Information Retrieval, 8, 655–681, 2005

A Bayesian Framework for XML Information Retrieval: Searching and Learning with the INEX Collection

BENJAMIN PIWOWARSKI bpiwowar@dcc.uchile.cl
Center for Web Research, DCC, Universidad de Chile, Blanco Encalada 2120, Santiago, Chile

PATRICK GALLINARI gallinar@poleia.lip6.fr
LIP6, 8, rue du capitaine Scott, 75015 Paris, France

Abstract. Most recent document standards like XML rely on structured representations. On the other hand, current information retrieval systems have been developed for flat document representations and cannot be easily extended to cope with more complex document types. The design of such systems is still an open problem. We present a new model for structured document retrieval which allows computing scores of document parts. This model is based on Bayesian networks whose conditional probabilities are learnt from a labelled collection of structured documents—which is composed of documents, queries and their associated assessments. Training these models is a complex machine learning task and is not standard. This is the focus of the paper: we propose here to train the structured Bayesian Network model using a cross-entropy training criterion. Results are presented on the INEX corpus of XML documents.

Keywords: Bayesian Networks, structured information retrieval, XML, machine learning for structured retrieval

1. Introduction

The expansion of the Web has been paralleled by the development of large electronic textual collections (e.g. electronic libraries, electronic journals and proceedings), and of semi-structured databases with textual or multimedia content. New textual representations, allowing interoperability, providing rich document descriptions and facilities to index and access different document parts and content types are needed in order to manage and access these resources. Several structured document representations and formats were then proposed during the last few years together with description languages like XML. They allow for richer descriptions with the incorporation of metadata, annotations and multimedia information inside a logical structure schema. Document structure for these new document formats is an important source of evidence, and should be considered together with textual content for information access tasks. Information retrieval engines should be able to cope with the complexity of these new document standards so as to fully exploit the potential of these representations and to provide new functionalities for information access. For example, users may need to access some specific document parts, navigate through complex documents or structured collections using the context provided by the logical structure. Queries may address simultaneously metadata, textual content and sometimes

logical document organisation. Extending information retrieval systems so that they can handle structured documents is not trivial and cannot be performed by a simple modification of existing flat IR systems. Many questions for designing such systems are still open. For example, there is no consensus on how to index these documents or on the design of efficient algorithms or models for performing information access tasks. Measuring structured IR systems performance is another area of open discussion in this field. The INEX initiative[1] has been launched in 2002 with the goal of developing research in this area. Different approaches and systems were presented and evaluated at INEX 2002 and 2003. We presented there a new model based on Bayesian networks (BN). It has been designed as a principled model for performing different IR tasks on collections of structured documents. BN parameters are learnt so that the model may adapt to different corpora and to different IR tasks. Basic versions of this model whose parameters were set up heuristically by hand performed rather well for the Content Only task of INEX 2003 and ranked in the top 10 according to different measures. However, in our tests for INEX 2003, we faced several difficulties for learning the parameters of these models from data. Training these stochastic models for Structured Information Retrieval (SIR) appears to be a complex problem with both conceptual and practical difficulties. The complexity comes from the definition of the task itself. Passages with very different characteristics (e.g. length, content or organisation) have to be ranked in the context of their enclosing document. Learning must take into account both passage characteristics and passage logical relationships. There is no general agreement for now on what the "optimal" output of a SIR system should be. This is one of the reasons for which evaluation measures for SIR systems are still an object of discussion and controversy. Accordingly, the goal of learning is not well defined. There are additional practical difficulties which are addressed later on.

In this paper, we consider ad-hoc retrieval for hierarchical document structures, i.e. we make the hypothesis that documents are represented as trees. This encompasses many different types of structured documents. For other cases as for example web sites, this is an approximation of the actual logical structure. The tree structure hypothesis allows keeping SIR model complexity down to a reasonable level. In this article, we focus on "content only" (CO) queries—that is queries that are expressed as a set of keywords without any structural constraint. The goal is then to rank first highly specific document elements—called in the following **doxels** (for DOCument ELement)—which are the most exhaustive and specific with respect to the query. We focus in the paper on the problem of training the BN model using one of the assessed INEX collection and we present a new training method. It is based on a discriminative training criterion instead of maximum likelihood used in a previous tentative.

The paper is organised as follows. Related work concerning Structured Information Retrieval (SIR) and BNs is reviewed in Section 2. Performance measures used in the experiments are briefly discussed in Section 3. The BN SIR model is then introduced in Section 4. This system makes use of local baseline IR models for scoring document elements which are described in Section 5. The training algorithm is described in Section 6 and a series of experiments on the INEX corpus are detailed and discussed in Section 7.

2. Related work

One of the pioneer works on document structure and IR is that of Wilkinson (1994) who attempted to use the document division in order to improve the performance of IR engines. For that he proposed several heuristics for weighting the relative importance of document parts and aggregating their contributions in the computation of the similarity score between a query and a document. Doing this way, he was able to improve a baseline IR system.

A more recent and more principled approach is the one followed by Lalmas and co-workers (Lalmas 1997). Their work is based on the theory of evidence which provides a formal framework for handling uncertain information and aggregating scores from different sources. In this approach, when retrieving documents for a given query, evidence about documents is computed by aggregating evidence of sub-document elements. Another important contribution is the HySpirit system developed by Fuhr and Rölleke (1998). Their model is based on a probabilistic version of Datalog. When complex objects like structured documents are to be retrieved, they use rules modelling how a document part is accessible from another part. The more accessible this part is, the more it will influence the relevance of the other part. A series of papers describing on-going research on different aspects of structured document storage and access, ranging from database problems to query languages and IR algorithms is available in a special issue of JASIST on XML retrieval,[2] and in recent SIGIR XML-IR workshops.[2,3,4]

In 2002, the Initiative for the Evaluation of XML Retrieval (INEX) was created in order to promote the evaluation of content-oriented XML retrieval. This led to two workshops (Gövert and Kazai 2002, Fuhr and Malik 2003) in which many different approaches were proposed. Most systems developed for the INEX CO task adapt to SIR "flat" text retrieval models like language models and vector space models. We have conceived a principled approach to SIR in which a BN is used to integrate information from various sources (baseline flat text models and document structure) in order to give a score to each document element.

BN (Pearl 1988, Jensen 1996) are a well known probabilistic framework which allows specifying graphically what the dependences between random variables are. Since Inquery (Callan et al. 1992), Bayesian networks have been shown to be a theoretically sound IR model, with state-of-the-art performance. Furthermore, they encompass different sound IR models. The simple network presented by Croft, Callan and Turtle (Callan et al. 1992) computes the probability that a query is satisfied by a document.[5] This model has been derived and used for flat documents. Ribeiro and Muntz (1996), Indrawan et al. (1994) and de Campos et al. (2003a) proposed other approaches also based on belief networks which were conceived for flat document retrieval. An extension of the Inquery model, designed to incorporate structural and textual information, has been proposed by Myaeng et al. (1998). In this approach, a document is represented by a tree. Each node of the tree represents a structural entity of this document (a chapter, a section, a paragraph and so on). This network is thus a tree representation of the internal structure of the document where the whole document is the root of the BN while the terms are its leaves. The relevance information goes from the document node down to the term nodes. When a new query is processed by this model, the probability that each query term represents the document is computed. In order

to obtain this probability, one has to compute the probability that a section represents the document, then the probability that a term represents this section and finally the probability that a query represents this term. In order to scale down the complexity of their model, the authors made several simplifying assumptions. Crestani et al. (2003a, 2003b) propose an extension of the BN model of de Campos et al. (2003a) for structured retrieval. Here the information goes in the reverse way compared to Myaeng et al. (1998), i.e. it goes from term nodes up to the document node.

Learning BN parameters is a well studied field (Krause 1998). One usually distinguishes between two learning problems: learning the structure of BNs and learning their parameters, i.e. their conditional probabilities. de Campos (2003a, 2003b) propose an algorithm for learning both structural relations encoding term relationships in their BN model and conditional probabilities for flat document retrieval. In our model, the BN structure is derived from the structure of the XML corpus. We thus focus on the parameters learning problem. Although different methods exist for that, they have mainly been experimented in a limited context and usually for a limited problem size (at least compared to the SIR problem handled here). Usually, for classification or diagnosis tasks, a unique BN is trained from a labelled data set. New examples are then presented to the BN model and BN inference allows computing the probability distribution for the different variables (e.g. class variables). Our approach is different from this unique BN strategy. Each document in the collection is modelled with a tree like BN. For a query, the probability of relevance for different document elements is then computed using this model. This probability is used in the final score computed for ranking the document elements. This is similar to modelling sequences for speech or biology with Hidden Markov Models or to the language model approach in IR, except that we do not consider here sequences but structured data. In this sense, this is a new learning problem. Also, the goal of SIR is to rank heterogeneous document elements. This is different from classical classification tasks where one has to make a hard decision on the class; the former is more difficult since the relative score values of doxels are important. The size of the collection and the number of document models which need to be handled are other sources of difficulty.

