

Using Ontology Dimensions and Negative Expansion to solve Precise Queries in CLEF Medical Task

Jean-Pierre Chevallet
IPAL-CNRS
Institute for Infocomm Research
21 Heng Mui Keng Terrace
Singapore 119613
viscjp@i2r.a-star.edu.sg

Joo-Hwee Lim
Institute for Infocomm Research
21 Heng Mui Keng Terrace
Singapore 119613
jooHwee@i2r.a-star.edu.sg

Saïd Radhouani
Centre universitaire d'informatique
24, rue Général-Dufour
CH-1211 Genève 4, Switzerland
CLIPS-IMAG France
Said.Radhouani@cui.unige.ch

Abstract

We present here the method we have used for indexing multilingual text part of the Image Medical CLEF Collection. The result of the textual querying is then mixed with the image matching. We show by our results that a fusion of two media are of a great benefice because the combination of text and image returns clear better results than the two separately. We focus in this paper on the textual indexing part using a medical ontology to filter the document collection. At first, we use the notion of ontology dimensions, which corresponds the split of the ontology into sub ontology. In our experiment we just use the first tree level of the MESH ontology. We have modelled and experimented two different approaches of the use of the ontology: the first one is an ontology filtering that can force some terms of one dimension to be present in the final document. We have noticed a strong improvement using this technique over the classic Vector Space Model. The second technique manages the preference of some terms among other in the same dimension. Our hypothesis is that precise document should emphasis only few terms of a given dimension. To compute this new constraint, we have set up a negative weight query expansion. Finally, the combination of the two methods produces the overall best results. To our opinion, it shows that for a given domain, adding explicit knowledge stored into an ontology tree, enable to classify the importance of terms used in the query and enhance the finale average precision.

Categories and Subject Descriptors

H.3 [Information Storage and Retrieval]: H.3.1 Content Analysis and Indexing; H.3.3 Information Search and Retrieval; H.3.4 Systems and Software; H.3.7 Digital Libraries; H.2.3 [Database Management]: Languages—*Query Languages*

General Terms

Measurement, Performance, Experimentation

Keywords

Ontology Dimensions, Negative Weight, Text and Visual Fusion

1 Introduction

The text part of the image CLEF 2005 medical collection is in three languages: English, French and German. This forms an unusual IR test collection because of the great variation of documents length and the use of very precise terms. On the query side, it seems clear that some regularity appears. Each query roughly deals with anatomy, modality of the image and pathology. We call these "dimensions" of the query. Hence we have made the obvious assumption that a relevant document to a query with dimensions, is the one that fulfils correctly to these dimensions. To precise this notion, we decide to refer dimension to ontology. For example, here are the first levels of the MESH ontology.

- Anatomy [A]
- Organisms [B]
- Diseases [C]
- Chemicals and Drugs [D]
- Analytical, Diagnostic and Therapeutic Techniques and Equipment [E]
- Psychiatry and Psychology [F]
- Biological Sciences [G]
- Physical Sciences [H]
- Anthropology, Education, Sociology and Social Phenomena [I]
- Technology and Food and Beverages [J]
- Humanities [K]
- Information Science [L]
- Persons [M]
- Health Care [N]
- Geographic Locations [Z]

We call dimension the sub tree of the hierarchy of an ontology. For the CLEF medical queries, we have only used the dimensions `Anatomy [A]`, `Diseases [C]` and `Analytical, Diagnostic and Therapeutic Techniques and Equipment [E]`. The question we face for this experiment is the way to take into account these dimensions into an Information Retrieval System. In the next section we present the way we have include these dimensions into the classical Vector Space Model. We have used a filtering Boolean method on ontology dimensions, combined with a negative weight terms vector expansion based also on ontology dimensions. We can classify this sort of task as "precise information retrieval", which could be in-between classical thematic IR and question answering. In the following sections we detail models and methods used for the use of these dimensions.

2 Using Ontology dimensions

When we want to take into account the notion of dimension relative to an ontology, it means that the query usually focuses to one instance or possible value in one dimension and exclude all others. For example, if we are searching one image about one special body region of the dimension `Anatomy` from the ontology, the choice of a body region explicitly exclude other body region. If the query is about "Abdomen" then all documents which are about an other body region are irrelevant. It is then clear that the use of dimension of an ontology leads to express the notion of

term exclusion at the query level. Unfortunately, there is no way for the Vector Space Model to express such term exclusions. In the next part, we discuss such extensions to this model.

