Enhancing a logic-based information retrieval model using term similarity

David E. Losada
Depto. de Electrónica y Computación
Universidad de Santiago de Compostela, Spain
dlosada@dec.usc.es

Fabio Crestani
Dept. of Information and Computer Science
University of Strathclyde, Scotland
F.Crestani@cis.strath.ac.uk

Abstract

Information retrieval systems tend to adopt very simple representations for the elements involved in the retrieval process. We believe that highly expressive formalisms can play a role in future retrieval systems, provided that they are complemented by successful and efficient indexing and retrieval techniques. In this work, we follow a bottom-up approach in which starting from a very simplistic logic-based representation for documents, we get inspiration from popular retrieval techniques in order to articulate more expressive document representations. We show empirically that the performance of the logic-based system improves when more expressive representations are provided.

1 Introduction

Information Retrieval (IR) systems tend to index documents with simplistic representations which do not capture the documents' contents in a complete and accurate fashion. These simple indexing methods, although very efficient from a computational point of view, lead to awkward situations, such as the term mismatch problem between the vocabulary used in the documents and the terms expressed in the query. There has been broad range of techniques that have attempted to solve this problem, including dimensionality reduction, query expansion and imaging [4, 5, 3], to name a few. Inspired by the success of these approaches, we wonder whether these techniques could be applied in the context of a logic-based IR model for obtaining better logical document representations and, further, if this connection is of any help in terms of retrieval performance.

In this paper we are particularly concerned with expressive document representations and term similarity. Our main interest is to determine whether or not the performance of a logic-based approach improves as the system is provided with more expressive logical document representations. We implement and evaluate different methods based on term similarity for describing the texts as logical formulae. In this respect, the ease of logic to handle partial representations is fundamental for dealing with the term mismatch problem. Basically, a logic-based retrieval system can represent the connection between vocabulary terms and documents in a flexible way.

We use a logical model of IR based on Propositional Logic and Belief Revision (PLBR) [6] as the basis for our experiments. Different methods for creating logical document representations have been implemented and evaluated. The construction of the logical formulae is driven by term similarity, global term distribution and document length. These notions, which have been successfully applied in popular IR models, allow the design of a more evolved logical representational process that is grounded on mainstream IR methods.

The rest of this paper is organized as follows. The basic foundations of the logic-based model are briefly sketched in section 2. Section 3 presents the use of term similarity for constructing partial document representations. Experiments are reported in section 4 and the paper ends with some conclusions in section 5.

2 A logic-based information retrieval model

We have chosen the logic-based PLBR model [6] for conducting this study because it was efficiently implemented and evaluated against TREC-like collections in the past [7]. Retrieval performance results are therefore available and we can evaluate here the benefits obtained with the new techniques for deriving logical formulae from text. We are not interested in comparing the performance of this logic-based approach against other state of the art models. On the contrary, we will focus on the PLBR model trying to determine whether or not the production of more expressive logical formulae is beneficial in terms of retrieval performance. Logical models are still far from the state of the art IR approaches in terms of retrieval performance. We believe that this is mainly due to: 1) the limitations in dealing with some notions handled recurrently by most IR models (e.g. non-binary term frequency) and 2) the poor use of the expressiveness at hand. Only once these limitations are somehow overcome a comparison against popular IR methods will make sense. On the other hand, logical models of IR enable to study the IR process in more detail and to carry out powerful analysis of the behaviour of different models.

The PLBR model works with propositional formulae and estimates relevance by means of a non-binary measure of distance between the logical formula representing a document, d, and the logical formula representing the query, q. This measure is based on Belief Revision (BR) techniques that compute distance between logical formulae[1]. Classical vectors with binary weights can be represented in logical form straightforwardly. Further, propositional logic supplies the facility to deal with *partial* representations. This means that we do not have to store information about all the index terms when representing a document. This feature is not standard in regular IR systems. For instance, in the vector space model we have to assign the corresponding weight to every dimension (term) in the space.

