# An Homogeneous Framework to Model Relevance Feedback

David E. Losada
AILab, Department of Computer Science
A Corunna, Spain
losada@dc.fi.udc.es

Alvaro Barreiro
AILab, Department of Computer Science
A Corunna, Spain
barreiro@dc.fi.udc.es

## ABSTRACT

Relevance feedback is an appreciated process to produce increasingly better retrieval. Usually, positive feedback plays a fundamental role in the feedback process whereas the role of negative feedback is limited. We think that negative feedback is a promising precision oriented mechanism and we propose a logical framework in which positive and negative feedback are homogeneously modeled. Evaluation results against small test collections are provided.

## 1. INTRODUCTION

Classical feedback approaches tend to limit the impact of negative feedback. Consider a feedback cycle as follows. An original query fires a first retrieval. Consider that all the relevant documents in the top N do not mention a term $t$ whereas all the non-relevant documents deal with $t$. It seems reasonable to think that $t$ is a good representative of the non-relevant documents and, hence, we should move the original query away from $t$. However, this is not always possible in classical models. For instance, consider the previous example within the vector space model. All the relevant documents have weight 0 for $t$ and all the non-relevant documents have weights greater than 0 for $t$. If the original vector $\vec{q}$ has weight 0 for $t$ then the new query would have a negative weight for $t$. Since negative weights are considered as 0 weights we cannot move the new query away from $t$ but, on the contrary, the best we can do is to say that we do not care about $t$. One could argue that this is not a feasible case because non-relevant documents are inherently more heterogeneous than relevant documents and, thus, it is difficult to extract good representatives for the set of unrelevant documents. Nevertheless, we can think on the set of unrelevant documents as a set of clusters of documents, each cluster dealing with a number of topics. Hence, if we move the query away from a given non-relevant document $d$ then other non-relevant documents in the same cluster than $d$ will likely be moved down in the rank.

Belkin et al. [1] interpreted negative feedback as the selection of important terms in non-relevant documents and showed improvements in performance in interactive IR. Hoashi and his colleagues [3] have recently claimed the importance of negative feedback in the context of filtering. Their approach is based on using a positive profile whose output is filtered by a negative profile. We propose a logical model in which documents and queries are represented as Propositional Logic formulas and the feedback process is formalized as a Belief Revision (BR) process. Logic allows us to handle positive and negative feedback in an homogeneous way.

## 2. MODEL

Propositional logic allows us to model binary-weighted vectors, e.g. $d = information \wedge science \wedge \neg maths$, but more expressive representations can also be handled, e.g. $d = (relevance \wedge feedback) \vee (document \wedge filtering)$. In order to measure the relevance of a document $d$ to a query $q$, we use the method proposed in [4] to get a non-binary measure of the entailment $d \models q$. An important circumstance is that this model was efficiently implemented [5] and, furthermore, evaluation against small collections was made [6].

We focus on a feedback process based on selecting terms from retrieved documents. Term selection approaches [2] have shown that expanding the query with well-selected terms produces significant improvements in performance. Basically, all the terms from retrieved relevant documents are collected and ordered by a given sorting technique. Top ranked terms are supposedly the important ones within the set of relevant documents and, thus, the query is expanded with these terms. Classical term selection techniques only operate on the retrieved relevant documents. We propose to select terms on the retrieved non-relevant documents as well.

Let us consider an initial logical query $q$ that retrieves a set of documents. We propose to revise the query using selected terms from relevant documents as positive terms and selected terms from unrelevant documents as negative terms. We build a revising formula which is the conjunction of all the terms, either positive or negative: $q_m = q \circ (t_{p1} \wedge \cdots \wedge t_{pn} \wedge \neg t_{n1} \wedge \cdots \wedge \neg t_{nm})$, where $t_{p1}, \ldots, t_{pn}$ are the selected terms from the retrieved relevant documents, $t_{n1}, \ldots, t_{nm}$ are the selected terms from the retrieved non-relevant documents and $\circ$ is a BR operator. Let $T$ be a logical theory and $A$ a new formula to be included in the theory. BR methods define a way to include the new information in the theory. If there is no contradiction between $T$ and $A$, the solution to the problem is trivial because the new


