

# Bayesian Networks for Structured Information Retrieval

Benjamin Piwowarski

LIP 6, Paris, France  
bpiwowar@poleia.lip6.fr

Huyen-Trang Vu

LIP 6, Paris, France  
vu@poleia.lip6.fr

Patrick Gallinari

LIP 6, Paris, France  
gallinar@poleia.lip6.fr

## ABSTRACT

We present a bayesian framework for XML document retrieval. This framework allows us to consider content only. We perform the retrieval task using inference in our network. Our model can adapt to a specific corpus through parameter learning and uses a grammar to speed up the retrieval process in big or distributed databases. We also experimented list filtering to avoid element overlap in the retrieved element list.

## Keywords

Bayesian networks, INEX, XML, Focused retrieval, Structured retrieval

## 1. INTRODUCTION

The goal of our model is to provide a generic system for performing different IR tasks on collections of structured documents. We take an IR approach to this problem. We want to retrieve specific relevant elements from the collection as an answer to a query. The elements may be any document or document part (full document, section(s), paragraph(s), ...) indexed from the structural description of the collection. We consider the task as a *focused retrieval*, first described in [1, 7].

The aim of the INEX (Initiative for the Evaluation of XML retrieval) initiative is to provide means, in the form of a large testbed (test collection) and appropriate scoring methods, for the evaluation of retrieval of XML documents. Among the different INEX tasks, we focused on free text queries (CO for Content Only) since many questions still remain open for this specific task. The Bayesian Network (BN) model is briefly described in section 2.1.

## 2. MODELS

The generic BN model used for the CO task was described in [8]. We only give here the main model characteristics. Our work is an attempt to develop a formal model for structured document access. Our model relies on bayesian networks and provides an alternative to other specific approaches for handling structured documents [6, 3, 4]. BN offer a general framework for taking into account relation dependencies between different structural elements. Those elements, which we call *doxels* (for Document Element) will be random variables in our BN.

We believe that this approach allows casting different access information tasks into a unique formalism, and that

these models allow performing sophisticated inferences, e.g. they allow to compute the relevance of different document parts in the presence of missing or uncertain information. Compared to other approaches based on BN, we propose a general framework which should adapt to different types of structured documents or collections. Another original aspect of our work is that model parameters are learnt from data. This allows to rapidly adapt the model to different document collections and IR tasks.

Compared to the model presented in [8], we have proceeded to different additions:

- We experimented with different weighting schemes for terms in the different doxels. Weight importance may be relative to the whole corpus of documents, to doxels labelled with the same tag, ...;
- We introduced a grammar for modelling different constraints on the possible relevance values of doxels in a same path ;
- For limiting the overlap of retrieved doxels, we introduced simple filtering techniques.

### 2.1 Bayesian networks

The BN structure we used directly reflects the document hierarchy, *i.e.* we consider that each structural part within that hierarchy as an associated random variable. The root of the BN is thus a "corpus" variable, its children the "journal collection" variables, etc. In this model, due to the conditional independence property of the BN variables, relevance is a local property in the following sense: if we know that the journal is (not) relevant, the relevance value of the journal collection will not bring any new information on the relevance of one article of this journal.

In our model, the random variable associated to a structural element can take three different values in the set  $V = \{N, G, E\}$  which is related to the specificity dimension of the INEX'03 assessment scale:

- N** (for Not relevant) when the element is not relevant;
- G** (for too biG) when the element is marginally or fairly specific;
- E** (for Exact) when the element has an high specificity.

For any element  $e$  and for a given query, the probability  $P(e = E|\text{query})$  gives us the *final* Retrieval Status Value (RSV) of this element. This value is used to order the different elements before returning the list of retrieved elements.

We considered two more types of random variables. The first one is the query need that is described as a vector of word frequencies. Note that this random variable is always observed (known). The second one is associated to *baseline* models and can take only two values : *relevant* and *not relevant*.