3 Partial document representations

A simple way to handle terms that occur in the text of a document is to include them as positive literals[2] in the propositional formula representing the document. Regarding unseen terms, the PLBR model has provision for establishing a distinction between a term for which we do not know whether or not it is significant with respect to a given document's semantics and a term for which we have positive evidence that it is not related at all with the document's content. Terms falling into the latter case should be represented as negative literals and terms falling into the former case should be simply omitted in the document's representation. The more omissions we made, the *more partial* the representation becomes. The classification of the unseen terms into these two categories becomes therefore the fundamental issue in this process. In order to drive this selection, we take inspiration from popular techniques which have advanced the field of IR. First of all, the set of seen terms is a valuable information which should be taken into account. If we are to introduce some unseen terms as negations we should look at the similarity between the unseen terms and the document terms. It is intuitive to think that if an unseen term is highly similar to the document then the most sensible decision is to omit the term in the logical representation of the document. Since we have some evidence (from the similarity measure) about the connection between the unseen term and the document, we just take the conservative choice of omitting the term in the logical representation[3]. On the contrary, an unseen term having no connection at all with the document terms is a good candidate to be negated because there is no evidence on the relatedness between the term and the document.

This means that we introduce term similarity *at indexing time* in order to produce more expressive logical formulae. Term similarity has been

[1] We refer to [6] for a complete analysis on the BR techniques applied, including the characteristics of the logical matching process.

[2] A literal is a propositional letter or its negation, e.g. $music$ is a positive literal whereas $\neg tv$ is a negative one.

[3] One can even argue that these terms should be positive literals into the document representation (document expansion) but we go here for the most conservative choice.

applied successfully to support retrieval in the context of popular IR models. For instance, in the vector-space model, novel matching functions were proposed based on term similarity [2]. These functions not only account for query-document matching terms but the non-matching query terms contribute also to the final retrieval score, depending on their similarity to the document terms. In Language Modeling for IR, the translation models proposed in [1] compute translation probabilities between terms using also some form of term similarity. It seems therefore sensible to apply some term similarity techniques, which were successful in well-known IR models, to classify unseen terms as potential negations or omissions in the context of the logic-based model.

3.1 Term similarity

The problem of defining a measure of similarity between terms has been addressed by many researchers. We have used the *Expected Mutual Information Measure (EMIM)* because it has been used with success in the past and can be estimated using co-occurrence data [2]. Given two terms t_i and t_j, $EMIM(t_i, t_j)$ is often interpreted as a measure of the statistical information contained in t_i about t_j (or vice versa, it being a symmetric measure). Formally, the EMIM measure is defined as follows:

$$EMIM(t_i, t_j) = \sum_{t_i, t_j} P(t_i \in d, t_j \in d) log \frac{P(t_i \in d, t_j \in d)}{P(t_i \in d) P(t_j \in d)} \quad (1)$$

where t_i and t_j are any two terms of the term space T. Van Rijsbergen [10] proposed a method to estimate EMIM between two terms using co-occurrence data that can be derived by a statistical analysis of the term occurrences in the collection.

Once we have a term-to-term similarity measure we still need to define a term-to-document similarity measure, which will be used to estimate the similarity between an unseen term and the set of seen terms and, subsequently, make a decision about how to express the term into the logical representation of the document. To this aim, we have experimented with the two following term-to-document similarity functions:

$$f_1(t_i, d_j) = \sum_{t_k \in d_j} EMIM(t_i, t_k) \quad (2)$$

$$f_2(t_i, d_j) = idf(t_i) \cdot \sum_{t_k \in d_j} EMIM(t_i, t_k) \quad (3)$$

where $idf(t) = logN/df_t$ (N is the collection size and df_t is the document frequency of the term in the collection). The first function computes the similarity from a term to a document as the sum of the similarities across document terms. The second function is slightly more complex since it introduces an idf component for the involved term. Since the EMIM function is based on co-occurrence data, it can be the case that common terms are assigned a high value for most documents just because they tend to co-occur with many terms. Hence, f_2 tries to compensate this fact by promoting terms that are infrequent across the collection.

Given a document, the set of unseen terms can therefore be ranked in decreasing order of similarity to the document using either f_1 or f_2. This rank needs to be split into two parts and the most similar terms are omitted in the logical document representation and the remaining unseen terms are expressed as negated literals in the document representation. As it will be explained in the next section, we propose that the division of the rank of unseen terms is done taking into account document length.

3.2 Document length

The last decision to make concerns the number of unseen terms that should be negated. If we introduce the same number of negations for all documents in the collection we would be implicitly assuming that all documents had the same chance of mentioning explicitly all the relevant material. This assumption is not appropriate, as demonstrated by the success of different document length normalization techniques in the mainstream IR literature. A long document may simply cover more material than a short one. We can even think on a long document as a sequence of unrelated short documents concatenated together. This view is called the *scope hypothesis* and contrasts with the *verbosity hypothesis*, in which a long document is supposed to cover a similar scope than a short document but simply uses more words. It is accepted that the verbosity hypothesis prevails over the scope hypothesis and, indeed, the control of verbosity stands behind the success of high performance document length normalization techniques, which use to apply some form of penalization for long documents [9].

