December 2003

Improving Document Representation by Accumulating Relevance Feedback: The Relevance Feedback Accumulation (RFA) Algorithm

Razvan Bot
New Jersey Institute of Technology

IMPROVING DOCUMENT REPRESENTATION BY ACCUMULATING RELEVANCE FEEDBACK: THE RELEVANCE FEEDBACK ACCUMULATION (RFA) ALGORITHM

Razvan Stefan Bot
New Jersey Institute of Technology
rsb2@njit.edu

Introduction

Relevance Feedback (RF) is a widely used technique in Information Retrieval (IR). Using relevance feedback, the searcher feeds back query-document pair relevance assessments into the system. Since its inception, RF was identified as the most important way to improve effectiveness of retrieval systems. Generally speaking, there are two main ways to use relevance feedback. One way, called query expansion, is to modify the initial query posed by the searcher. Then new terms/concepts are added to the initial query in order to improve it. The improved query supposedly represents a more accurate representation of the searcher's actual information need. The second way of using relevance feedback is to alter the document representation. By doing this, weights of terms from the query are increased in those documents assessed relevant by the searcher. This technique creates a dynamic document representation space, where weights of terms/concepts may fluctuate.

Most of the literature is concentrating in exploiting query expansion. One drawback of query expansion is that the mechanism is not able to capture relevance feedback across searchers, over time. To achieve this, a retrieval system must provide another mechanism that, over time, accumulates feedbacks into the permanent document representation. This corresponds to the second way of using relevance feedback: document space transformations.

This paper presents **WISEearch** proposal, an information retrieval system that accumulates relevance feedback in time and across users. The algorithm is inspired by early work in this direction done by Friedman et al. (1967), Brauen et al. (1968), Brauen (1969) and Ide (1969). The novelty is the use of relevance assessment history in order to derive concepts that best characterize documents. The procedure uses ***support***, a data mining measure, to identify these concepts. The whole idea stands behind the following main assumptions:

(a) *Every document in the searchable document space is best characterized by a small set of terms.*
(b) *The terms characterizing a document might or might not occur in that document.*
(c) *Queries consist of short natural language statements. Maximum 25 words.*

The rest of this document presents a short overview of the Relevance Feedback Accumulation (RFA) algorithm, evaluation design and research contributions.

Methodology—THE RFA Algorithm

The goals of RFA algorithm are: **(1)** to reduce the dimensionality of document representation space. Less index terms per document; **(2)** improve document representation. Better terms are used to index each document. These terms are called concept terms; and **(3)** improve retrieval effectiveness as an immediate effect of the first two goals.

RFA algorithm defines two types of terms:

(a) **Single term**: a term consisting of only one word. Example: "*information*" or "*retrieval*".
(b) **Composite term**: a term consisting of two words. A good example is the term "*information retrieval*" which has a more powerful semantic meaning than the single terms "*information*" and "*retrieval*" considered separately.

RFA algorithm is based on the assumption that every document is best characterized by a set of few terms. These terms are called concepts. A concept then is a term that represents the topicality of a document. As a note: it is not necessary that a concept can be found within the document for which it stands as a concept. The difficult part is to identify these concepts. Most of the automatic indexing algorithms are focusing on finding the importance of terms within documents by calculating frequency-based measures. This is also called lexicographic analysis. Their direct assumption is that if a term occurs several times within a document that term might be a concept in the document. This, of course, is not always the case. Also, the above-referred automatic indexing algorithms cannot support the case of a concept term that does not appear throughout the document. RFA algorithm creates and maintains a dynamic document representation space as a response to the above problems. The importance of concepts is not given any more by a simple lexicographic analysis of the documents' content. The importance of terms it is derived from RF assessments in time and across users. A document concept in this context is identified as a term that has a reasonable **support** among all queries from all relevance assessments of this document. Support is a data mining measure emphasizing the occurrence percentage of an item in a set of transactions. In this case a query term is considered as an item while the query is considered to be the transaction. See Table 1 for an example.