Our work is an attempt to develop a formal modelling of document elements retrieval for structured IR. Our modelling relies on Bayesian networks and provides an original approach to the problem. We believe that this approach allows casting different access information tasks into a unique formalism, and that these models will allow performing sophisticated inferences. For example, they allow computing the relevance of different document parts in the presence of missing or uncertain information. Compared to other approaches based on BNs, we propose a general framework which should easily adapt to different types of structured documents or collections. Another original aspect of our work is that model parameters are learnt from the data.

3. Measuring the performance for SIR systems

In this section, we first present the assessment scale used in INEX before introducing some of the measures which were proposed for SIR system evaluation. In many IR evaluation frameworks, the relevance value of a document is restricted to 0 (not relevant) or 1 (relevant). Such a scale is not suited for XML retrieval because of the "nested" nature of the

XML elements forming a document (Piwowarski and Lalmas 2004). Since INEX 2003, the following two-dimensional scale was proposed for the assessments:

- Exhaustivity (Ex), which describes the extent to which the document component discusses the topic of request.
- Specificity (Sp), which describes the extent to which the document component focuses on the topic of request.

Both exhaustivity and specificity are on a 4-points scale: not specific/exhaustive (0), marginally specific/exhaustive (1), fairly specific/exhaustive (2) or highly specific/exhaustive (3).

The evaluation of SIR systems for CO is an open problem and different measures were proposed and used for INEX, none of them being fully satisfying. The measures used in 2002, 2003 and 2004 for the evaluation are generalisations of Precision and Recall. We will make use here of the highly specific version of inex_eval measure which will be used for INEX 2004. There are several variants of the inex_eval measure. "Strict RP" credits only highly specific and highly relevant doxels, "generalised RP" credits the doxels according to their degree of relevance. The latter version favours systems which retrieve large elements (for example whole documents) while the former does not credit near matches. The generalised version is not well suited for SIR (Piwowarski and Gallinari 2003, Kazai et al. 2004). The "highly specific" RP (RP_{hs}) version[6] we chose for the experiments presented here, considers only highly specific doxels in the evaluation. It offers a good compromise since retrieving highly specific doxels better reflects the goal of SIR than retrieving any relevant element (generalised version) or only exact match elements (strict version). All measures ignore overlap between retrieved doxels. See Gövert et al. (2003) for a detailed description of these measures.

We will also make use of the Expected Ratio of Relevant (ERR) doxels measure (Piwowarski and Gallinari 2003). This measure is based on a new hypothetical user behaviour which extends the classical IR assumptions: a user examines the list of retrieved document sequentially in the reverse order of their relevance. With ERR, the structural context of the retrieved elements is taken into account. While the user examines the list of retrieved doxels as for classical IR, he or she is allowed to browse through the doxels in the structural context of the retrieved one. It is then assumed that the user will access the different context doxels with a given probability. This browsing behaviour is influenced by the specificity of the context doxels inferred by the BN. ERR can be seen as a generalisation of Recall for this user behaviour. INEX 2003 submissions were also evaluated using this measure. It was shown that for different specific behaviours, ERR performed better than inex_eval (Piwowarski and Gallinari 2003) with respect to the desired behaviour of a SIR system. One drawback of ERR is that it only measures the recall of the hypothetical SIR user and there is for now no such a generalisation for precision.

We believe that ERR like measures which take into consideration specific SIR user behaviour are better adapted to evaluate SIR systems than simple extensions of Precision-Recall. However, none of the measures proposed for now is fully satisfying. In the experiments (Section 7), performance will be evaluated with both "highly specific" inex_eval and

ERR. A presentation and a discussion of the different measures proposed at INEX 2003 can be found in Kazai (2003).

4. The Bayesian networks framework for structured information retrieval

4.1. Bayesian networks for structured IR

Belief networks (Pearl 1988) are stochastic models for computing the joint probability distribution over a set of random variables. They are Directed Acyclic Graphs (DAGs) whose nodes are random variables and edges correspond to probabilistic dependence relations between 2 variables. The structure of the DAG reflects conditional independence properties between variables, the joint probability of a set of variables $X_1, \ldots, X_n$ decomposes as follows:

$$P(X_1, \ldots, X_n) = \prod_{i=1..n} P(X_i \mid pa(X_i))$$

where $pa(X_i)$ is the parent set of X_i.

For SIR, BNs offer a natural framework for integrating different information sources corresponding to content and logical structure information. Let us consider as an example a simple document composed of 2 sections with respectively 1 and 2 paragraphs. Different BNs can be used for modelling this document. We will use as an illustration the simple BN shown in figure 1 where a structural element (corresponding to document D or to section S node) is connected in the DAG to its immediate descendents, and the paragraph nodes Pg are leaves. Variables D, S and Pg associated to the document, section and paragraph doxels will take their value in a set of relevance values, for example (relevant R, non relevant $\neg R$). Suppose for now that, given a query q, $P(X = R \mid q)$ is used for scoring the doxel associated to variable X in the BN. Using the conditional independence hypothesis in this document BN model, the score of Pg_2 for example can be computed as:

$$P(Pg_2 = R \mid q) = \sum_{d,s_2} P(Pg_2 = R \mid s_2, q) P(s_2 \mid d, q) P(d \mid q)$$

where the summation is over all possible values d and s_2 (R and $\neg R$ in this example) of BN variables D and S_2. The score of an element thus depends on its context as defined by the dependence relations encoded in the BN. With this simple model, this context is reduced

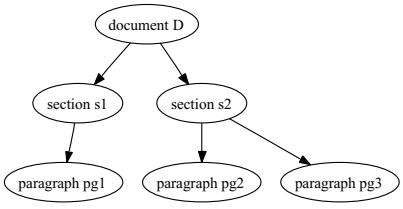


Figure 1. A simple BN for modelling a document composed of 2 sections with respectively 1 and 2 paragraphs.

to the element ancestors. With more complex models, the context could be increased to include other BN variables like siblings, etc.

4.2. A Bayesian network model for structured information retrieval

Let us consider a hierarchically structured collection like the INEX corpus. Documents are organised in a category hierarchy in which the corpus is the root node, journal collections (a set of journals) are its immediate descendents, followed by journals, articles and so on. We view retrieval for such a collection as a stochastic process in which a user goes deeper and deeper in the corpus structure: the user starts its search at the "root node" of all categories, and then selects one or several categories in which relevant documents should be. For each category, he or she selects subcategories and/or documents within these categories. This process is iterated until the user has found the doxels which are the most exhaustive and specific—we will say that these doxels are SIR-relevant in order to avoid the confusion with the relevancy as defined in flat document databases.

The BN structure we use directly reflects this document hierarchy and retrieval follows the above stochastic process. We consider that each structural part within the hierarchy has an associated random variable. The root of the BN is thus a "corpus" variable, its children the "journal collection" variables, etc (see figure 2). The whole collection is thus a large BN which reflects the doxel hierarchy in the collection. Searching this model amounts to perform the retrieval process described above: higher level doxels are first evaluated and depending on their score, their descendents may be selected and scored 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 knew that the journal is (not) relevant, the relevance value of the journal collection would not bring any new information on the relevance of one article of this journal. This choice of model structure has mainly been motivated by practical considerations. We also considered using models taking into account relations between siblings, or models where the dependency between variables reflects a

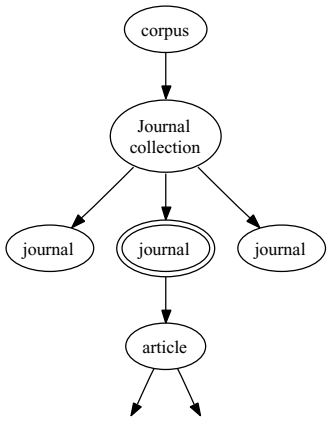


Figure 2. Independence in the BN. Knowing the relevance of a journal (the double circled Journal variable in the figure), the relevance of the journal collection has no influence on the articles relevance within this journal.

different semantic between the doxels. We gave up with these models since their complexity becomes rapidly prohibitive, even for dealing with medium size collections like INEX.