From a different perspective, the length of a document is a good indicator of how much we can trust the document's text to estimate its actual contents. For instance, this has been exploited by some Language Modeling approaches to obtain more accurate statistical estimations. In [11], the difference of data uncertainty in short and long documents is taken into account and, as documents are larger, the uncertainty in the estimations becomes narrower. A similar idea drives our logic-based approach because long documents are supposed to indicate more exhaustively their contents and, hence, more assumptions on the non-related terms will be taken. We follow a simple strategy in which, given a document text, the number of negated unseen terms grows linearly with the size of the document, which is measured as the number of different terms mentioned by the document. Formally, given the set of seen terms in a document, $Seen_d$, and the size of the largest (smallest) document in the collection, max_dl (min_dl) we compute the number of negated literals for a given document as:

$$N_d = |V| - |Seen_d| - \lfloor \frac{(max\_dl - |Seen_d|) \cdot MAX\_OT}{max\_dl - min\_dl} \rfloor$$

where V is the vocabulary composed of all terms in the collection and MAX_OT is a parameter that controls the maximum number of omissions allowed. This will be the key parameter to control partiality. If MAX_OT is equal to 0 then we allow no omissions and, hence, all unseen terms have to be negated. This approach can be referred to as *closed-world assumption* because it assumes that all what we have not seen explicitly is actually false. As MAX_OT grows the representations become more partial and documents receive different treatment depending on their length. Large documents get less omitted terms and, as a consequence, more negated literals will be introduced in their logical representations.

3.3 Example

Let us consider the alphabet $T = \{a, b, c, d, e, f, g, h, i, j\}$ and two documents d_1 and d_2, such that the sets of seen terms are $Seen_{d_1} = \{a, b\}$ and $Seen_{d_2} = \{c, d, e, f, g, h, i\}$, respectively. Imagine also that the longest document's size is 9, the shortest document's size is 2 and the MAX_OT parameter is set to 5. The number of negations for each document is:

$$N_{d_1} = 10 - 2 - \lfloor \frac{(9-2) \cdot 5}{9-2} \rfloor = 3$$

$$N_{d_2} = 10 - 7 - \lfloor \frac{(9-7) \cdot 5}{9-2} \rfloor = 2$$

Note that the final representation of the short document, d_1, will be much more partial (5 omissions) than the one for d_2, which will have a single omitted term. Observe also that a closed-world assumption would have assigned 8 negations to d_1 and 3 negations for d_2, leading to complete document representations (no omissions). On the contrary, the new approach tends to compensate this difference. In order to get the final logical representations we need the EMIM similarity values. The next table presents the EMIM scores for the vocabulary terms (for simplicity, we only show the values which are relevant for the present example):

$EMIM$ (t_i, t_j)	c	d	e	f	g	h	i	j
a	.001	.002	.003	.003	.003	.004	.005	.005
b	.002	.002	.002	.003	.004	.004	.004	.005

Since we are allowing 3 negated terms in d_1, it will have 5 omitted terms. Then, we have to take the set of unseen terms for d_1, $\{c, d, e, f, g, h, i, j\}$, and rank them in decreasing order of similarity to the document. For the sake of clarity, we will use in this example the similarity function f_1 (eq. 3).

$f_1(c, d_1) = \sum_{t_i \in d_1} EMIM(c, t_i) = 0.003$

$f_1(d, d_1) = 0.004$, $f_1(e, d_1) = 0.005$

$f_1(f, d_1) = 0.006, f_1(g, d_1) = 0.007,$

$f_1(h, d_1) = 0.008, f_1(i, d_1) = 0.009,$

$f_1(j, d_1) = 0.010$

This means that f, g, h, i and j will be omitted and c, d and e will be negated in d_1. The final logical representation for d_1 will be: $d_1 = a \wedge b \wedge$

$\neg c \wedge \neg d \wedge \neg e$. A similar process can be applied for d_2, leading to a logical formula having 7 positive literals and 2 negative literals. Note that we allow 5 omitted terms in the logical representation of d_1 whereas d_2 is assigned a logical representation having a single omitted term. This compensates the fact that d_2 is larger.