We present two different ways to take into account dimensions. The first one maps the initial query to ontology dimensions. This method produces one query vector per ontology dimension. The second is simpler and still uses only one vector but add negative weight to exclude terms belonging to the same dimension. We present at first the mapping of the query to ontology.

2.1 Filtering corpus by ontology dimensions

Our aim is to take into account dimensions present in query to build a precise Information Retrieval System. The basic idea is to split the initial query Q into several sub queries, each addressing one ontology dimension. Our goal is to give some terms priority depending the ontology dimension they belong to. For that purpose, we use Boolean expressions on the sub query. It is a Boolean pre filtering technique on top of Vector Space Model system.

At first, we extract dimension term sets from query. It means that we dispatch query terms to sub dimension queries. We use the ontology to determine the correct dimension of a term. We do not solve yet any ambiguity because the query belongs to a precise domain. We call this split *a mapping* between the query Q and each ontology dimension O_i . For a given ontology O , we define as O_i to be the set of terms under this dimension. The mapping is just a term set intersection: $Q_i = Q \cap O_i$, where Q_i is a sub query for the dimension i .

Once, sub query dimensions Q_i are extracted we build a Boolean query by the disjunction of all terms and we query the whole document collection. The result is a sub set \mathcal{D}_i of the collection where each document $D \in \mathcal{D}_i$ contains at least one term of the query dimension Q_i . These sets of document are precise because they contain explicitly dimensions terms that are in the query.

In order to solve the original multidimensional query, we have finally to combine these dimensions. It is done by a Boolean expression on the dimensions. In that way, a conjunction forces dimensions to be present together. We can reduce this constraint using a disjunction. We compute this Boolean dimension constraint formula using all sub sets $\{\mathcal{D}_i\}$.

We obtain a final sub set of document \mathcal{D} that has been filtered by the ontology in two ways: first having at least one term from the query from a given dimension, and second by selecting some dimension to appears together in the selected document. For example, for an initial query Q containing three dimensions regarding the ontology O , sub query Q_1 , Q_2 and Q_3 are build, and \mathcal{D}_1 , \mathcal{D}_2 , \mathcal{D}_3 are obtained after a disjunction Boolean retrieval. If we decide that a relevant document must include dimension 1 and dimension 2 or only dimension 3, we compute the sub document set by the boolean formula $\mathcal{D} = (\mathcal{D}_1 \cap \mathcal{D}_2) \cup \mathcal{D}_3$.

After this filtering, the next step is to query this sub set \mathcal{D} using the full original query Q using the classical Vector Space Model, that gives us the final document ranking. A similar approach based on term pre-filtering before Vector Space Model querying has been apply in multilingual CLEF [3]. Another method to combine VSM querying and dimension filtering can be to perform the filtering after the vector querying (post filtering). Doing the other way just reduces the number of documents to deal with for the final ranking.

In the following sections, we detail the other use of ontology dimensions by negative vector extension.

2.2 Negative weight for VSM

Vector Space Model (VSM) for Information Retrieval basically uses the inner product on term weight to compute the matching between document and query. In the case of a binary weighting, the inner product is the *size of the intersection term set* between document and query. If fact, even with non binary weighting, and for example with a normalisation of weight that leads the inner product to be the cosine value of the angle between the two vectors, the paradigm in use is still a sort of weighted term intersection.

Each vector dimension is associated to an indexing term. A non null value associated to a term (i.e. to its dimension) means that this term is relevant for the document indexed by this

vector. In practice, this notion of relevance is closely related to the presence of this term into the document after some transformation like stemming. It follows that for large collection and for small document, most of the dimensions of a given vector are null. It means also that only query terms that participate to the intersection are effective for the computation of the Relevance Status Value (RSV or matching score) that ranks the document. A longer query may have no effect on the RSV.

As the underlying meaning of the matching process is term intersection, only positive weights are used in this model. Our idea is to exploit the fact that in a general vector space, the value of a weight can be negative. Using negative weight just enhances the expressivity of this model and enables to reuse both indexing technique and software. Negative weights can appear [4] during the relevance feedback process. As mention by Harman: “Generally a negative weight reflect the fact that a term has much higher distribution in non-relevant documents than in relevant ones [...]”. Other approach to IR based from logic like in [7] suggests that negative weights could be an interesting way of explicitly expresses the notion of “non relevance”. However, in this approach negative weighting is still related to relevance feedback.

We want to use ontology dimensions, and then to express term exclusion. Logic modelling is one way to achieve this goal by more powerful query expression.