For a given query, a local relevance score is computed for each doxel via the baseline score models. This score only depends on the query and the doxel constraint without influence other doxels in the tree. Based on these local scores and on parameters, BN inference is then used to combine evidence and scores for the different doxels in the document model. For computing the local score, different models could be used. We used in our experiments simple retrieval methods and classical ones such as Okapi. The first one (*ratio*) compute for each element the value  $S_1$ :

$$S_1(\text{element}) = \frac{\sum_{\text{term}t} tf_{\text{query}}(t) \frac{tf_{\text{element}}(t)}{tf_{\text{parent}}(t)}}{\sum_{\text{term}t} tf_{\text{query}}(t)}$$

where  $tf_{\text{parent}}$  denotes the term frequency in the parent of the element,  $tf_{\text{element}}$  the term frequency within the element and  $tf_{\text{query}}$  within the query. The second one (*weight ratio*) is simply  $S_1$  divided by a decreasing function of the element length:

$$S_2(\text{element}) = \frac{S_1(\text{element})}{\log(20 + \text{length}(\text{element}))}$$

where the length of the element is number of words that this element and its descendants contain. All those formulas and coefficient were determined empirically. The main advantages of these formulas are that they are naturally bounded (between 0 and 1) and that they can be computed *locally*. We can then simply define the probability that an element is relevant ( $R$ ) for the first (resp. second) model  $M_1$  ( $M_2$ ) by:

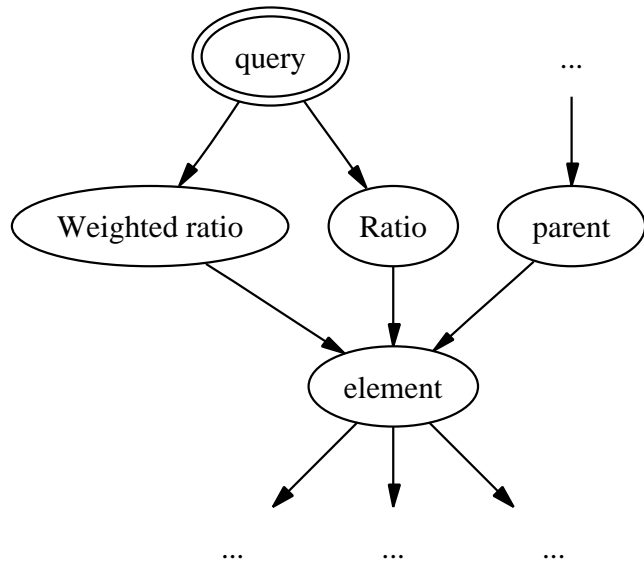
$$P(M_i = R|\text{query}, \text{elementcontent}) = S_i \text{ with } i \in \{1, 2\}$$

We also tried to add the classical Okapi model, but as its RSV are harder to normalize, we were not able to integrate it with success into our BN framework.

In our model, the probability that element is in the state  $N$ ,  $G$  or  $E$  depends on the parent state and on the fact that  $M_i$  has judged the element as relevant or not relevant (figure 1). We can then compute the probability using this formula for any element  $e$  and any state  $v \in V$ :

$$P(e = v|\text{query}) = \sum_{\substack{v_p \in V \\ r_1, r_2 \in \{R, -R\}}} \theta_{c(e), v, v_p, r_1, r_2} \\ \times P(e \text{ parent} = v_p) \\ \times P(M_1 = r_1|\text{query}) \\ \times P(M_2 = r_2|\text{query})$$

where  $\theta$  is a learnt parameter that depends on the different states of the four random variables (element state, parent state, baseline model 1 and 2 relevance) and on the category



**Figure 1: Bayesian Network model (detail view).** The element state depends on the parent state and on the relevance of the element for the model *ratio* ( $M_1$ ) and *weighted ratio* ( $M_2$ )

$c(e)$  of the element. The categories used in our experiment are shown in table 1. In our BN, scores are computed recursively with the above formula: we begin by the biggest doxels (INEX volumes) and then with smaller doxels (article, body paragraph, ...).

### Adding a grammar to the BN

We used a grammar in order to add some constraint on the retrieval inference process. That grammar enables us to express coherence rules on scored doxels within the same document path:

- A non relevant element *may not* have a relevant descendant:

$$\forall c, r_1, r_2, \theta_{c, v, N, r_1, r_2} = 0 \text{ if } v \in \{G, E\}$$

- A highly specific ( $G$ ) element has either a non relevant ( $N$ ) or a highly specific ( $E$ ) child

$$\forall c, r_1, r_2, \theta_{c, G, E, r_1, r_2} = 0$$

The main interest of this grammar is to provide us a way to make a decision about whether we can find an element which has a higher RSV in the set of descendants of a given element. Indeed, we can show that:

$$P(e = E|\text{query}) \leq P(p = E|\text{query}) + P(p = G|\text{query}) \quad (1)$$

where  $p$  is the parent of the doxel  $e$ .