The important thing to mention is that the concepts discovery is user-driven. In time the concepts from each document will reflect the user's general perception regarding which are the most important terms to describe the document. The immediate logic augmentation RFA makes is to eliminate terms with low support in a document from that document's representation. In this way the dimensionality of each document's representation is reduced.

Table 1. SUPPORT Measure Exemplification

Example
Suppose document D was assessed as relevant to the following queries: – Q1(term_1, term_2, term_3) – Q2(term_1, term_2) – Q3(term_1, term_4) and – Q4(term_1, term_2, term_3) In this case the support for each of the terms is: – support(term_1)=4/4 ➔ 1.0 – support(term_2)=3/4 ➔ 0.75 – support(term_3)=2/4 ➔ 0.5 and – support(term_4)=1/4 ➔ 0.25 The interpretation is that if we set the support threshold at 0.5 than term_1 and term_2 can be considered concept terms because appeared in more than 50% of the queries to which D was judged as relevant. term_3 and term_4 are not considered concept terms. They are most probably accidental terms due to the user's inability to accurately formulate his/her information need.

Before presenting the overview of the RFA algorithm one must emphasize the way terms are generated from queries. Let's consider **Q(term_1, term_2, term_3, term_4)** be a query composed of four terms. The single terms of **Q** are: **term_1**, **term_2**, **term_3** and **term_4**. The composite terms are derived from the original query **Q** by using a heuristic called ***ordered terms pairing***. This heuristic takes all pairs of single terms from a query (maintaining their ordering in the query), and builds composite terms by aggregating them. In the case of **Q** considered above the heuristic will generate the following composite terms: **[term_1, term_2]**, **[term_1, term_3]**, **[term_1, term_4]**, **[term_2, term_3]**, **[term_2, term_4]** and **[term_3, term_4].** See example in Table 2. After generating the composite terms, Q is considered to be composed by the union of all single and composite terms. The rationale behind the idea is the fact that many times individual words (single terms) do not provide enough meaning with respect to the topicality of a document. From this point on the notion ***term*** denotes both single and composite terms, treated similarly by the RFA algorithm.

Table 2. Ordered Terms Pairing Heuristic Illustration

Example
Let's consider the query **Q(data mining algorithms)**. The terms extracted from this query are: (1) Single terms: **data**, **mining** and **algorithms** (2) Composite terms: **data mining**, **data algorithms** and **mining algorithms** So the query is considered to have contained 6 terms: *data*, *mining*, *algorithms*, *data mining*, *data algorithms* and *mining algorithms*.

The RFA generic algorithm consists of two steps:

STEP 1: the document term modification takes place for each document whenever a new relevance feedback assessment is available. Suppose (Q, D) represents a relevance assessment. This means that document D was judged as being relevant to query Q. There are two cases:

Case A: when a term occurs in the query Q but not in the document D, this term is introduced as a new indexing term for document D. The term is given an initial weight. In this way new terms, potential concepts, are introduced as indexing terms for document D.

Case B: when a term occurs in both the query Q and the document D, the weight of the term is modified according to its support with respect to D.