3.4 Global term frequency

In [8], a simple method to produce logical formulae from plain text was tested. The techniques applied did not use any kind of term similarity but they still allowed the construction of logical formulae with varying degree of partiality. This work is therefore a good reference to compare with. Given a document, the unseen terms are ranked in decreasing order of appearances in the whole collection (global term frequency) and the top ranked terms are omitted and the remaining terms are included as negations into the logical formula. This use of global term frequency information is somehow similar to smoothing for language modeling. Indeed, LM smoothing strategies tend to quantify the relatedness between unseen terms and document's contents using the probability of the word in a reference model, which is usually estimated from a large collection reflecting the general use of the language. This means that a null probability is not assigned for a term which was not seen in the text of a document. The fact that we have not seen it does not make it impossible. It is often assumed that a non-occurring term is possible, but no more likely than what would be expected by chance in the reference collection. If a given term is infrequent in the document base then it is very unlikely that documents that do not mention it are actually related to this topic (and, thus, very unlikely that any user that wants to retrieve those documents finds the term useful when expressing her/his information need). On the other hand, frequent terms are more generic and have more chance to present connections with the topics of documents even in the case when they are not explicitly mentioned. This suggests that unseen infrequent terms are good candidates to formulate negations in the logical indexing process.

Once we have a rank of the unseen terms in decreasing order of appearances in the collection, we only need to apply a document-length dependent approach such as the one explained in 3.2 in order to set a threshold. This approach will also be included in the experiments for the purpose of comparison.

4 Evaluation

We designed some experiments in order to test the effectiveness of the novel approach based on term similarity. The experiments involved three different approaches to represent texts as logical formulae. The baseline experiments were executed under a *closed-world assumption*, in which all unseen terms are simply negated in the logical representation of the document. The simple approach based on global term frequency, sketched in section 3.4, was also included in the experiments. The third method tested was the approach based on term similarity. We ran experiments for both EMIM-based functions designed in section 3.1.

The document base utilized was the WSJ collection in TREC disks 1&2. This dataset is composed of 173,252 news articles. We took 100 TREC topics and divided them into two equal-sized sets of topics (training and test). We used the training set to tune the parameters for each approach and, next, we did not allow any sort of adjustment for the experiments with the test set. Stop words were removed and the remaining terms were stemmed using Porter's algorithm. Logical queries are constructed by simply connecting their stems through logical conjunctions. Queries were long because all topic subparts were processed (Title, Description and Narrative). As usual in TREC-like experimentation, the top 1000 documents were used for evaluation.

We first ran the training experiments for the baseline run (labeled as CWA). The PLBR model is controlled by an internal parameter α. Roughly speaking, α determines how much a query-document matching term contributes to the measure of distance between a document and a query[4]. Table 1 depicts the performance values for the cwa training experiments (only the most

[4] In a pure propositional approach a query-document matching term would add 0 to the distance but the PLBR model was extended to include an addend for matching terms. The value of α belongs to the interval $[0, 1]$ and the

Topics #151-#200			
α	0.4	0.5	0.6
cwa	0.0719 1533	0.1055 1760	**0.1520** **1751**

Table 1: CWA (baseline) - Training

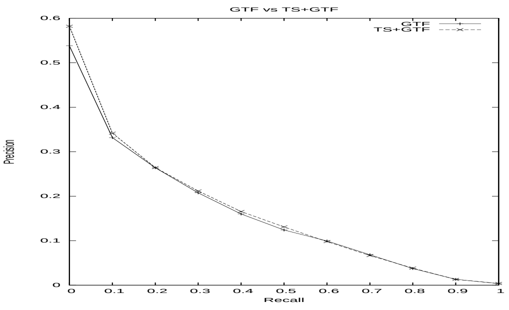

Figure 1: GTF vs TS+GTF

significative runs are shown). Each cell shows the non-interpolated average precision and the total number of relevant retrieved documents. The best results were found for a value of α equal to 0.6 (in bold).

In the method based on global term frequency (labeled here as GTF), we have an additional parameter to tune, MAX_OT, which controls the maximum number of omissions that we allow in the logical representation of a document. Experimental results are shown in table 2. The best training run was obtained with $MAX\_OT = 2000$ and $\alpha = 0.5$.