4.2.1. BN variables and notations. Each random variable in the BN can take its values in a finite set. Existing BN models for flat (Callan et al. 1992) or structured (Myaeng et al. 1998) documents use binary values $(R, \neg R)$. This is too limitative for SIR since quantifying a doxel relevance is more complex than for whole documents and should somewhat be related to the two dimensional scale (specificity, exhaustivity) proposed for INEX. The state space of the BN random variables should then be increased to better reflect this relevance space. Note that once again a compromise has to be found since increasing this space increases the model flexibility but also its complexity. A reasonable compromise was found experimentally with a state space of cardinality 3, $\mathbf{V} = \{I, B, E\}$ with:

I for **Irrelevant** when the element is not relevant;
B for **Big** when the element is highly exhaustive and marginally or fairly specific;
E for **Exact** when the element is highly exhaustive and specific.

The exact relation of these variable values with the two-dimensional scale of INEX will be described in Section 4.1; $v_x \in \mathbf{V} = \{I, B, E\}$ will denote a realisation of the variable X. The Retrieval Status Value (RSV) used with this model can be defined in several ways: the form of this RSV function depends on the semantic of the BN variables. A simple choice would be to use $P(X = E \mid q)$, i.e. a SIR-relevant doxel would be ranked according to the probability that it is both highly specific and exhaustive. Note that this does not take into consideration the context of the doxel. The retrieval process starts from top nodes and goes down in the corpus structure. It is then natural to take into account the state of parent nodes for scoring a doxel. For any element X and for a given query q, we propose to use $P(X = E \wedge X\text{'}s \text{ parent} = B \mid q)$ as the final Retrieval Status Value (RSV) of this element. This means that an element can be judged SIR-relevant only if its parent has already been judged relevant but not highly specific. This is consistent with the SIR-relevance definition: in the structured collection, the ancestors and therefore the parent of SIR-relevant elements are also exhaustive but are less specific. In the experiments, this choice led to better performance than the simple $P(X = E \mid q)$. Besides these variables, there are two more types of random variables in the BN. The first one corresponds to the query need: it is denoted Q and its realisation q. Q is a random vector of word frequencies taking its values in a multidimensional real space. This random variable is always observed (known). Document textual information is not directly modelled in this BN for complexity reasons. Instead a series of baseline IR models is used to compute local relevance scores for each doxel given a query. For each local baseline model, this score only depends on the doxel content and on the query. It is then independent from the context of the doxel in the XML tree. The global score for each doxel then combines these local scores and depends also on the doxel context in the BN. These local baseline models have been adapted from classical (flat) retrieval IR models. In the experiments presented here variants of the Okapi model were used for baselines. In the BN model a random variable is associated to each local scorer and doxel. Let $M_i(X)$ denote the random variable associated to the i^{th} local baseline model

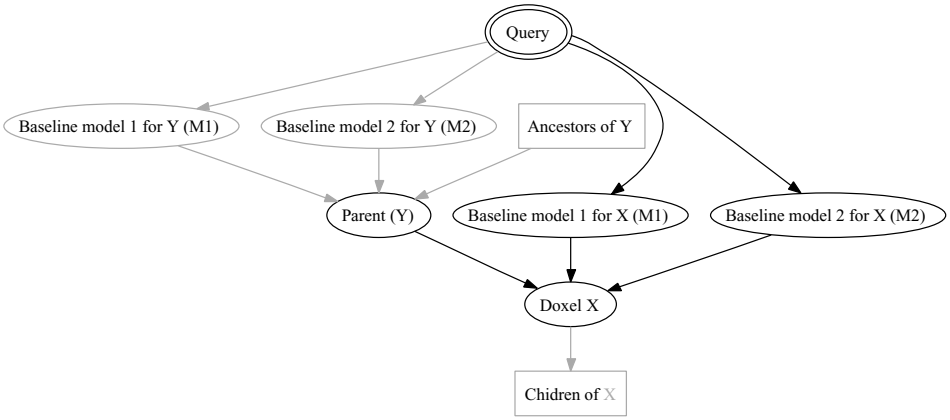


Figure 3. A local view of the BN—two local baseline models (model 1 and 2) are used here.

and to doxel X and m_i its realisation. As in classical IR this variable will take two values: R (relevant) and $\neg R$ (not relevant), i.e. $m_i \in \{R, \neg R\}$. The local relevance score of X given the query q for the ith model is $P(M_i(X) = R \mid q)$.

To summarise, the BN model is entirely defined by the variables X associated to each doxel which take their values in $\mathbf{V} = \{I, B, E\}$, conditional probabilities $P(X \mid pa(X))$ for all X, the Q variable corresponding to the query taking its values in $R^{|V|}$ (a vector of real components, one for each term of the vocabulary set V), the $M_i(X)$ local variables taking their values in $\{R, \neg R\}$ and the corresponding probabilities $P(M_i(X) = R \mid q)$. The three variable types are illustrated in figure 3. q, m_i and v_X will denote realisations of the corresponding BN variables and n the number of local baseline models used in the BN. The doxel score for ranking is $P(X = E \wedge X\text{'s parent} = B \mid q)$.

4.2.2. Doxel scoring in the BN. Based on the local scores $M_i(X)$ and on the BN conditional probabilities, BN inference is then used to combine evidence and scores for the different doxels in the document model. In our tree like model, the probability that element X is in state I, B or E depends on its parent state and on the fact that the local baseline models have judged the element as relevant or not relevant (figure 3). The probability that X is in a given state $v_x \in V$ is then:

$$P(X = v_x \mid q) = \sum_{\substack{v_Y \in \mathbf{V} \\ m_1, \ldots, m_n \in \{R, \neg R\}}} P(X = v_X \mid Y = v_Y, M_1(X) = m_1, \ldots, M_n(X) = m_n) \times P(Y = v_y \mid q) \times P(M_1(X) = m_1) \times \cdots \times P(M_1(X) = m_1) \quad (1)$$

This summation simply expresses the marginalisation of the joint probability of X and its parent variables. In this expression, the summation is over all the possible values of $v_y, m_1, \ldots, m_n$ (v_X can take any value in $\mathbf{V} = \{I, B, E\}$, and each m_i can take values in $R, \neg R$).

Table 1. Conditional probabilities associated to node X in figure 3. Each table entry gives the probability that X is I, B or E given the values of its parent variables Y and $M_i(X)$ for $i = 1, 2$. For example, the "*" in the table represents three values: $P(X = I \mid Y = B, M_1(X) = R, M_2(X) = \neg R)$, $P(X = B \mid Y = B, M_1(X) = R, M_2(X) = \neg R)$ and $P(X = E \mid Y = B, M_1(X) = R, M_2(X) = \neg R)$.

$P(X = I/B/E \mid Y, M_1(X), M_2(X)$		Y		
$M_1(X)$	$M_2(X)$	I	B	E
R	R			
R	$\neg R$		*	
$\neg R$	R			
$\neg R$	$\neg R$			

$P(X = v_x \mid Y = v_Y, M_1(X) = m_1, \ldots, M_n(X) = m_n)$ is usually encoded into conditional probability tables, one table for each doxel. Table 1 for example is the conditional probability table for doxel X in figure 3.

In many situations, for BN applications, these parameters are fixed by hand using prior knowledge, or when all variable values are known for each example, they are estimated by simply counting the co-occurrences of the different dependent variable values. Here, the situation is different. There are a large number of variables with no precise indication on what their value should be for each training example so that the parameters cannot be fixed *a priori* and cannot be estimated by simple counting. These parameters (i.e. conditional probabilities table entries $P(X = v_x \mid Y = v_Y, M_1(X) = m_1, \ldots, M_n(X) = m_n)$) will then be learnt as described in Section 6. Each table entry is a real number in [0, 1] and these entries verify the constraint

$$\sum_{v_X \in \{I,B,E\}} P(X = v_X \mid Y = v_Y, M_1(X) = m_1, \ldots, M_n(X) = m_n) = 1$$

as X takes its values in $\{I, B, E\}$. In order to limit the number of free parameters in the BN model, additional constraints were imposed on these conditional probabilities. Doxels were grouped in categories—each category corresponds to a set of tags with a similar semantic. All the doxels within a same category share the same conditional probability table. This means that conditional probability tables are the same for all the doxels in the same category—for the whole collection. Note that doxels in a given category may appear in the same document and/or in different documents. The categories used for the experimentations described in this paper are given in Table 2: 7 distinct categories were used. This parameter sharing strategy is common in stochastic modelling.

