple executions of IR models, nor training classification methods, and it can work without user intervention;

---

[5] Also known as the problem of *Ranking Refinement*.
[6] Keep in mind that irrelevant documents will be similar to the query in some degree since such documents were obtained by an IR system through that query in the first place.

therefore, our MRF overcomes the main limitations of current post-processing techniques.

1.2 Structure of the paper

The rest of the paper is organized as follows. Section 2 introduces the proposed Markov Random Field for ranking refinement in document retrieval. Section 3 describes the experimental platform used to evaluate and compare our ranking strategy. Section 4 presents the experimental results. Finally, section 5 depicts our conclusions.

2 System Description

A general outline of the proposed method is given in Figure 1. Given a query, the IR system retrieves from a given collection of documents a list of files sorted according to a relevance criteria. From this list, some relevant documents are selected based on a relevance feedback approach[7]. For each document in the list, the textual features are extracted. The text contained in each document in the list, the query given by the user, and a subset of information selected via relevance feedback, are combined to produce a re-ordered list. This re-ranking is obtained based on a Markov random field (MRF) model that separates the relevant documents from irrelevant ones, generating a new list by positioning the relevant documents first, and the others after. Next we give a brief review of MRFs, and then we describe in detail each component of the proposed method.

2.1 Markov Random Fields

Markov Random Fields (MRF) are a type of undirected probabilistic graphical models that aim at modelling dependencies among variables of the problem in turn [5, 9, 6, 16]. MRFs have a long history within image processing and computer vision [7]. They were first proposed for denoising digital images [5, 9, 6, 16] and since then a large number of applications and extensions have been proposed.

MRF modelling has appealing features for problems that involve the optimization of a configuration of variables that have interdependencies among them. Accordingly, MRFs allow the incorporation of contextual information in a principled way. MRFs rely on a strict probabilistic modelling, yet they allow the incorporation of prior knowledge by means of potential functions. For those reasons, in this paper we adopted an MRF model for refining the initial ranking of a set of documents retrieved by some IR system. The rest of this sections summarizes the formalism of MRFs.

An MRF is a set of random variables $F = \{f_1, ..., f_N\}$ indexed by sites or nodes where the following conditions hold:

$$P(f_i) \geq 0, \forall f_i \in F \tag{1}$$

---

[7] In the context of the Relevance Feedback track from INEX, we were given as feedback *relevant passages* instead of full documents though the Evaluation Platform.
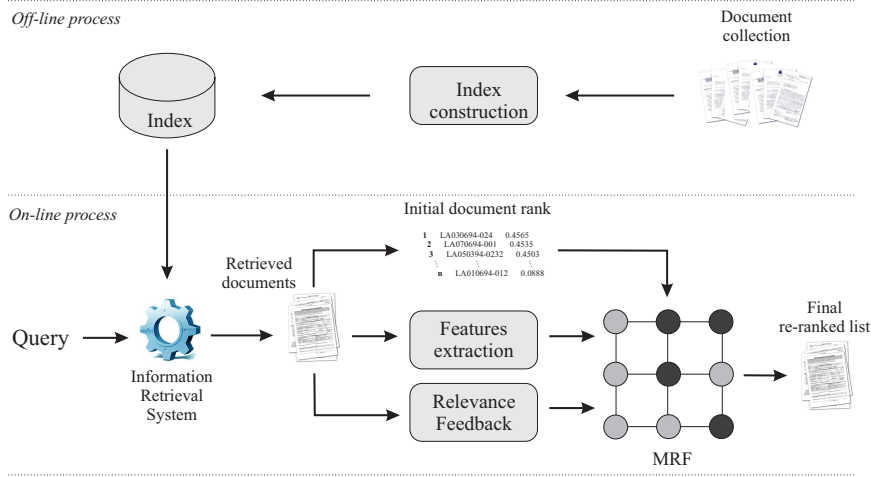
Fig. 1. Block diagram of the proposed ranking refinement method employed in the INEX 2012

$$P(f_i|f_{S-\{i\}}) = P(f_i|\mathcal{N}(f_i)) \tag{2}$$

where $\mathcal{N}(f_i)$ is the set of neighbours of f_i according to the neighbouring system $\mathcal{N}$. Formula 1 is the so called positivity condition and avoids negative probability values, whereas expression 2 states that the value of a random variable depends only on the set of neighbours of that variable.