2.3 Logical negation

Beside the basic Boolean model which enables query to be any logical expression, some author promote the fact that the matching in Information Retrieval is closely related to a logical deduction [9]. For example, in [2] the closed world assumption is adopted.

In the logical IR model, a query matches a document if there is a deduction link from the document d to the query q . This can be modelled by the entailment $d \models q$. In this case, the matching is obtained when all models¹ of the logic formula associated with d are included in those associated with q .

The more obvious choice for document associated formula is a model with only one interpretation. In IR, logical interpretations are easy to understand: a term t is associated to the *true* value for a document d (i.e. in the interpretation of the logic formula of the document d) if it is a relevant index term to d . Choosing only one logical interpretation to model a document, leads to unique interpretation for every term in the logic language and leads to a sort of close world assumption. In fact either a term is relevant (or a good index) for a document, either it is not. For $d \models q$ example, given the set of possible index terms $\{a, b, c, d, e\}$, a document d indexed with a, b will have the following associated logic formula: $a \wedge b \wedge \neg c \wedge \neg d \wedge \neg e$. Then the set of terms $\{a, b, c, d, e\}$ is then associate to only one interpretation: $\{t, t, f, f, f\}$.

On the query side, every logical formula can be used. The meaning is different from documents. We use direct terms (i.e. not negated) to describe the fact we want a term to be an indexing term of the document. We use a negated term to explicitly discard a term from being into the indexing set of the retrieved document. The conjunction is used to force two situations to be valid for the relevant document, and disjunctions are alternatives. If a term does not appear in the query it means that user has no particular needs concerning this term. It may or may not appear in a relevant document.

The query is associated to a set of logical interpretation. Adding a negated term reduces² the set of interpretation and then reduces the set of relevant document.

This is exactly what we would like to use for expressing our query with ontology dimensions. For each ontology dimension, we would like to select on term and exclude all the others. Unfortunately the Vector Space Model does not enable logical negation in a query.

¹We recall that a logical interpretation is a Boolean function that associates each term to a logical value. A model for a logic formula is an interpretation that turns the formula to be true.

²In fact divide by two.

2.4 Negation as negative weighed terms

The common way to introduce negation to VSM is to perform a post matching filtering. To our point of view, this solution is too strict because of the Boolean interpretation of the formula. If we keep the matching schema of the VSM with inner product, the evolution choices are limited. We could choose for example the Belief revision operator in [6] that seems to be a good extension VSM. But for these experiments, we finally decided to test a more simplistic extension. We propose to explore another possibility.

We would like to use the same paradigm as the logical modelling: *adding a term to a query tends to reduce the matching possibilities*. As we have seen earlier, this paradigm is false for the VSM. For example, there is no differences between a query $a \wedge b \wedge c$ and the query $a \wedge b$ for a document indexed only by $\{a, b\}$.

For that we refer to the logical modelling for IR and we add a second stage that transforms some interpretation set to a three valued vector the following ways: the true value t is associated to the value 1, the false value f is associated to -1 . If there exists several interpretations in the set that are identical except for one value then we associated this interpretation to the value 0. In this way, we are not able to interpret every logical formulae but only conjunction of direct and inverse variables.

For example, given the term set $\{a, b, c, d, e\}$, a document indexed by the formula $a \wedge b \wedge \neg c \wedge \neg d \wedge \neg e$ is associated to the vector $\{1, 1, -1, -1, -1\}$. The query $a \wedge b \wedge \neg c$ is associated to the interpretation set: $\{\{t, t, f, f, f\}, \{t, t, f, f, t\}, \{t, t, f, t, f\}, \{t, t, f, t, t\}\}$ and hence associated to the vector $\{1, 1, -1, 0, 0\}$.

Starting for this new modelling, we propose to still use the inner product $V_d \times V_q$ to compute the RSV. This matching is no more based on one term set intersection, but on the combination of four term set intersection (see formula above). In fact, the matching is the sum of the intersection of term within the same sign, minus the intersection of terms that are in opposition. In the following formula, we note V^+ the positive vector obtained from the positive of null value, and V^- the positive vector obtained from the negatives or null values, so that $V = V^+ - V^-$.

$$\begin{aligned} RSV(d, q) &= V_d \times V_q = (V_d^+ - V_d^-) \times (V_q^+ - V_q^-) \\ &= V_d^+ V_q^+ + V_d^- V_q^- - (V_d^+ V_q^- + V_d^- V_q^+) \end{aligned}$$

With this model we have the following new properties compared to normal VSM:

- Adding a term to a query always change the RSV;
- For a given document the maximum RSV is obtained when the query has only one interpretation which is equal to the one of the document;
- A null RSV is possible event if document and query share common indexing terms.