Table 2. Element categories used in the experiments on INEX 2003.

Tag name of doxel X	Category c_X
bib, bibl, ack, reviewers 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 index-entry, item-none, item-bold, item-both, item-bullet, item-diamond, item-letpara, item¬mdash, item-numpara, item-roman, item-text	list istem
hdr, hdr2, hdr1, h3, h2, h2a, h1a, h1, h	header
bdy, article	container
ss, ss1, sec1	section
*(any other tag)	misc

In order to enforce these constraints on the BN conditional probabilities, the probability estimates are defined as follows:

$$\begin{aligned} P(X = v_X \mid Y = v_Y, M_1(X) = m_1, \ldots, M_n(X) = m_n) \\ = F_X(\Theta, v_X, v_Y, m_1, \ldots, m_n) \\ = \frac{e^{\theta_{c_X, v_X, v_Y, m_1, \ldots, m_n}}}{\sum_{v \in \mathbf{V} = \{I, E, B\}} e^{\theta_{c_X, v, v_Y, m_1, \ldots, m_n}}} \end{aligned} \tag{2}$$

In (2), the $\theta_{c_X, v_X, v_Y, m_1, \ldots, m_n}$ are real values to be learnt. There is one such parameter for each tag category c_X and value set $v_X, v_Y, m_1, \ldots, m_n$. All the doxels sharing the same value set $c_X, v_X, v_Y, m_1, \ldots, m_n$ will share this parameter. θ_{c_X} for short will denote the vector of parameters associated to any node with tag c_X, i.e.

$$\theta_{c_X} = \left\{ \theta_{c_X, v_X, v_Y, m_1, \ldots, m_n} \mid v_x, v_y \in V,\ m_1, \ldots, m_n \in \{R, \neg R\} \right\}$$

Θ denotes the set of all the $\theta_{\ldots}$ parameters in the BN: $\Theta = \cup_{c_X} \theta_{c_X}$. The denominator in (2) ensures that conditional probabilities sum to 1. The particular form (2) has been found convenient here. Constraints could be taken into account in the formulation of the optimisation problem instead of using this particular form for the conditional probabilities. This would lead to much more complex learning algorithms. Note that according to the BN structure, query q does not appear explicitly in conditional probability (2). It appears implicitly in values $m_1, \ldots, m_n$ and v_Y.

5. Local baseline model: Okapi

Let us now present the local baseline models used for computing local doxel scores in the experiments. The BN has been designed so as to integrate scores computed by different baseline local scorers. Many baseline models could be used with this BN model. The only requirement is that their output can be interpreted as a probability of relevance given

the query. We used here Okapi variants as our local baseline models. Okapi (Walker and Robertson 1999) was adapted in order to (1) reach reasonable performance on the INEX corpus (and on a structured collection in general) and (2) compute a score which could be interpreted as a probability with this model. We did not consider feedback; this was left for further investigation.

The local RSV of a doxel X for a given query q, computed by the local baseline Okapi, is defined by:

$$\text{Okapi}(q, X) = \sum_{j=1}^{\text{length}(q)} \omega_{j,X} \frac{(k_1 + 1)tf_{X,j}}{K_X + tf_{X,j}} \times \frac{(k_3 + 1)qtf_j}{k_3 + qtf_j} \tag{3}$$

where k_1 and k_3 are constants,[7] length(q) is the number of terms in query q. (3) is similar to the classical Okapi except for the index X appearing in $\omega_{j,X}$, K_X and $\text{tf}_{X,j}$. Okapi makes use of different statistics relative to the document collection such as term occurrences or mean document length. Since for SIR elements to be retrieved are doxels and not plain documents, these statistics have to be adapted. Values $\omega_{j,X}$ and K_X are defined as follows:

- $\omega_{j,X} = \log(\frac{N - n_j + 0.5}{n_j + 0.5})$. In Okapi N is the number of documents in the collection and n_j the number of documents containing term j. There are different options for adapting these collection statistics to SIR. We will present here tests where these two values were defined respectively with respect to the classical document set ("document frequency") as in Okapi, to the set of all doxels ("element frequency") or to the set of doxels with the same tag ("tag frequency").
- $K_X = k_1((1-b) + b\frac{dl}{avdl})$ where b is a constant and in Okapi dl is the document length and $avdl$ is the average document length. Here dl was replaced by the doxel length and three different weighting schemes were tested for $avdl$: the average length taken respectively over all the doxels ("corpus"), over all the doxel with the same tag ("tag") or over all the sibling doxels ("parent").

Results for the whole INEX collection are shown in figure 4 for the ERR and the highly specific RP_{hs} measures. In this figure each experiment is denoted by two letters L1–L2. The first letter L1 corresponds to the "doxel collection" which is used for the computation of the document frequency. It can take three values: D ("document frequency"), E ("element frequency") and T ("tag frequency"). The second one L2 (length normalisation) can also take three different values: C ("corpus"), T ("tag") and P ("parent"). Both ERR and RP_{hs} measures order the different models similarly. Worst performance are obtained with (.-C) models length normalisation (3 bottom curves on figure 4 top). Length normalisation with respect to all doxels is ineffective, probably because of the large variability in doxel length in the collection. Normalisation with respect to doxels with the same tag or with respect to siblings is to be preferred. Models (E-.) and (T-.) have a middle range performance. Classical inverse document frequency (D-.) seems better suited than element or tag frequency. Other models perform equally well for ERR while being slightly different for RP_{hs}. Based on these results, for the BN experiments, we retained only two models among the best for both measures: the D-P (Document frequency—Parent length normalisation) and D-T (Document frequency—Tag length normalisation) versions of Okapi. The two models have a similar performance,

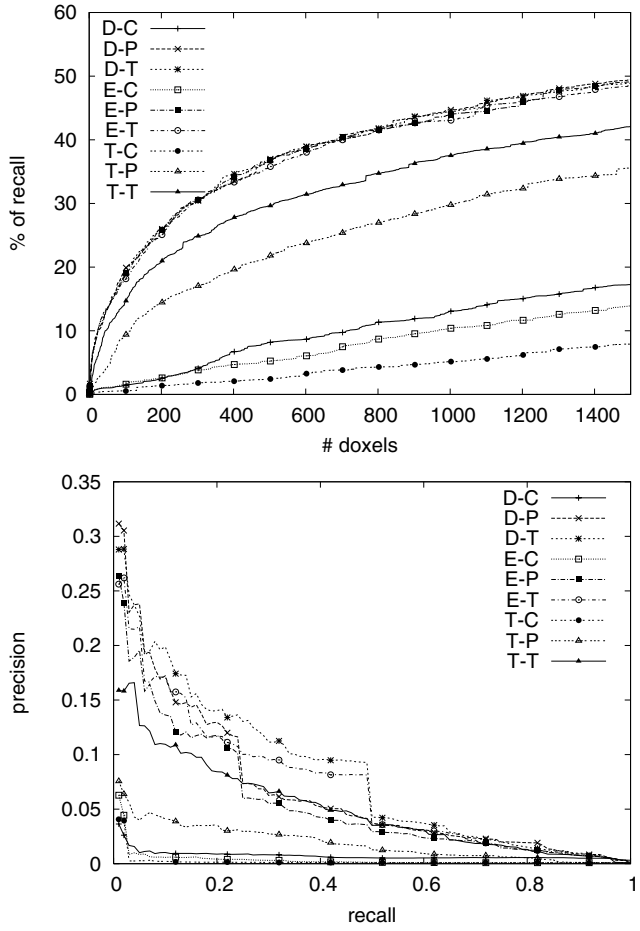


Figure 4. Okapi runs with the entire INEX collection; measures are "highly specific" inex_eval precision recall - only highly specific elements are taken into account in this measure (bottom) and ERR measure (top). The ERR measure is a generalisation of recall (Piwowarski and Gallinari 2003). The first letter corresponds to the "doxel collection" considered for the computation of the document frequency, it can take three values: D ("document frequency"), E ("element frequency") and T ("tag frequency"). The second one (length normalisation) can also take three different values: C ("corpus"), T ("tag") and P ("parent ").

but behave differently—the latter tends to rank higher smaller doxels. For INEX 2003 a similar model ranked 6 among 56 other models for the strict inex_eval measure.