Concerning the approach based on term similarity, the first training runs were rather disappointing. When we omitted a significant number of terms ($MAX\_OT \geq 1000$) the performance was equal or worse than the baseline performance. Nevertheless, we inspected the ranks of unseen similar terms and it seemed that the first hundreds of terms were clearly connected to the document topics but the quality of the terms decreased significantly for positions down in the rank. This happened for both term similarity functions, f_1 and f_2. We hypothesize that this could be due to the fact that at the top of the list the similarity is based mostly on term co-occurrence while down in the list it is based more on term co-absence. This seems to be a feature of the EMIM measure. In order to overcome this limitation we designed an hybrid approach as follows. Since the ideal number of omissions for the PLBR model with the GTF method seems to be around 2000[5], we set MAX_OT to 2000. The top ranked unseen terms in the term similarity list were considered as omissions and, in order to have 2000 omissions, the remaining terms were taken from the GTF list. In this way, we tried to take advantage from the ability of the term similarity-based approach to detect highly related terms and, on the other hand, avoid noise taking terms from the GTF list which seemed more consistent at low ranks. We did experiments taking 50, 100 and 300 terms from the term similarity list and the remaining omissions were obtained from the GTF list. The training results of this hybrid approach are shown in table 3. The first conclusion we can draw is that there is not significant difference between the two term similarity functions. Second, it seems that retrieval performance is highly sensible to the quality of the terms taken from the ts list. The values of the performance ratios are very inconsistent when the number of terms taken from the ts list varies. This is especially noticeable for low values of α (e.g. 0.4). On the other hand, when $\alpha = 0.6$, the performance results are less sensitive to the quality of the ts list. As explained above, the PLBR model gives more relative importance to the matching terms as α grows. This means that with low values of α the role of non-matching terms becomes more important and, hence, the distinction between omissions and negations is fundamental. On the other hand, a high value of α leads to a retrieval performance which is roughly the same as the performance obtained with the baseline (cwa). This is not surprising because we are indeed giving relative low importance to the non-matching terms.

Once we adjusted the parameters for the three methods, we ran additional experiments with the set of test topics. Results are shown in table 4. The hybrid approach, which combines term similarity and global term frequency techniques, is the best performing method. The improvement with

contribution to the final distance from matching terms grows as α grows. This value is usually significantly less than 1, which is the value added to the distance by a contradiction (e.g. tv in the query and $\neg tv$ in the document). Indeed, previous experimentation on the PLBR model showed that it is optimum for values of α around 0.5.

[5] This optimum value was also very stable in previous experimentations with the method based on global term frequency [8].

Topics #151-#200							
	MAX_OT						
α	1000	2000	3000	4000	5000	10000	50000
0.4	0.1320	0.1544	0.1475	0.1420	0.1422	0.1136	0.0736
	2013	2090	1849	1912	1845	1639	1539
0.5	0.1470	**0.1687**	0.1562	0.1537	0.1613	0.1526	0.1075
	2010	**2048**	1786	1837	1950	1810	1764
0.6	0.1561	0.1513	0.1289	0.1041	0.1290	0.1426	0.1452
	1864	1738	1447	1298	1578	1522	1748

Table 2: GTF - Training

Topics #151-#200 (MAX_OT=2000)						
	f_1			f_2		
α	50(ts)+1950(gtf)	100(ts)+1900(gtf)	300(ts)+1700(gtf)	50(ts)+1950(gtf)	100(ts)+1900(gtf)	300(ts)+1700(gtf)
0.4	0.072	0.1608	0.073	0.072	0.1610	0.073
	1533	2072	1537	1533	2078	1537
0.5	0.1055	**0.1758**	0.1065	0.1055	**0.1764**	0.1065
	1759	**2027**	1761	1759	**2042**	1761
0.6	0.1520	0.1543	0.1520	0.1520	0.1543	0.1520
	1571	1684	1752	1751	1688	1752

Table 3: TS+GTF - Training

respect to the baseline run is very significant. Although the overall difference in non-interpolated average precision between GTF and the hybrid method is small, the improvement is consistent across most recall levels (as shown graphically in fig. 1). Moreover, the improvement in precision is especially large at low recall levels (e.g. 0.5821 vs 0.5379 at recall level 0). We also performed the sign test between the GTF run and the TS+GTF run revealing that the difference is statistically significant (5%).

The experimental outcome seems to indicate that the use of term similarity for producing partial document representations was beneficial in terms of retrieval performance and it is especially good to improve the quality of the top ranked documents. Nevertheless, we still have to study alternative methods to compute the similarity between an unseen term and a document and analyze deeply the reasons behind the low quality of the term similarity list at low positions.

5 Conclusions and future work

We have proposed a method based on term similarity that drives the construction of logical document representations. The ability of logic to deal with partial representation was exploited in combination with popular IR approaches in order to get more expressive document representations.