It has been shown that an MRF follows a Gibbs distribution [3], where a Gibbs distribution of the possible configurations of F with respect to $\mathcal{N}$ has the following form:

$$P(F) = Z^{-1} \times e^{-\frac{1}{T}E(F)} \tag{3}$$

where Z is a normalization constant and the T is the so called temperature parameter (a common choice is $T = 1$) and $E(F)$ is an energy function of the following form:

$$E(F) = \sum_{c \in C} V_c(f) = \sum_{\{i\} \in C_1} V_1(f_i) + \sum_{\{i,j\} \in C_2} V_2(f_i, f_j) + \dots \tag{4}$$

where "$\dots$" denotes possible potentials V_c defined over higher order neighbourhoods $C_3, C_4. \dots, C_K$; each C_i defines a neighbourhood system of order i between the nodes of the MRF. Often the set F is considered the union of two subsets of random variables $X \cup Y$; where X is the set of observed variables and Y is the set of output variables, which state we would like to predict. Potentials V_c are problem dependent and commonly learned from data.

One of the main problems in MRFs is that of selecting the most probable configuration of F (*i.e.,* an assignment of values to each variable f_i of the field). Such configuration is determined by the configuration of F that minimizes expression 4, for which a diversity of optimization techniques have been adopted [5, 9, 6, 16].

2.2 Proposed Model

In our case we consider a MRF in which each node corresponds to a document in the list. Each document is represented as a random variable with 2 possible values: *relevant* and *irrelevant*. We consider a fully connected graph, such that each node (document) is connected to all other nodes in the field; that is, we defined a neighbourhood scheme in which each variable is adjacent to all the others. Given that the number of documents in the list is relatively low (100, 300 and 1000 in the experiments), to consider a complete graph is not a problem computationally, and allows us to consider the relations between all documents in the list.

For representing the documents, and evaluating the internal and external similarities, we consider all the words contained in each document (except stopwords), it is worth mentioning that we did applied a stemming process to all documents. To describe the documents we used a binary bag of words (BOW) representation, in which each vector element represents a word from the collection vocabulary; and the *example texts* are represented in the same manner. The internal and external similarities are considered via the energy function described next.

2.3 Energy Function

The energy function of the MRF combines two factors: the similarity between the documents in the list (*internal* similarity); and external information obtained from the original order and the similarity of each document with the provided feedback (*external* similarity). The internal similarities correspond to the interaction potentials and the external similarities to the observation potentials. The proposed energy function takes into account both aspects and is defined as follows:

$$E(F) = \lambda V_c(f) + (1 - \lambda)V_a(f) \tag{5}$$

Where V_c is the interaction potential and it considers the similarity between random variable f and its neighbours, representing the support that neighboring variables give to f. V_a is the observation potential and represents the influence of external information on variable f. The weight factor λ favours V_c ($\lambda > 0$), V_a ($\lambda = 0$), or both ($\lambda = 0.5$).

V_c is defined as:

$$V_c(f) = \begin{cases} \bar{Y} + (1 - \bar{X}) \ if \ f = irrelevant \\ \bar{X} + (1 - \bar{Y}) \ if \ f = relevant \end{cases} \tag{6}$$

Where $\bar{Y}$ represents the average distance between variable f and its neighbours with irrelevant value. $\bar{X}$ represents the average distance between variable f and its neighbors with relevant value. The distance metric used to measure the similarity between variables is defined as: $1 - dice(f, g)$, where $dice(f, g)$

represents the *Dice* coefficient [8], and is defined as: $dice(f,g) = \frac{2|f \cap g|}{|f \cup g|}$. V_a is defined as follows:

$$V_a(f) = \begin{cases} (1 - dist(f,e)) \times g(posinv(f)) \; iff = irrelevant \\ \\ dist(f,e) \times g(pos(f)) \qquad\qquad iff = relevant \end{cases} \tag{7}$$

The V_a potential is obtained by combining two factors. The first indicates how similar, $dist(f,e)$, or different, $1 - dist(f,e)$ is the f variable with the *example texts* (e) (*i.e.*, the information provided by the feedback). Where $dist(f,e)$ is defined as: $1 - dice(f,e)$. The second is a function that converts the position in the list given by a base IR machine to a real value. The function used $g(x) = exp(x/100)/exp(5)$ [2][8]. The function $pos(f)$ returns the position of the document f in the original list, $posinv(f)$ returns the inverse position of the f variable in this list.