For the BN model, one needs scores in the form of probability estimates. Okapi score does not range between 0 and 1. The normalisation of Okapi is discussed in Robertson (2002) in the context of filtering, where it is proposed to make a regression of the original Okapi score via a logistic function. We used this idea here with the following transformation:

$$P(M_{\text{Okapi}}(X) = \text{R} \mid q) = \frac{1}{1 + e^{\alpha \times \text{Okapi(q,X)/length}(q) - \beta}} \tag{4}$$

This formula gives the normalised score for the local baseline variants of Okapi model. The α and β parameters are estimated on the whole INEX 2002 database. Score (3) is dependant on the query length. Since the parameters of the logistic function should be valid for queries of varying length, this score was divided in (4) by the query length. We then computed the mean okapi score μ and the standard deviation σ for all the CO queries of INEX 2002.[8] We then set α and β such that the probability $P(M_{\text{Okapi}}(X) = R \mid q)$ is 0.5 when the score is μ and 0.75 when the score is $\mu + \sigma$. These values were chosen empirically. This is different from Robertson (2002) where the parameters of the regression are estimated for each query. This would not be realistic here because of the increased complexity of SIR.

6. Learning with BNs

Training BNs for SIR is a challenging machine learning task. Most applications of BNs deal with diagnosis or classification and usually, a single BN is used for a task and trained on a series of examples. In our case, there is one BN per document and it is used for scoring the document doxels for any query. There are many practical difficulties for this training task. First there is an important heterogeneity in the data set. For example there is a large variability in doxel length and content and the number of training examples is relatively small in the INEX database with respect to this variability (30 queries). Also, training requires a coherent labelling of the dataset. Query assessment for INEX is a tedious and non trivial task. Semi-automatic annotation tools are being developed in order to help assessors and to verify the coherence of the assessments. However, for now, although assessments for INEX 2003 were much more satisfying than for the preceding collection INEX 2002, they are not yet complete, coherent and homogeneous. They may then lead to contradictory judgements and mislabelling which is damageable for learning. Another difficulty comes from the nature of the task and the fact that the learning goal is not well defined. At last, ranking is a more difficult task than classification since the relative values of scores are important.

In any case, training the BN is a non standard application of machine learning and such a new situation usually requires extensive experimentations with different models and a lot of tuning for the learning parameters before finding an appropriate solution. We report here the solutions we have developed so far to this problem.

Training a BN is usually performed by maximizing the likelihood of the model over a training set. Different algorithms may be used for that. In the simplest situation when evidence is known for all variables, the conditional probabilities can be estimated by simple counting. When the values of some variables are unknown other algorithms can be used, one of the most popular being the EM (Estimation-Maximisation) algorithm (Dempster et al. 1977). Krause (1998) provides a review of training algorithms for BNs. In our case, for each query the set of variables with evidence consists of all variables associated to nodes with a relevance judgement. All other variable states are unknown or hidden in the terminology of BNs. Iterative methods like EM have then to be used for training.

Experiments performed with EM maximum likelihood training on INEX led to disappointing results. The algorithm allows learning the conditional probabilities: the data likelihood regularly increases with EM iterations. However for these different experiments

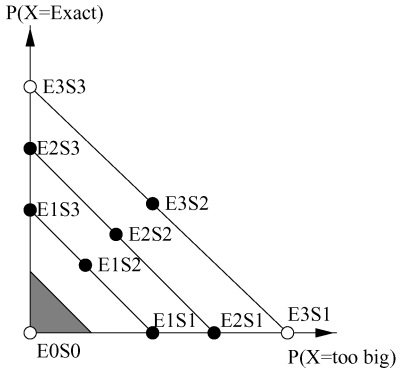
the performance measured with "highly specific" inex_eval or ERR was lower than the one obtained with Okapi D-P or D-T alone. Extensive experimentations did not allow us to solve this problem. Although we do not have a theoretical analysis of this phenomenon, we can provide some hints for explaining this failure. ML maximises the probability of observing all examples where an example is a triple (query, document, doxel assessments for this document). A first difficulty comes from the small proportion of assessed doxels in the collection, which means that only a small proportion of the BN variables will get evidence. A second difficulty comes from the observed variable labelling in the BN for ML. For ML training, with this BN model, each variable X shall take one and only one value among I, B or E. INEX assessments are on a two dimension scale (Exhaustivity, Specificity) with 4 possible values on each dimension. On these 4×4 possible values, only 10 are valid. Each of these 10 assessments should then be mapped onto the 3-dimensional space $\mathbf{V} = \{I, B, E\}$. Said otherwise, to each assessment for doxel X, we associate target values 1 or 0 for the probabilities $P(X = v_X \mid \text{parent variables of } X \text{ in the BN}, q)$ for $v_X \in \mathbf{V}$. First this means that we loose a large part of the information present in the assessments since the 10-dimensional assessment space is mapped onto the 3 dimensional $\mathbf{V}$ space. Second, the target probability distribution is only a very crude approximation of the goal of learning which should reflect the desired ranking of the doxels. Learning a more adequate probability distribution would involve a more complex BN model which would include real random variables. This would be prohibitive for SIR. We therefore propose to use another training criterion for the BN model: the cross-entropy (CE) between a target distribution and the distribution learnt by the BN. This criterion allowed us to reach more satisfying performance which is promising. As will be seen below, it reflects more closely the goal of learning for SIR, and allows much faster training than EM. As for ML, in order to learn node conditional probabilities with CE, a mapping must be defined between an assessment and its associated node variable value. We first describe below the mapping adopted here.

6.1. Mapping the INEX relevance scale to the probability distribution of the BN states

Let us denote $Ex_i Sp_j$ a relevance assessment where i and j may be 0, 1, 2 or 3. For each instance of $Ex_i Sp_j$ associated to a BN node X, we define a target probability distribution $P(X = v_X)$ for $v_X \in \mathbf{V} = \{I, B, E\}$. The corresponding mapping is defined in figure 5.

This mapping defines a semantic for the BN node values: E means highly specific and exhaustive, I irrelevant and B means highly exhaustive but marginally specific. Note that this mapping preserves more information with respect to the assessments than the one used for ML since ML needs *evidence* while CE needs a distribution over I, B and E. It is also closer to the learning goal since it provides ranked targets for the doxels instead of 0/1 labels for ML. To illustrate the difference between ML and CE mappings, consider figure 5. With ML, we must map each assessment to one of the three extreme points—the white discs in figure 5. With CE, we can choose any value in the space defined by the three white discs. This led to two kinds of problems for ML:

- Some assessment values were not mapped to any point: for instance, $\mathrm{Ex}_1\mathrm{S}_\mathrm{p}1$ is neither fully "too big" (it is not Ex_3), nor "relevant" (it is not $\mathrm{Ex}_3\mathrm{Sp}_3$), nor "irrelevant" (it is not



$P(X=...)$	E	B	I
$\mathrm{Ex}_3\mathrm{Sp}_3$	1	0	0
$\mathrm{Ex}_3\mathrm{Sp}_2$	0.5	0.5	0
$\mathrm{Ex}_3\mathrm{Sp}_1$	0	1	0
$\mathrm{Ex}_2\mathrm{Sp}_3$	0.75	0	0.25
$\mathrm{Ex}_2\mathrm{Sp}_2$	0.375	0.375	0.25
$\mathrm{Ex}_2\mathrm{Sp}_1$	0	0.75	0.25
$\mathrm{Ex}_1\mathrm{Sp}_3$	0.5	0	0.5
$\mathrm{Ex}_1\mathrm{Sp}_2$	0.25	0.25	0.5
$\mathrm{Ex}_1\mathrm{Sp}_1$	0	0.5	0.5
$\mathrm{Ex}_0\mathrm{Sp}_0$	0	0	1

Figure 5. Mapping the INEX scale to a probability distribution on states *I*, *B* and *E*. The table to the right gives the exact distribution associated to each $EX_i Sp_j$ assessment. The graph to the left gives the same information in a more intuitive way. Since $P(X = E) + P(X = B) + P(X = I) = 1$, we removed the $P(X = I)$ axis from the graph.

$\mathrm{Ex}_0\mathrm{Sp}_0$). Therefore, with ML, some information was lost. With CE, we can use this type of information.