The first experiments revealed that a pure method based on term similarity was good to identify a few hundreds of highly related terms but, if we want to omit more terms, the quality of the subsequent logical representations is harmed. Hence, we designed a hybrid approach in which a significant number of unseen terms were omitted in the logical representation of a document. The set of omitted terms contains terms which are similar to the document and terms which are frequent in the whole document base. This new method showed good retrieval performance, especially at low recall levels.

Future work will be focused on studying alternative ways to compute the similarity between a term and a document because we still feel that there are some inconsistencies in the behaviour of the term similarity approach that we want to solve.

Acknowledgements

The first author warmly thanks *Fundación Caixa Galicia*, which supported financially a summer stay in Glasgow to develop this research. David E. Losada belongs to the “Ramón y Cajal” program, whose funds come from “Ministerio de Educación y Ciencia” and the FEDER program. This work was also partially supported by projects TIN2005-08521-C02-01 (from “Ministerio de Educación y Ciencia”) and PGIDT03PXIC10501PN (from “Xunta de Galicia”).

Topics #101-#150			
	CWA	**GTF**	**TS+GTF** (100(ts)+1900(gtf))
Recall	$\alpha = 0.6$	$MAX\_OT = 2000, \alpha = 0.5$	$MAX\_OT = 2000, \alpha = 0.5$
0.00	0.4576	0.5379	0.5821
0.10	0.2842	0.3317	0.3417
0.20	0.2154	0.2639	0.2642
0.30	0.1788	0.2075	0.2117
0.40	0.1445	0.1600	0.1655
0.50	0.1195	0.1240	0.1309
0.60	0.0923	0.0993	0.0979
0.70	0.0717	0.0684	0.0665
0.80	0.0397	0.0373	0.0384
0.90	0.0188	0.0131	0.0129
1.00	0.0098	0.0035	0.00 38
Avg.prec. (non-interpolated)	0.1319	0.1482	0.1514
% change		+12.4%	+14.78%
Total relevant retrieved	1828	2301	2356
% change		+25.9%	+28.9%

Table 4: CWA(baseline), GTF, TS+GTF - Test

References

[1] A. Berger and J. Lafferty. Information retrieval as statistical translation. In *Proc. SIGIR-99, the 22nd ACM Conference on Research and Development in Information Retrieval*, pages 222–229, Berkeley, USA, August 1999.

[2] F. Crestani. Exploiting the similarity of non-matching terms at retrieval time. *Information Retrieval*, 2(1):25–45, 2000.

[3] F. Crestani and C. J. van Rijsbergen. Probability kinematics in information retrieval. In *Proceedings of the 18th ACM Conference on Research and Development in Information Retrieval (SIGIR'95)*, pages 291–299, Seattle, USA, July 1995.

[4] S. Deerwester, S. Dumais, G.W. Furnas, T. Landauer, and Harshman. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41(6):391–407, 1990.

[5] D. Harman. *Information retrieval: data structures and algorithms*, chapter Relevance feedback and other query modification techniques. Prentice Hall, New Jersey, USA, 1992.

[6] D. E. Losada and A. Barreiro. Using a belief revision operator for document ranking in extended boolean models. In *Proc. SIGIR-99, the 22nd ACM Conference on Research and Development in Information Retrieval*, pages 66–73, Berkeley, USA, August 1999.

[7] D. E. Losada and A. Barreiro. Propositional logic representations for documents and queries: a large-scale evaluation. In F. Sebastiani, editor, *Proc. 25th European Conference on Information Retrieval Research, ECIR'2003*, pages 219–234, Pisa, Italy, April 2003. Springer Verlag, LNCS 2663.

[8] D. E. Losada and A. Barreiro. Negations and document length in logical retrieval. In A. Apostolico and M. Melucci, editors, *Proceedings of SPIRE-2004, the 11th Symposium on String Processing and Information Retrieval*, Padova, Italy, 2004. Springer.

[9] S. Robertson and S. Walker. Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval. In *Proc. SIGIR-94, the 17th ACM Conference on Research and Development in Information Retrieval*, pages 232–241, Dublin, Ireland, July 1994.

[10] C. J. van Rijsbergen. A theoretical basis for the use of co-occurrence data in information retrieval. *Journal of Documentation*, 33(2):106–119, 1977.

[11] H. Zaragoza, D. Hiemstra, and M. Tipping. Bayesian extension to the language model for ad hoc information retrieval. In *Proc. 26th ACM Conference on Research and Development in Information Retrieval, SIGIR'03*, pages 4–9, Toronto, Canada, 2003.