It is then trivial to extend this model using fuzzy matching values for the weight of the vectors. In the following we present how we simplify and apply it to the CLEF experiment.

2.5 Negative expansion on VSM using ontology

We make the two following assumptions: first we suppose terms are organized into an ontology hierarchy. Second, we suppose that a term in a query includes all other sub-term of the hierarchy but also excludes terms of other branches of the ontology. It is also a sort of close world assumption.

With this new model, one can then express queries with negated element to constraint and focus on some terms. This has been used for building a query that explicitly excludes indexing terms that are on the same ontology dimension. For example, if terms $\{a, b, c\}$ and $\{d, e, f\}$ belongs to the same ontology dimension, one build the query: $(a \wedge \neg b \wedge \neg c) \wedge (d \wedge \neg e \wedge \neg f)$.

Our extension has one implementation problem: every document have a full sized non null dimension. It means that the traditional inverse file is no more half empty. One second drawback is that one must complete document vector at indexing time with terms from the ontology.

For this first experiment we have then simplify the formula to:

$$RSV(d, q) = V_d^+ V_q^+ - V_d^+ V_q^-$$

This simplify the problem and enables to test our approach for CLEF experiments but still have the drawback of having very large queries, because a lot of terms have then a non null value. For CLEF experiments, we have work on a reduced set of ontology dimensions terms to limit this problem.

3 Details of the indexing process

In this part we detail the indexing process and the use of the two models. For this experiment, we have decided to index all texts after a minimum of Natural Language Treatment. We have computed the part of speech (POS) of all documents of this collection. The simple treeTager[8] has been used for this task. For the indexing part, we have use the XIOTA experimental system [1].

3.1 Basic sequence of document treatments

The first treatment that has been applied to the collection is some XML correction and some re-tagging actions. The XML correction is just the transformation of some characters (like '<'), that should not appear in XML documents. For the MIR collection we have noticed a strong regularity in the document framework, and we have decided to reconstruct a documents framework by replacing some regular texts like "Brief history" into XML tags.

The second step on text treatment is the part of speech tagging. We think that for a precise information retrieval system, it is better to perform a filtering on POS tag than using a dictionary. We have used TreeTagger for this phase. Before the tagging we have selected the fields we think are worth value to be indexed. This step is very important because putting all the document fields could results on noise into the index, as avoiding some could lead to silence. In the following, we list the tag that has been retained:

CasImage: CASIMAGE_CASE ID Description Diagnosis ClinicalPresentation KeyWords Anatomy
Chapter Department Title

PathoPic (en): IMAGE ID Diagnosis Synonyms Description

PathoPic (ge): IMAGE ID Diagnose Synonyme Beschreibung

MIR: CASE ID IMAGES FINDINGS

Peir: IMAGE ID Description

Starting from the part of speech tagging, all documents from the same language are following a parallel processing path. It means that we have in fact three sets of indexing for each of the three languages: French, English and German.

The next step is the selection of indexing terms. We have used a filtering on POS tags instead a classical stop words method. We have keep only nouns, adjectives and abbreviations³. We finally obtain a term vector for each document in a given language. Documents in a given language from all collection are merged in the same indexing matrix. We have used the classical `ltc` indexing scheme of the vector space model for both query and document vectors.

Each query language is used to query the corresponding index matrix. The result is a set of documents IDF which are unique throughout the whole collection.

We finally make the fusion for all three languages by selecting only the best matching value when same document is retrieved from several languages in the same time. Taking the maximum value between languages just emphasis the language where the matching is more efficient to compute.

³In the TreeTager notation is it the list NOM,ADJ,ABR,JJ,NN,NNS,NP,ADJD,ADJA,NE

3.2 Negative dimension query expansion

In order to increase the precision of the system, we make a sort of closed word assumption relatively to a given ontology. We suppose that one term that appears in a query will exclude all other terms that could replace this term in the same level of the ontology.