Having described each potential, the proposed energy function is defined as:

$$E(F) = \begin{cases} \lambda \bar{Y} + (1 - \bar{X}) + (1 - \lambda)[1 - dist(f,e)) \times g(posinv(f)] \; iff = irrelevant \\ \\ \lambda \bar{X} + (1 - \bar{Y}) + (1 - \lambda)dist(f,e) \times g(pos(f)) \qquad\qquad iff = relevant \end{cases} \tag{8}$$

The initial configuration of the MRF is obtained by relevance feedback. That is, the subset of documents that contain relevant passages selected via relevance feedback are initialized as relevant, and all other documents as irrelevant. Then, the MRF configuration of minimum energy (MAP) is obtained via stochastic simulation using the ICM algorithm. At the end of this optimization process, each variable (document) has a value of relevant or irrelevant. Based on these values, a new re-ordered list is produced, by positioning first the relevant documents according to the MRF, and then the not-relevant ones.

3 Experimental Setup

In this section we describe the experimental setup that we employed for the proposed method during the INEX 2012 competition. A brief description of the base IR system used is given as well as its configuration. Besides this, we describe an additional ranking refinement strategy employed in our submitted runs, as well as the documents collection and the evaluation measures.

3.1 Base IR System

As we have mentioned before, our ranking refinement strategy does not depend on any particular IR system. However, in order to perform our experiments

---

[8] The intuitive idea of this function is such that it first increases slowly so that the top documents have a small potential, and then it increases exponentially to amplify the potential for those documents in the bottom of the list.

we employed as base IR system the well known information retrieval system LEMUR-INDRI. This system is part of the Lemur Project[9] started in 2000 by the Center for Intelligent Information Retrieval (CIIR) at the University of Massachusetts, Amherst, and the Language Technologies Institute (LTI) at Carnegie Mellon University. Particularly the LEMUR-INDRI toolkit is a search engine that provides state-of-the-art text search facilities, a rich structured query language for different text collections, and is considered as a robust system capable of producing comparable results to new IR schemes.

For all our experiments, the collections were indexed by this tool using a probabilistic language model. For this purpose, collections were preprocessed by applying stop word elimination as well as a stemming process. For our experiments we employed a list of 571 stop words available in the CLEF site[10]. Additionally, for the stemming process we employed the well known Porter algorithm [10].

As baseline results we considered the performance obtained under this configuration employing the LEMUR-INDRI search engine.

3.2 Query Expansion via Relevance Feedback

A query expansion via relevance feedback process is a controlled technique which main goal is to reformulate a query. In other words, a relevance feedback strategy is normally a previous step for a query expansion (QE) process. The basic idea is to select a set of k words which are related to a set of documents that have been previously retrieved and tagged as relevant by some user. Further, this words are added to the original query [11]. In order to apply a relevance feedback process it is necessary to perform a first search (*i.e.,* a first retrieval process) which generates an ordered list of documents. Afterwards, the user selects from the first positioned documents those that he considers as relevant (*i.e.,* the user establishes the documents' relevance). This relevance judgements that the user just gave to the documents are employed to compute a new set of values that indicate in a more accurate form the impact of each word in the original query[11].

As an alternative solution, we performed some experiments applying a QE process. For this, every time some feedback was given, we selected the k most frequent/less frequent words for its addition to the original query in order to perform a new retrieval process. Among the disadvantages of QE is the computational cost implied, since it is necessary to perform a second retrieval process. Besides this, relevance feedback strategies have shown to be sensitive to the quality of the added words, since adding an *irrelevant word* could be very harmful for the IR system

---

[9] http://www.lemurproject.org
[10] Cross Language Evaluation Forum (http://www.clef-campaign.org/).
[11] When the relevant documents are identified by some automatic process, it is assumed that documents placed at the top positions of the list are in fact relevant, and the new set of words that will be added to the query are automatically selected; this type of feedback is known as *blind relevance feedback*.

3.3 Data set

In the framework of the INEX 2012 Relevance Feedback track we were provided with the Wikipedia XML Corpus as the test collection. This collection was created from the October 8, 2008 dump of the English Wikipedia articles, and contains 2,666,190 articles, which represent more that than 50 GiB of disk space.

3.4 Evaluation

The evaluation of results was carried out using a measure that has demonstrated its pertinence to compare IR systems, namely, the Mean Average Precision (MAP). MAP is defined as follows:

$$MAP = \frac{1}{|Q|} \sum_{i=1}^{|Q|} \left(\frac{\sum_{r=1}^{m} P_i(r) \times rel_i(r)}{n} \right)$$

Where $P_i(r)$ is the precision at the first r documents, $rel_i(r)$ is a binary function which indicates if document at position r is relevant or not for the query i; n is the total number of relevant documents for the query i, m is the number of relevant documents retrieved and Q is the set of all queries.