### Learning parameters

In order to fit a specific corpus, parameters are learnt from observations using the Expectation Maximization (EM) algorithm. An observation  $O^{(i)}$  is a query with its associated

tags	category $c(e)$
ss, ss1, sec1	section
bib, bibl, ack, reviewers	misc
ip, ip1, ip2, ip3, bb, app, p1, p2	paragraph
figw, fig	figure
l1, l2, l3, l4, l5, l6, l7, l8, l9, la, lb, lc, ld, le, numeric-list, numeric-rbrace, bullet-list, index	list
index-entry, item-none, item-bold, item-both, item-bullet, item-diamond, item-letpara, item-mdash, item-numpara, item-roman, item-text	item
hdr, hdr2, hdr1, h3, h2, h2a, h1a, h1, h	header
bdy, article	container
* (any other tag)	other

**Table 1: Element categories**

relevance assessments (document/part is relevant or not relevant to the query). EM [2] optimizes the model parameters  $\Theta$  with respect to the likelihood  $\mathcal{L}$  of the observed data:

$$\mathcal{L}(O, \Theta) = \log P(O|\Theta)$$

where  $O = \{O^{(1)}, \dots, O^{(|O|)}\}$  are the  $N$  observations. Observations may or may not be *complete*, *i.e.* relevance assessments need not to be known for each structural element in the BN in order to learn the parameters. Each observation  $O_i$  can be decomposed into  $E_i$  and  $H_i$  where  $E_i$  corresponds to structural entities for which we know whether they are relevant or not, *i.e.* structural parts for which we have a relevance assessment.  $E_i$  is called the evidence.  $H_i$  corresponds to hidden observations, *i.e.* all other nodes of the BN.

In our experiment, we used for learning the 30 queries from INEX'02 and their associated relevance assessments.

## 2.2 Filtering

A Structured IR system has to cope with overlapping doxels, as it may for example return a section and a paragraph. In order to avoid duplicate information, it might be interesting to filter out the returned result in order to choose between different levels of granularity. We thus developed a simple filtering algorithm which we describe below. The basic idea is to remove an element when another element in the retrieved list contains or is contained by the element. For INEX'03, we chose a very simple filtering mainly motivated by intuition.

The filtering we chose removes some of the retrieved doxels in the list while preserving the relative ranking of other document components. Kazai et al. [5] had this idea with the so-called BEP<sup>1</sup>. We can consider our filtering step as an instance of BEP which doesn't take into account hyperlinks.

<sup>1</sup>Best Entry Point

Filtering is a necessary step for improving the effectiveness of Structured IR systems.

We tried the three following strategies:

**Root oriented** If a doxel appears on the retrieved list, its descendants in the document tree will not give any new information if they appear below in the list. We thus remove any element in the ranked list if an ancestor is higher in the list. This simple method favors large doxels which is in conflict with the objective (retrieve the most specific doxels as possible).

**Leaf oriented** This is the inverse of the previous approach. We remove an element from the list when there is a descendant higher. The limit of this method is that when the latest is not relevant, then all the other informations brought by the ancestor are lost for the user.

**BEP** BEP strategy cumulates root and leaf oriented filtering. That is, an element is kept only if there is neither descendant nor ancestor higher in the retrieved list.

We chose the "Root oriented" strategy for some of the official submissions for INEX'03.

## 3. EVALUATION

The definition of a measure is based on an hypothetical user behaviour. Hypothesis used in classical measures are subjective but do reflect a reality. In the Structured IR framework (SIR), we propose a measure that estimate the number of relevant doxels a user might see and that does depend on a specific user behaviour.

We made three specific hypothesis on the user behaviour. First, the user eventually consults the structural context (parent, children, siblings) of a returned doxel. This hypothesis is related to the inner structure of documents; Second, the specificity of a doxel influences the behavior of the user. Third, The user will not use any hyperlink. More precisely, he will not jump to another document. This hypothesis is valid in the INEX corpus but can easily be removed in order to cope with hyperlinked corpora.