*SIGIR'01,* September 9-12, New Orleans, Louisiana, USA..

| R \ P | BR | P | PN | BR | P | PN |
|---|---|---|---|---|---|---|
| 0.00 | 0.320 | 0.376 | 0.422 | 0.319 | 0.408 | 0.463 |
| 0.10 | 0.254 | 0.311 | 0.351 | 0.298 | 0.383 | 0.438 |
| 0.20 | 0.190 | 0.212 | 0.237 | 0.236 | 0.343 | 0.363 |
| 0.30 | 0.151 | 0.173 | 0.195 | 0.182 | 0.254 | 0.292 |
| 0.40 | 0.103 | 0.109 | 0.135 | 0.144 | 0.203 | 0.227 |
| 0.50 | 0.084 | 0.097 | 0.114 | 0.128 | 0.182 | 0.215 |
| 0.60 | 0.070 | 0.072 | 0.091 | 0.084 | 0.130 | 0.145 |
| 0.70 | 0.051 | 0.053 | 0.058 | 0.060 | 0.092 | 0.102 |
| 0.80 | 0.040 | 0.041 | 0.045 | 0.050 | 0.080 | 0.083 |
| 0.90 | 0.022 | 0.019 | 0.023 | 0.042 | 0.072 | 0.071 |
| 1.00 | 0.016 | 0.015 | 0.017 | 0.041 | 0.071 | 0.070 |
| Avg.prec. | 0.118 | 0.134 | 0.154 | 0.144 | 0.201 | 0.224 |
| %chg | | +13.8% | +20.6% | | +40.1% | +56.1% |
| | | CACM | | | Cranfield | |

| R \ P | BR | P | PN | BR | P | PN |
|---|---|---|---|---|---|---|
| 0.00 | 0.412 | 0.413 | 0.523 | 0.127 | 0.241 | 0.220 |
| 0.10 | 0.201 | 0.221 | 0.242 | 0.060 | 0.120 | 0.179 |
| 0.20 | 0.161 | 0.168 | 0.183 | 0.047 | 0.045 | 0.075 |
| 0.30 | 0.136 | 0.138 | 0.147 | 0.020 | 0.018 | 0.014 |
| 0.40 | 0.115 | 0.117 | 0.119 | 0.012 | 0.013 | 0.008 |
| 0.50 | 0.102 | 0.103 | 0.102 | 0.009 | 0.010 | 0.006 |
| 0.60 | 0.086 | 0.088 | 0.085 | 0.006 | 0.007 | 0.005 |
| 0.70 | 0.073 | 0.075 | 0.069 | 0.005 | 0.005 | 0.004 |
| 0.80 | 0.063 | 0.065 | 0.057 | 0.004 | 0.004 | 0.003 |
| 0.90 | 0.050 | 0.050 | 0.043 | 0.003 | 0.003 | 0.003 |
| 1.00 | 0.038 | 0.039 | 0.035 | 0.003 | 0.003 | 0.003 |
| Avg.prec. | 0.131 | 0.134 | 0.146 | 0.027 | 0.043 | 0.047 |
| %chg | | +2.6% | +11.6% | | +59.3% | +76.9% |
| | | CISI | | | LISA | |

**Figure 1: Evaluation results**

theory $T \circ A$ is just $T \wedge A$. However, if contradiction arises some old knowledge (from $T$) has to be deleted in order to get a consistent new theory.

We regard the original query $q$ as a theory to be revised with the feedback information. Through $t_{p1} \wedge \cdots \wedge t_{pn}$, we are including information from relevant documents in the new query and, hence, relevant documents will have a good chance of being retrieved. On the other hand, $\neg t_{n1} \wedge \cdots \wedge \neg t_{nm}$ is used to reject non-relevant documents. The latter expression cannot be handled by classical models. An important point is that we developed an algorithm that computes the revision $q \circ (t_{p1} \wedge \cdots \wedge t_{pn} \wedge \neg t_{n1} \wedge \cdots \wedge \neg t_{nm})$ in polynomial time.