Intuitively, this measure indicates how well the system puts into the first positions relevant documents. It is worth pointing out that since our IR system was configured to retrieve 1000 documents per query, MAP values are measured at 1000 documents.

On the other hand, $P@N$ is defined as the percentage of retrieved relevant items at the first N positions of the result list. Finally $R - Prec$ is defined as the precision at R-th position in the ranking of results for a query that has R relevant documents.

3.5 Experiments definition

The use-case of the INEX 2012 relevance feedback track is as follows: assume a single user searching with a particular query in an information retrieval system that supports relevance feedback. The user highlights relevant passages of text in returned documents (if any) and provides this feedback to the information retrieval system. The IR system re-ranks the remainder of the unseen results list in order to provide more relevant results to the user.

Accordingly, we conducted a series of experiments with the following objectives: $i)$ to test the results of the proposed method compared with the traditional re-ranking strategies, $ii)$ to evaluate the sensitivity of the method to the model parameters.

We defined 10 different configurations, which are described below:

- BASE-IND: represents the experiment performed using just the INDRI IR machine. For this experiment, if some feedback is provided, the feedback is ignored and the next retrieved document is showed.

– BASE-IND-QE20tMF: this experiment was performed using a QE strategy as re-ranking method. Once an initial documents list is provided by INDRI, the system keep delivering documents until some feedback is provided. If some feedback occurs, our systems reformulates the original query adding the 20 most frequent terms contained in the feedback passages and applies a new retrieval process. The new retrieved documents list is then showed to the user. This procedure is repeated every time some feedback occurs.

– BASE-IND-QE20tLF: this configuration works in a similar form to the previous experiment, although the only difference is that we reformulate the original query by adding the 20 less frequent terms.

– BASE-IND-QE20tMFandLF: this configuration works in a similar form to the previous experiment, although the only difference is that we reformulate the original query by adding the 20 most frequent and the 20 less frequent terms.

– RRMRF-xxxD-Lxx: these experiments represent the runs that employed our proposed markov random field as re-ranking strategy. The first three x's represent the number of documents that were used to construct the field, whereas the second x's represent the lambda (λ) parameter value. This configuration works as follows: once we have retrieved an initial list of documents using INDRI, our system keeps delivering documents until some feedback is provided. If some feedback occurs, our proposed method constructs a virtual *example text* (e) employing all the information contained in the feedback and marks as relevant those documents that provided the feedback. After the iteration process we show to the user the next relevant document. This process repeats every time some feedback is provided.

4 Results

Table 1 shows the evaluation results from all the submitted runs by our team. It is important to mention that the INEX 2012 Relevance Feedback track employed a new methodology for submitting results. During this campaign, participant teams were provided with an Evaluation Platform (EP) that worked as an online tool for providing the queries as well as for providing the feedback (if any) for every document showed to the EP. A total of 50 queries were processed, hence, Table 1 shows the average results obtained across the 50 queries.

Notice that even when the baseline configuration does not obtained a very high performance, obtained results are among the best performances. Remember that the baseline method means that we are using only the output produced by the INDRI IR machine. Therefore, obtained results indicate that INDRI was not able to retrieve a significant number of relevant documents, resulting in low recall levels.

A preliminary analysis indicate us that the configuration of our IR machine was not the most adequate for the type of queries that we processed. Most of the queries consisted on a set of general terms that do not necessarily represent a query formulated in natural language. We believe that using a boolean or a

Experiment	MAP	R-Prec	P@5	Recall
BASE-IND	**0.1015**	**18.28%**	45.60%	**25.93%**
BASE-IND-QE20tMF	0.0775	13.96%	39.60%	21.25%
BASE-IND-QE20tLF	0.0395	7.18%	30.80%	5.61%
BASE-IND-QE20tMFandLF	0.0728	13.64%	39.20%	20.03%
RRMRF-100D-L0.3	0.0940	16.12%	44.80%	16.40%
RRMRF-100D-L0.5	0.0946	15.95%	45.20%	16.40%
RRMRF-300D-L0.3	0.1002	17.69%	45.20%	22.76%
RRMRF-300D-L0.5	0.1004	18.05%	**46.00%**	22.76%
RRMRF-1000D-L0.3	**0.1015**	18.24%	45.60%	**25.93%**
RRMRF-1000D-L0.5	**0.1015**	18.24%	45.60%	**25.93%**

Table 1. Official Evaluation results obtained in the framework of the INEX 2012 Relevance Feedback track

vectorial model instead of a probabilistic one could provide better results in terms of the *recall* measure.

As can be observed, results obtained by the configurations that employed a query expansion strategy as re-ranking mechanism obtained the worst set of results. This indicate that terms considered during the query reformulation were somehow irrelevant even when they were provided by an user.