- In order to avoid loss of information, it is possible to map different assessments onto the same point: for example in the ML mapping, $\mathrm{Ex}_3\mathrm{Sp}_3$ and $\mathrm{Ex}_2\mathrm{Sp}_3$ were mapped to the same point: the observation in this case was *E* (Exact). This mapping is obviously not desirable. A solution would be to add extra states to **V**. But this would make the BN more complex (the complexity for learning is proportional to $|\mathbf{V}|^2$—the number of different possible values a node and its parent can take—where $|\mathbf{V}|$ is the number of states). The output would be less easy to interpret and it would be difficult to define an adequate RSV function for instance.

6.2. *Training algorithm*

The training criterion is the cross-entropy between target variable values as they are defined by the above mapping and the values computed by the BN:

$$Q(\Theta) = -\sum_q \text{weight}(q) \sum_j \sum_{v_j \in V} P_T(X_j = v_j \mid q) \log P_\Theta(X_j = v_j \mid q) \tag{5}$$

where P_Θ is the probability to be estimated and P_T is the target distribution. We normalise the contribution of each query by

$$\text{weight}(q) = (\text{number of assessed nodes for } q)^{-1}.$$

The q summation is over the set of all training queries and the j one over the set of all variables X_j with a known probability distribution $P_T(X_j = v_j \mid q)$ for $v_j \in V$. The latter variables correspond to the doxel set with a known assessment in the training document set.

Compared to ML, this criterion gives a better approximation of the desired distribution at the different nodes and in this sense is thus closer to the goal of learning for SIR.

Minimizing $Q(\Theta)$ can be performed via gradient descent. The error derivative with respect to parameter θ writes:

$$\frac{\partial Q}{\partial \theta} = -\sum_q \text{weight}(q) \sum_j \sum_{v_j \in V} \frac{P_T(X_j = v_j \mid q)}{P_\Theta(X_j = v_j \mid q)} \frac{\partial P_\Theta(X_j = v \mid q)}{\partial \theta} \tag{6}$$

where the summation is the same as for (5). The update formula for parameter θ is (the derivation is provided in the appendix):

$$\begin{aligned}\theta^{(t+1)} \leftarrow \theta^{(t)} + \epsilon \sum_q \text{weight}(q) \sum_j \sum_{v_j \in \mathbf{V}} \frac{P_T(X_j = v_j \mid q)}{P_\Theta(X_j = V_j \mid q)} \\ \times \sum_{l \in anc(j)} \sum_{v_l, v_{pa(l)} \in \mathbf{V}} P_\Theta(X_j = v_j \mid X_l = v_l, q) P_\Theta(X_{pa(l)} = v_{pa(l)} \mid q) \\ \times \sum_{m_1, \ldots, m_n \in \{R, \neg R\}} \frac{\partial F_{X_l}(\Theta, v_l, v_{pa(l)}, m_1, \ldots, m_n)}{\partial \theta} \prod_{s=1}^{n} P(M_s(X_l) = m_s \mid q)\end{aligned} \tag{7}$$

where ϵ is the learning rate. The first summations over q, j and v_j are similar to those in (5). In the second summation ($l \in anc(j)$), for each value v_j of a variable X_j with a known assessment, we sum up all the contributions, with respect to a given parameter θ, of its ancestor pairs (X_l, parent $X_{pa(l)}$) where X_l is an ancestor of X_j. This contribution is modulated by the error term $P_T\left(X_j = v_j \mid q\right) / P_\Theta\left(X_j = v_j \mid q\right)$ and by the probability that X_j is in the state v_j if its ancestor X_l and $X_{pa(l)}$ are respectively in the states v_l and $v_{pa(l)}$.

The complexity of the update formula is:

$$O((\#\text{ queries}) \times |\mathbf{V}|^3 \times (\#\text{ BN nodes}) \times (\text{BN depth}) \times n)$$

where "# queries" is the number of queries in the training set, $|\mathbf{V}|$ is the number of states in the BN (3 here), "# BN nodes" is the number of nodes in the BN, "BN depth" is the average depth of assessed nodes in the BN and n is the number of local baseline models.
The implementation of the algorithm is straightforward and follows formula 7: we first loop on the queries, then on each BN node for which we have an assessment for the query, then on the different values of the latter variable, etc. All the parameters are updated in parallel. Practically, the learning process is fast and the main computational resources are needed for the evaluation of the BN performance.

Different gradient algorithms could be used here. For the experiments we used a simple gradient descent algorithm where the learning rate ϵ was automatically set up by a line search. We used the Armijo algorithm (Culioli 1994) which finds the largest epsilon $\epsilon^{(\text{opt})}$ for which:

$$Q(\Theta + \varepsilon \nabla Q(\Theta)) \leq Q(\Theta) + \epsilon^{(\text{opt})} \alpha \|\nabla Q(\Theta)\|^2 \tag{8}$$

We used $\alpha = 0.3$ for our experiments, we started with a value of 0.1 for $\varepsilon^{(0)}$, and divided this value by 2 until the inequality (8) was verified. Parameters Θ were then updated to $\Theta + \varepsilon^{(\text{opt})} \nabla Q(\Theta)$. Note that training for a node only requires the knowledge of its ancestor values which leads to a much faster training algorithm than for EM. The reason is that our CE criterion is only defined on variables for which there is an assessment. With ML and EM, the summation is over *all* variables in the BN whether their desired value is known (assessed) or not.

7. Experiments

We describe below experiments performed on the INEX 2003 corpus. We divided the 30 queries of INEX 2003 into two sets of 15 queries each (sets A and B). The queries were chosen at random for each set. Each set was used alternatively for training and test: training was performed on A and test on B and vice versa.

Figures 6 and 7 respectively give train and test learning curves for the BN trained with only one local baseline model (Okapi-D-T) and with two local baseline models (Okapi-D-P and D-T). We will refer respectively to the former and the latter as BN-1 and BN-2. For these figures, measures are averaged over sets A and B: training curves are an average of training performance on A and on B, test curves are an average performance of test on A (B being the training set) and on B (A being the training set). For each figure, x-axis corresponds to gradient iterations and y-axis gives a normalised performance measure. Different measures are plotted on each graph: "Error" corresponds to the CE value, RP_{hs} to the "highly specific" recall-precision average measure, ERR@ to the ERR criterion measured for the 10 first retrieved doxels (ERR@10), for the first 50 doxels (ERR@50) and averaged over 1500 retrieved doxels (ERR@avg). For both RP_{hs} and ERR measures, the plotted curve gives the ratio $\frac{\text{measure BN-1}}{\text{measure Okapi-D-T}}$ for BN-1 and $\frac{\text{measure BN-2}}{\text{max(measure Okapi-D-T, measure Okapi-D-P)}}$ for BN-2. For the CE "Error" curve, the graph corresponds to the ratio $\frac{\text{CE error at t}}{\text{CE error at t} = 0}$. For the different measures, Okapi-D (either D-T or D-P) is used as a reference for measuring the improvement brought by the BN. Any point above the $y = 1$ axis means an improvement over the baseline Okapi-D. The normalisation shows the relative improvement with respect to this baseline, irrespective of the measure scale. In each figure, the top graph gives the measure values on the training set while the bottom graph gives the measures on the test set. All these performances are relative to Okapi baseline. Therefore the BN "absolute" performance for the task could be inferred from the absolute Okapi performance curves from figure 4.

In all experiments, the error curve for CE clearly decreases for both train and test, meaning that the algorithm actually optimises in an effective way the CE criterion. Note that this mean error rapidly reaches a minimum—after about 1000 iterations in these experiments—while the other measures still evolve for a while.