## 3. EVALUATING FEEDBACK

Since Propositional Logic is simple enough, we were able to extract logical representations for documents and queries applying classical techniques. Documents and queries from test collections are often divided into several subfields. Each subfield (after removing stopwords and stemming) is represented as a clause of a DNF formula[1]. This leads to a logical representation of documents and queries divided into several views. Intuitively, each subfield represents a different view of the semantics of the document/query.

We evaluated the feedback model against CACM, Cranfield, CISI and LISA. A residual evaluation methodology was applied. The top ten documents were used for relevance feedback. Not all the original queries can be considered for this evaluation because some of them retrieve all their relevant documents in the top ten and some of them retrieved no relevant documents in the top ten. Specifically we used 47 CACM queries, 190 Cranfield queries, 60 CISI queries and 19 LISA queries. We applied the postings method to select terms. The posting of a term is the number of relevant (non-relevant) documents in which it occurs.

Figure 1 presents the precision vs. recall figures. The first column of each table presents the base residual run (BR), i.e. the initial run without feedback and with the top ten documents removed. We tried out several expansions of the query varying the number of positive terms. In the second column (P) we show the results for the best run. Once the set of positive terms used for expansion is fixed, we expand the query with those positive terms and a set of negative terms. We varied the number of negative terms and, in the third column (PN), we show the performance results for the best run.

The selection of both positive and negative terms was the best approach in all collections. The improvements over the approach that only selects positive terms are remarkable.

## 4. CONCLUSIONS

It seems clear that a general framework that allows to model negative terms is desirable for the process of feedback. Our experiments have demonstrated that the use of negated terms in queries is very useful to reject non-relevant documents. The use of binary weights is not a particular restriction of the model. Indeed, we are now developing an extension of the model to handle term similarity and inverse document frequency.


## 5. ACKNOWLEDGMENTS

This work was partially supported by projects refs. PB97-0228 and PGIDT99XI10201B.


## 6. REFERENCES


[1] N. Belkin, J. Perez Carballo, C. Cool, S. Lin, S. Park, S. Rieh, P. Savage, C. Sikora, H. Xie, and J. Allan. Rutgers' TREC-6 interactive track experience. In *Proc. TREC-6*, pages 597–610, Gaithersburg, USA, November 1997.

[2] D. Harman. Towards interactive query expansion. In *Proc. ACM SIGIR-88*, pages 321–331, Grenoble, France, June 1988.

[3] K. Hoashi, K. Matsumoto, N. Inoue, and K. Hashimoto. Document filtering method using non-relevant information profile. In *Proc. ACM SIGIR-2000*, pages 176–183, Athens, Greece, July 2000.

[4] D. E. Losada and A. Barreiro. Using a belief revision operator for document ranking in extended boolean models. In *Proc. ACM SIGIR-99*, pages 66–73, Berkeley, USA, August 1999.

[5] D. E. Losada and A. Barreiro. Efficient algorithms for ranking documents represented as DNF formulas. In *Proc. ACM SIGIR-2000 Workshop on Mathematical and Formal Methods in Information Retrieval*, pages 16–24, Athens, Greece, July 2000.

[6] D. E. Losada and A. Barreiro. Rating the impact of logical representations on retrieval performance. In *Proc. DEXA'2001 Workshop on Logical and Uncertainty Models for Information Systems, LUMIS'2001 (to appear)*, Munich, Germany, September 2001.


[1] A DNF formula has the form: $c_1 \vee c_2 \vee \ldots$ where each $c_j$ is a conjunction of literals (also called *clause*): $l_1 \wedge l_2 \wedge \ldots$. A literal is a propositional letter or its negation.