We make the hypothesis that given a dimension (say Anatomy), only the term associate to the concept that appears in the query should be kept and all other should be avoid. We illustrate here some part of the MESH ontology.

```
Body Regions [A01]
  Abdomen [A01.047] +
  Back [A01.176] +
  Breast [A01.236] +
  Extremities [A01.378] +
  Head [A01.456]
    Ear [A01.456.313]
    Face [A01.456.505] +
    Scalp [A01.456.810]
    Skull Base [A01.456.830] +
  Neck [A01.598]
  Pelvis [A01.673] +
  Perineum [A01.719]
  Thorax [A01.911] +
  Viscera [A01.960]
```

For example, if in the query we have the term “Head”, then every terms not under this concept should be set as irrelevant and will have a negative weight in the query.

3.3 Results for CLEF 2005

For this test, we have limited ourselves to only two actual dimensions: anatomy and modality for all three languages. Also we have used a reduced term set for query negative expansion. The Boolean pre-filtering⁴ that has been used is only to force at least one ontology dimension to be present into the relevant document. Using this filter we obtain 20,75% of MAP (run IPALI2R_T) which is not an absolute good value but it is the second best value of all CLEF tests for this year.

We have then test the use of the negative query expansion. As this technique was new, we have decide limit the impact of this extension by using only two dimension, reducing the size of the ontology and by distributing the positive weight to the negative value so that the sum of all negative expansion of a term is equal in absolute value to the positive term weight. We show in the following a query example and the negative expansion of this query.

```
<vector id="11" size="5">
<c id="head" w="0.4149327576"/>
<c id="image" w="0.4474848211"/>
<c id="mri" w="0.3636860549"/>
<c id="sagittal" w="0.6271265149"/>
<c id="view" w="0.3194260299"/>
</vector>

<c id="abdominal" w="-0.0345777298"/>
<c id="chest" w="-0.0345777298"/>
<c id="liver" w="-0.0345777298"/>
<c id="head" w="0.4149327576"/>
```

⁴We have technically in fact a post filtering which is equivalent to pre-filtering and is here possible due to the small number of documents in this test collection

```

<c id="image" w="0.4474848211"/>
<c id="sagittal" w="0.6271265149"/>
<c id="skin" w="-0.0345777298"/>
<c id="abdomen" w="-0.0345777298"/>
<c id="infarction" w="-0.0345777298"/>
<c id="mri" w="0.3636860549"/>
<c id="kidney" w="-0.0345777298"/>
<c id="aorta" w="-0.0345777298"/>
<c id="femur" w="-0.0345777298"/>
<c id="brain" w="-0.0345777298"/>
<c id="view" w="0.3194260299"/>
<c id="heart" w="-0.0345777298"/>

```

With the combination of the two methods we have obtained the overall best text CLEF results of 20,84% of MAP (run IPALI2R_Tn). The increase of average precision using negative query expansion with an ontology is then encouraging. We explain this results by the fact document that are focus more clearly on one concept in one dimension are more precise and tend to be rank in better position than document that mixes different concept into the same ontology dimension.

3.4 Results analysis

In this part, we detail some intermediate results we have done on the collection to better understand the influence of our model on the quality of the results. For these tests, the query dimensions are: Anatomy, Modality, and Pathology. Using the Vector Space Model on documents tagged with the POS analyzer, without taking into account the query dimensions, we obtain 17.25% of MAP. Using the negative query expansion, we then obtain 17.32% of MAP. It is a small improvement about 0.4%. The distribution of the weight of the positive term among the negative one, strongly limit the change in the RSV, and explain also the limitation of improvement. In fact we have made the following implicit hypothesis :

A document containing many different terms from the same ontology dimension is less relevant than a document containing terms from different ontology dimensions.

The negative terms expansion is a consequence of this hypothesis. The improvement obtained show that this hypothesis is valid for this particular test collection. In the following runs, we now always use the negative query expansion technique. To take into account the query dimensions, we have tested several assumptions, that leads to several Boolean dimension combinations.

*Relevant documents must include at least one of the three query dimensions (if they exist)*⁵

For this case, documents that contain anatomy, modality or pathology dimensions in the query, are relevant. For this run, the map is about 19.64%. So the improvement of the result is about 13.4%.

Relevant documents must include all the three query dimensions (if they exist).

For this case, the result has decrease. We think that it is due to the fact that the CLEF documents do not usually contain terms describing the modality. For this reason, we prefer the following assumption:

Relevant documents must contain the anatomy and the pathology dimensions terms of the query.

In this case, our system provides a MAP of 21.39%. So the improvement, compare to the result of the previous run, is about 8.9%.