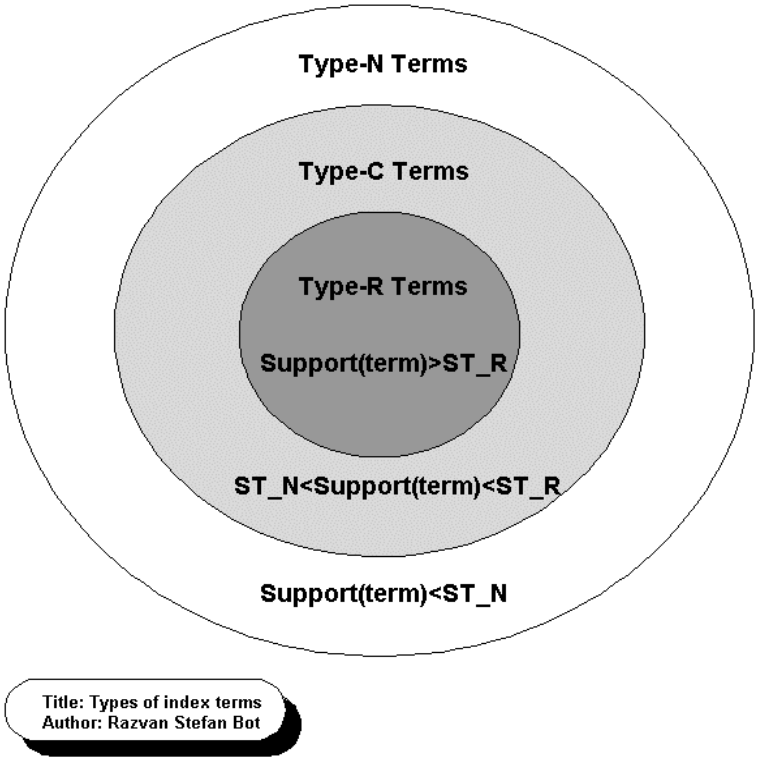

Figure 1. Types of Indexing Terms

STEP 2: after a predetermined number of relevance judgments are available for a document D, the terms characterizing D are re-classified in three type categories according to their support values (see Figure 1):

- **Type R** terms: relevant terms, having high support.
- **Type C** terms: candidate terms, having moderate support.
- **Type N** terms: non-relevant terms, having low support.

The type N terms will not be considered indexing terms anymore but they will still be kept in the data structure, because they might increase their support with future relevance assessments and become C or R type terms again.

Figure 2 illustrates the RFA algorithm. One can notice that the concepts to represent document D are discovered by analyzing/mining the available relevance feedback for this document. The RFA algorithm maintains a database where relevance feedback assessments are accumulated. In this way, fewer and better concepts are discovered for each document, using prior relevance feedback history.