After initial fluctuations, the RP_{hs} measure increases in all cases. For one local baseline scorer (figure 6) the mean improvement is about 1.1 both for training and test. For two local scorers (figure 7) this improvement reaches a value of about 1.2 and is more stable. With the ERR measures, the figures are somewhat different for the one and two models cases. For the former, ERR@50 shows a clear improvement over the baseline, ERR@avg

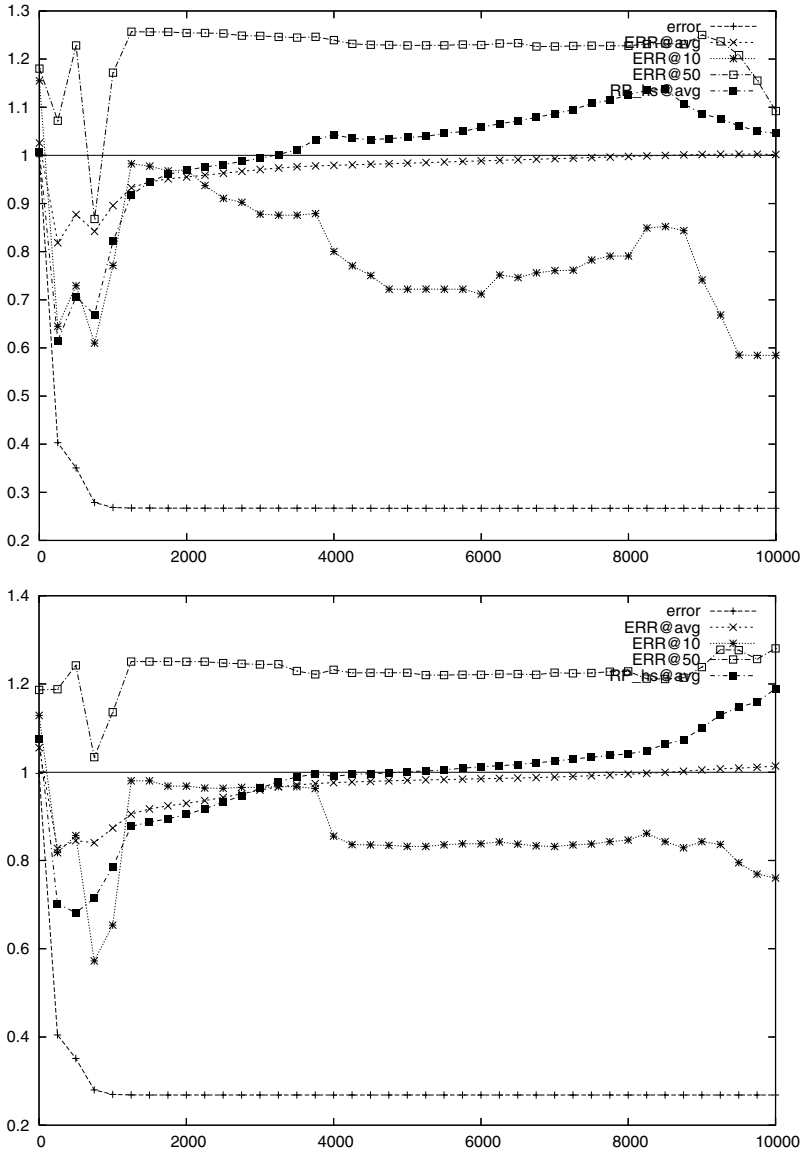


Figure 6. Train (top) and Test (bottom) learning curves with Okapi-D-T base model (BN-1). *X*-axis corresponds to learning iterations and *y*-axis to ratios BN-measure/baseline Okapi measure. The 5 plotted measures are detailed in the text.

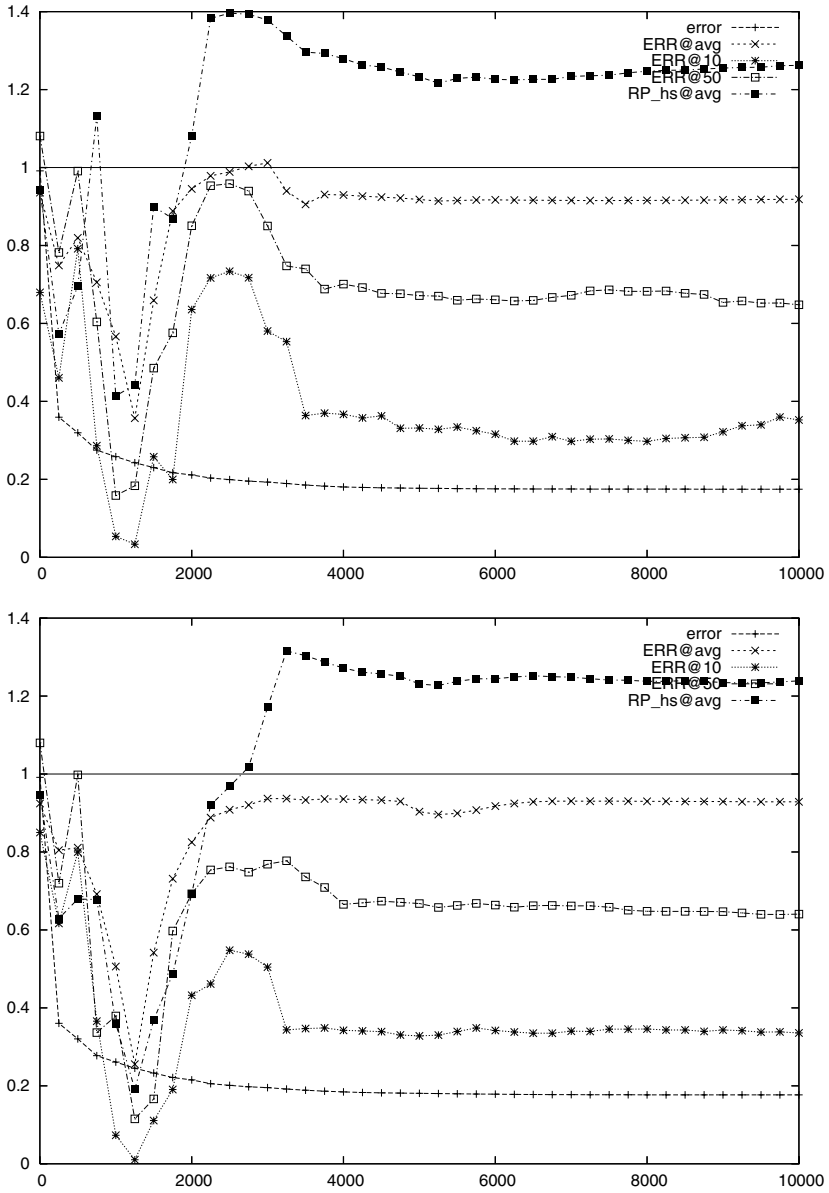


Figure 7. Train (top) and Test (bottom) learning curves with both Okapi-D-P and D-T base models (BN-2). X-axis corresponds to learning iterations and y-axis to ratios BN-measure/baseline Okapi measure. The 5 plotted measures are detailed in the text.

has performance similar to the baseline model while ERR@10 is much lower. For the latter, ERR@avg is slightly below the baseline, followed by ERR@50 and ERR@10 which is still lower. BN-2 behaves poorly with respect to the ERR@ measures while it does pretty well with RP_{hs}. It is affected by a small subset of the queries which get low BN scores.

Before proceeding, let us recall that *near matches* are taken into account with ERR, while they are not with RP_{hs}. Curves for BN-1 (figure 6) indicates that BN-1 moves higher in the retrieved list highly specific elements (RP_{hs} curve) and also near matches (ERR@50). For BN-2 the situation is different: as for BN-1, BN-2 also ranks higher highly specific elements but some of the near matches retrieved by Okapi are ranked lower by this BN.

Globally, there is a good agreement between learning and test curves for all measures. This is a positive indication on the behaviour of the algorithm: the algorithm learns and generalises correctly according to what has been learnt. The interpretation of SIR error measures is less trivial since they differ from the training criterion and since each measure reflects only partial aspects of the SIR goal. RP_{hs} is closest to the training criterion since highly specific (Sp_3) doxels are highly rewarded with the INEX mapping chosen here (figure 5). RP_{hs} increases in all cases when training proceeds. The correspondence between ERR measure and CE is less direct.

In order to provide more insight into the behaviour of the models, we have plotted different measures of the performance variability with respect to queries. On figure 8 (measure RP_{hs}) and figure 9 (measure ERR@50) the ratios of the performance of BNs over baseline model(s) after 9000 iterations are plotted for each of the 30 queries. Performance for both sets A and B appear on the same figure: the 15 training (test) scores for set A and the 15 training (test) scores for set B are displayed together. There is then one statistic for a given triple (query, BN, measure) and the numbers on the plots identify individual queries. It can be noted that for all plots, performance for most individual queries are similar whether they belong to the training or test set which means that there is a relative robustness in the learning process. Said otherwise, whatever the training set is, the BN(s) learn similar statistics. The variability between the individual query scores is relatively high. For RP_{hs} a few queries get lower scores with BN(s) and a larger number get a higher score. For ERR@50, this is somewhat similar but many query scores lie around the $y = 1$ axis. This information is plotted in a more synthetic way using boxplots.[9] On figure 8 (RP_{hs}), the median line is above 1 and reaches 2 for BN-2. Roughly speaking for all graphs of figure 8, about 65 % of the BN scores are above that of the Okapi baseline. The dispersion for values lower than the median for BN-2 is higher than for values higher than the median. For ERR@50, the median value is near 1 in all cases. Extreme points are relatively far from the median. The general figures are similar for both BN-1 and BN-2. On figure 9 (bottom), it can be seen that two extreme points (labelled 98 and 100) are zero. They correspond to queries with very low Okapi and BN scores, for which ERR@50 for BNs is 0.