Finally, notice that our proposed method it is able to provide a better ranking when using 300 documents and lambda 0.5, since it provides better results at the first five positions of the final list ($P@5$). In general, we can observe that using few documents and a high value of lambda our method is able to produce acceptable results (almost similar to those obtained when using 1000 documents). however, the main limitation of our system was the INDRI initial bad performance (low *recall* values). It is important to mention, as established in [14], the proposed Markov random field depends on having high *recall* levels.

5   Conclusions

This paper proposed a method for improving the ranking of a list of retrieved documents by a IR system. Based on a relevance feedback approach, the proposed method integrates the similarity between the documents in the list (*internal* similarity); and external information obtained from the original order, the query and the provided feedback (*external* similarty), via a MRF to separate the relevant and irrelevant documents in the original list.

Experiments were conducted in the framework of the INEX 2012 Relevance Feedback track. For our experiments we avoid using any specialized external resources, since we were interested in evaluating the pertinence of the method employing only textual (document's words) features. Results showed that considering few documents and providing more importance to the internal similarities among documents, the proposed method is able to reach an acceptable performance. An initial analysis indicates that for this collection, it is necessary to

employ as IR method a traditional boolean or vectorial model in order to improve the recall levels of the IR machine, which is an important condition for the proposed method to work properly.

References

1. Baeza-Yates R., and Ribeiro-Neto B. (1999) *Modern Information Retrival.* Addison Wesley.
2. Chávez O., Sucar L. E., and Montes M. (2010). Image Re-ranking based on Relevance Feedback Combining Internal and External Similarities. In *The FLAIRS Conference*, Daytona Beach, Florida, USA.
3. Geman S., and Geman D. (1987) Stochastic Relaxation, Gibbs Distribution, and the Bayesian Restoration of Images. In *Readings in Computer Vision: Issues, Problems, Principles, and Paradigms.* pp. 564-584.
4. Grossman D. A., and Frieder O. (2004) *Information Retrieval, Algorithms and Heuristics.* Springer,2nd edition.
5. Kemeny J., Snell J.L. and Kanpp A.W. (1976) Denumerable Markov Chains. New York-Heidelberg-Berlin, Springer Verlag.
6. Lauritzen S. L. (1996). Graphical Models. Oxford University Press, New York NY.
7. Li S. Z. (2001) Markov Random Field Modeling in Image Analysis. 2nd. Edition, Springer.
8. Mani, I. (2001). Automatic Summarization. In *Natural Language Processing*, Vol. 3 . John Benjamins Publishing Co.
9. Pearl J. (1988) Probabilistic Reasoning in Intelligent Systems. Morgan Kaufman, San Mateo CA.
10. Porter M. F. (1997) An Algorithm for suffix stripping. Morgan Kaufman Publishers Inc. pp. 313-316.
11. Salton G., and Buckley C. (1990) Improving Retrieval Performance by Relevance Feedback. In *Journal of the American Society for Information Science*: 41(4), pp. 288-297
12. Villatoro-Tello E. , Montes-y-Gómez M., and Villaseñor-Pineda L. (2009) A Ranking Approach based on Example Texts for Geographic Information Retrieval. In *Post-proceedings of the 9th Workshop of the Cross Language Evaluation Forum CLEF 2008.* Vol. 5822, pp. 239-250. Lecture Notes in Computer Science. Berlin: Springer-Verlag.
13. Villatoro-Tello E., Villaseñor-Pineda L., and Montes-y-Gómez (2009) M. Ranking Refinement via Relevance Feedback in Geographic Information Retrieval. In *Mexican International Conference on Artificial Intelligence MICAI 2009.* Vol. 5845, pp. 165-176. Lecture Notes in Computer Science. Berlin: Springer-Verlag.
14. Villatoro-Tello E. , Juárez-González A., Montes-y-Gómez M., Villaseñor-Pineda L., and Sucar E. L. (2012) Document Ranking Refinement Using a Markov Random Field Model. In *Journal of Natural Language Engineering* Volume 18, issue 02, pp. 155-185
15. Yang L., Ji D., Zhou G., Nie Y., and Xiao G. (2006) Document Re-ranking Using Cluster Validation and Label Propagation. In *Proceedings of the 2006 ACM CIKM International Conference on Information and Knowledge Management.* pp. 690-697.
16. Winkler G. (2006) Image Analysis, Random Fields and Markov Chain Monte Carlo Methods. *Springer Series on Applications of Mathematics*, 27, Springer.