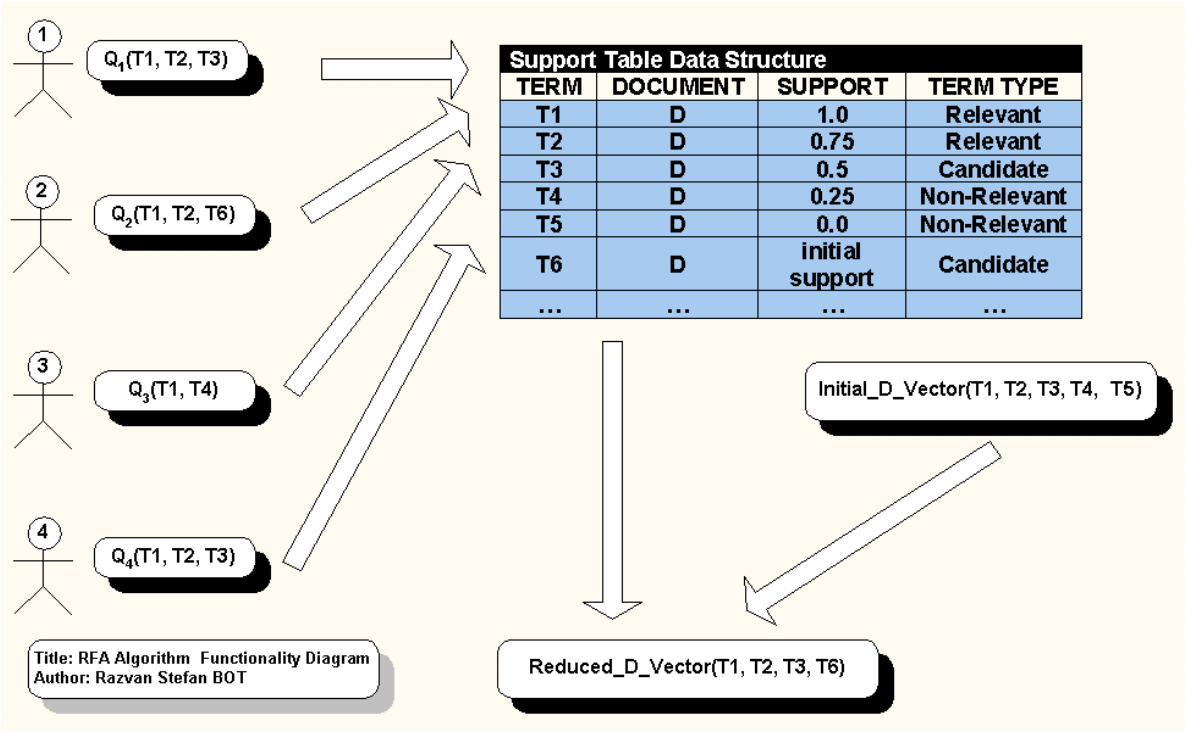

Figure 2. RFA Algorithm Functionality Diagram

Evaluation

Evaluation is done by comparing results achieved on the TREC collection, while using RFA algorithm, with two other IR systems: **(1)** a standard vector space (VSM) retrieval system and **(2)** a standard VSM retrieval system augmented with Brauen's (1969) algorithm. The three evaluation measures are: **(a)** the average number of index terms per document, **(b)** the quality of the derived document index terms and (c) retrieval effectiveness in terms of average precision and recall.

Contributions

The research contributions made to the field of Information Retrieval by this thesis proposal are:

(a) Revitalizes a useful topic, Document Representation Modification, a potentially valuable technique that was given little attention since its inception. The technique was proven to improve retrieval effectiveness by Friedman et al. (1967), Brauen et al. (1968), Brauen (1969) and Ide (1969)

(b) Presents a novel dimensionality reduction technique that is implemented by the RFA algorithm. Many other techniques were previously used to reduce the number of terms to index a document. Stemming for example reduces the dimensionality of the representation space by grouping words/terms into synonymy-based equivalence classes. Still many of these equivalence classes are not content bearing with respect to the topicality of documents. Another acclaimed method is Latent Semantic Indexing (LSI) developed by Deerwester (1990). This technique performs complex mathematical/statistical analysis on the whole document collection in order to come up with a reduced dimension document vector for each document. The technique might be useful but it is totally inefficient. The time complexity is too large and it is not suited at all for indexing large document collections.

(c) Presents a novel way of identifying the indexing terms for documents. Good indexing terms identification was a problem since the inception of Information Retrieval. It is hard to find those terms/words in a document (or not in the document) that best characterize that document's topicality. Previously, almost all techniques were based on automatic indexing by lexicographic analysis. In this case the importance of a word in a document, is solely assessed by computing frequency-based measures normalized in different ways. But there is no proven theory stating that the frequency of a word in a document is related to its importance in that document. That is why many of the terms generated by standard lexicographic analysis are of no use. RFA algorithm identifies the concepts to represent documents from the users' point of view. The words that are used most of the time by users to describe a document are considered to be concept terms in that document, and they are picked as indexing terms. The source of evidence for this user-view analysis is given by the relevance feedback history, accumulated by the retrieval system in special data structures.

Simple and Efficient Algorithm. RFA is an algorithm based on a simple idea. Deriving concept terms from users' point of view is straightforward and it does not need complex meaningless mathematical models to be formalized. At the same time, the algorithm is suited for retrieval systems governing large document collections. The overhead generated by implementing RFA on top of a standard VSM retrieval system is minimal.

References

Brauen, T. L., R. C. Holt, et al. (1968). "Document Indexing Based on Relevance Feedback." *Report ISR-14 to the Ntional Science Foundation*, Section XI, Department of Computer Science, Cornell University, Ithaca, NY(June).

Brauen, T. L. (1969). "Document Vector Modification." *Scientific Report ISR-17*(September).

Deerwester, S., S. T. Dumais, et al. (1990). "Indexing by latent semantic analysis." *Journal of the American Society for Information Science* **41**(6): 391-407.

Friedman, S. R., J. A. Maceyak, et al. (1967). "A Relevance Feedback System Based on Document Transformations." *Scientific Report* **ISR-12**(June): Section X.

Ide, E. (1969). "Relevance Feedback in Automatic Document Retrieval System." *Report ISR-15 to the National Science Foundation*, Cornell University, Ithaca, NY(January): 81-85.