In the previous cases, we have supposed that all dimensions have the same importance in the query. This is assumption is not valid in all cases. Indeed, terms describing modality in the query

⁵We have manually discarded "image" term.

are not discriminative. For example, a CT can be an image of a liver or emphysema, etc. We think also that the terms describing the pathology are, sometimes, non discriminative. For example, a lesion can be a lesion of a skin or a lesion of a nerve. For these reasons, we suppose that the anatomy is the most discriminative dimension. We also do the following assumption:

Relevant documents must includes anatomy dimension terms of the query

For this run, the result is about 20.85%. Compare to the previous run, the result has decrease about 2.6%. But compare to the first run (only with negative query expansion), the result has increase about 20.4%.

3.5 Mixing textual and visual index

Textual index has been merged with visual indexing. Visual medical indexing leaning based is described in [5] Text content is supposed to be closer to semantics, as image contains information that cannot be fully transcript into text. Our goal in this experiment is to show that combination of image and text index can give better results than separate results.

To compute the new raking list from image and text, as we are working on the same corpus (image ID), we have the hypothesis that the absolute relevance status value should be the same in the two list. In practice of course they differs. We have then to rescale the RSV of the two lists using a linear transformation so that the RSV of the top document is always equal to 1.

We have then proposed two simple merging techniques: for each document in both ranked list, either we keep the best (max) ranking value, either we compute an average value. Keeping the best value, follows the hypothesis that either one media (text or image) is best to answer a query, as computing the average supposes that both are always equally participating to the ranking. We have obtained the following results:

Run	Fusion Method	negative query exp.	results MAP
IPALI2R_TI _{an}	Average	YES	28.21%
IPALI2R_TI _a	Average	NO	28.19%
IPALI2R_TI _{mm}	Max	YES	23.25%
IPALI2R_TI _m	Max	NO	23.12%

Results show clearly that both visual and textual participate to the ranking. It is finally very interesting to notice that this combination outperforms both text only and image only by a large amount: from text using image increase 35% of MAP. In the same time, from image using text we measure an increase of 291% ! We can also notice that this combination outperforms all other methods used to index this collection this year. We can finally conclude that for this collection, the quality of text indexing seems to have a greater influence than expected. It can be explain by the fact that query are related to a focused domain, where a term is not really ambiguous and is related to a strong and precise meaning.

4 conclusion

The results we have obtained for this participation are very encouraging and tend to show the interest of the use of explicit knowledge when solving precise queries. Benefits of mixing text and image are also very clear. The use of an ontology seems useful either a final filtering and as a negative query expansion.

This work has been done under the IPAL I2R laboratory, a joint lab founded by CNRS from the French side and A-STAR from the Singaporean side. This work has also been done in relation with the French CLIPS IMAG laboratory and the Centre Universitaire d'Informatique of Switzerland.

References

- [1] Jean-Pierre Chevallet. X-iota: An open xml framework for ir experimentation application on multiple weighting scheme tests in a bilingual corpus. *Lecture Notes in Computer Science (LNCS), AIRS'04 Conference Beijing*, 3211:263–280, 2004.
- [2] Yves Chiaramella and Jean Pierre Chevallet. About retrieval model and logic. *The Computer Journal*, 35(3):233–242, 1992.
- [3] Jacques Guyot, Saïd Radhouani, and Gilles Falquet. Ontology-based multilingual information retrieval. In *CLEF Workshop, Working Notes Multilingual Track, Vienna, Austria*, 21–23 September 2005.
- [4] Donna Harman. Relevance feedback revisited. In *SIGIR '92: Proceedings of the 15th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 1–10, New York, NY, USA, 1992. ACM Press.
- [5] Joo-Hwee Lim and Jean-Pierre Chevallet. A structured learning approach for medical image indexing and retrieval. In *CLEF Workshop, Working Notes Medical Image Track, Vienna, Austria*, 21–23 September 2005.
- [6] David E. Losada and Alvaro Barreiro. Using a belief revision operator for document ranking in extended boolean models. In *SIGIR '99: Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 66–73, New York, NY, USA, 1999. ACM Press.
- [7] David E. Losada and Alvaro Barreiro. Rating the impact of logical representations on retrieval performance. In *DEXA-2001 Workshop on Logical and Uncertainty Models for Information Systems, LUMIS-2001*, pages 247–253, September 2001.
- [8] Helmut Schmid. Probabilistic part-of-speech tagging using decision trees. In *Proceedings of International Conference on New Methods in Language Processing*, sept 1994.
- [9] C. J. van Rijsbergen. A new theoretical framework for information retrieval. In *ACM Conference on Research and development in Information Retrieval, Pisa*, pages 194–200, 1986.