The measure we propose is the expectation of the number of relevant doxels a user sees when he consult the list of the  $k$  first returned doxels divided by the expectation of the number of relevant doxels a user see if he explores all the doxels of the database. We denote this measure by ERR (for Expected Ratio of Relevant documents). This measure is normalized so it can be averaged over queries.

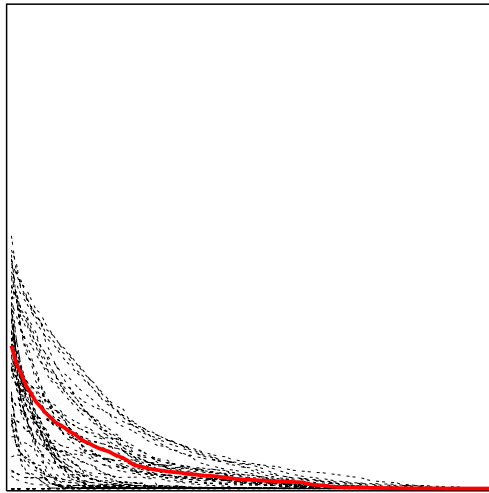
## 4. EXPERIMENTS

We performed different experimentations. We give below the performance of one of the BN models (figure 1). At the time of writing, our submitted runs range from the fifth to the thirty-sixth rank over fifty six submitted runs.

## 5. CONCLUSION

We have described a new model for performing IR on structured documents. It is based on BN whose conditional probability functions are learnt from the data via EM. This

quantization: generalized; topics: CO  
 average precision: 0.0479  
 rank: 18 (56 official submissions)



**Figure 2: Recall-precision for a BN model**

model uses a grammar for restricting the allowed state of a doxel in our BN knowing the state of its parent. The BN framework has thus three advantages:

1. it can be used in distributed IR, as we only need the score of the parent element in order to compute the score of any its descendants;
2. it can use simultaneously different baseline models: we can therefore use specific models for non textual media (image, sound, ...) as another source of evidence;
3. Whole part of the corpus can be ignored when retrieving doxels using inequality (1).

The model has still to be improved, tuned and developed, and several limitations have still to be overcome in order to obtain an operational structured information retrieval system. In particular, we should improve the baseline models<sup>2</sup> and further experiments are thus needed for tuning the learning algorithms and for filtering.

## 6. REFERENCES

- [1] Y. Chiamella, P. Mulhem, and F. Fourel. A Model for Multimedia Information Retrieval. Technical report, IMAG, Grenoble, France, July 1996.
- [2] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum Likelihood from incomplete data via de EM

<sup>2</sup>We tried to use one of the Okapi variants but it gives a score that can not be easily normalised

algorithm. *The Journal of Royal Statistical Society*, 39:1–37, 1977.

- [3] N. Fuhr and T. Rölleke. HySpirit - a Probabilistic Inference Engine for Hypermedia Retrieval in Large Databases. In H.-J. Schek, F. Saltor, I. Ramos, and G. Alonso, editors, *Proceedings of the 6th International Conference on Extending Database Technology (EDBT)*, Valencia, Spain, 1998. Springer, Berlin.
- [4] T. Grabs and H.-J. Schek. ETH Zrich at INEX: flexible information retrieval from XML with PowerDB-XML. Dec. 2002.
- [5] G. Kazai, M. Lalmas, and T. Rölleke. A Model for the Representation and Focussed Retrieval of Structured Documents based on Fuzzy Aggregation. In *String Processing and Information retrieval (SPIRE 2001) Conference*, Laguna de San Rafael, Chile, Sept. 2001.
- [6] M. Lalmas. Dempster-Shafer's Theory of Evidence Applied to Structured Documents: Modelling Uncertainty. In *Proceedings of the 20th Annual International ACM SIGIR*, pages 110–118, Philadelphia, PA, USA, July 1997. ACM.
- [7] M. Lalmas and E. Moutogianni. A Dempster-Shafer indexing for the focussed retrieval of a hierarchically structured document space: Implementation and experiments on a web museum collection. In *6th RIAO Conference, Content-Based Multimedia Information Access*, Paris, France, Apr. 2000.
- [8] B. Piwowarski, G.-E. Faure, and P. Gallinari. Bayesian networks and INEX. In *Proceedings of the First Annual Workshop of the Initiative for the Evaluation of XML retrieval (INEX)*, DELOS workshop, Dagstuhl, Germany, Dec. 2002. ERCIM.