We took a closer look at these two queries. Both share a common assessment "style" with many highly specific and highly relevant nested elements. For example, a section, one of its subsections and a paragraph of the latter were labelled highly specific and exhaustive. This is in contrast for instance with query 126 which has been assessed by the authors and for which the BN performs always better than the baseline. The conceptualisation of SIR is clearly not the same for the two assessors. Note that instructions for INEX are in accordance

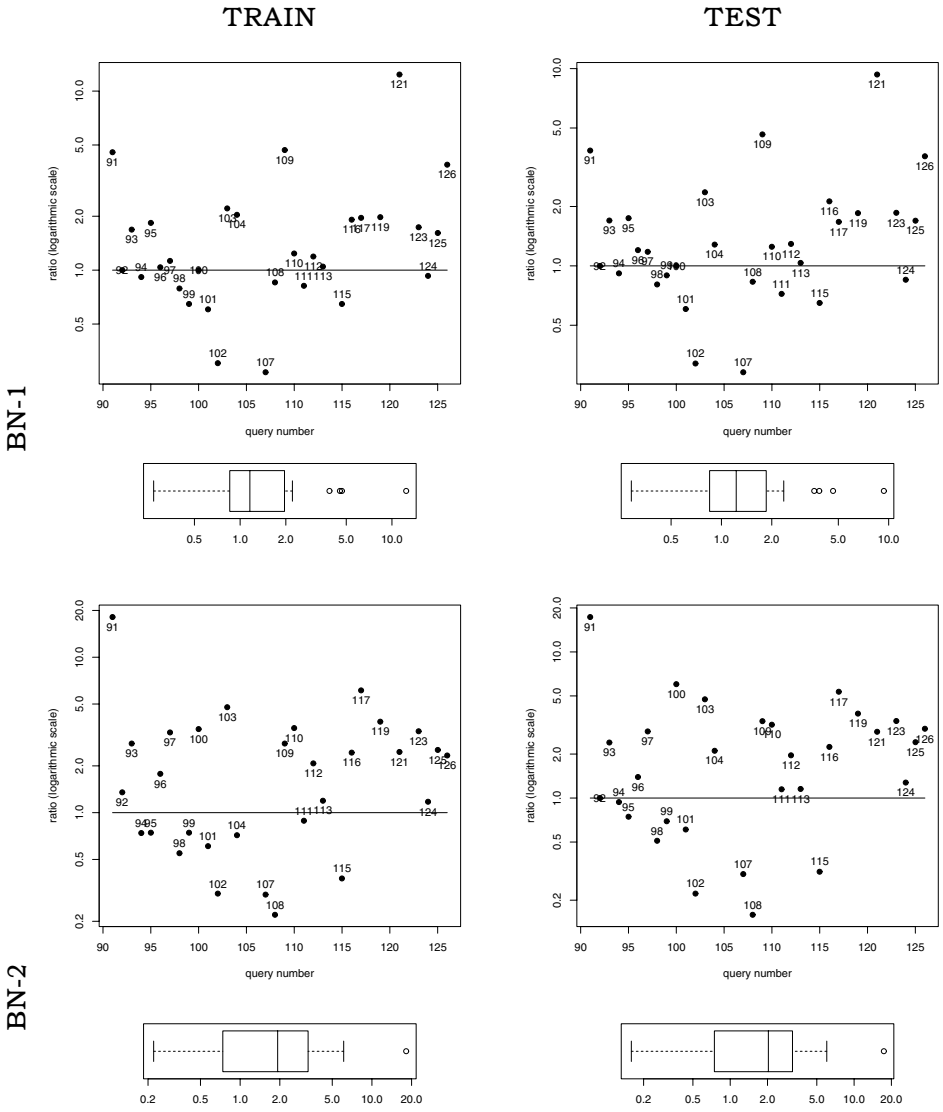


Figure 8. Plots correspond to RP_{hs} ratio between the BN models and Okapi-D measured for all 30 individual queries after 9000 iterations of the learning algorithm. Numbers on the x-axis for the plots correspond to the query identifiers used in INEX (integers between 90 and 125). The same information appears in a more synthetic way with the boxplots computed over all ratio values.

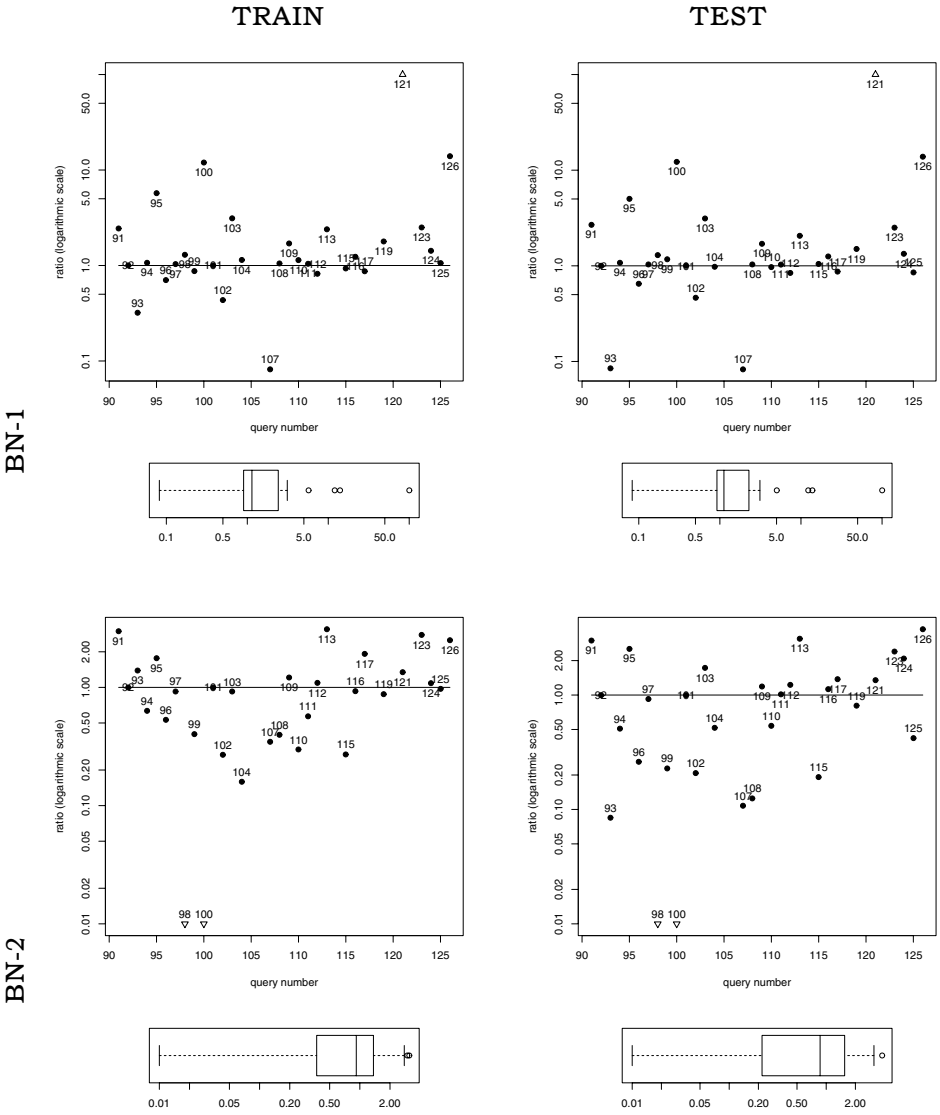


Figure 9. Plots correspond to ERR@50 ratio between the BN model and Okapi-D computed for all 30 individual queries after 9000 iterations of the learning algorithm (the ratios were bounded by 0.01 and 100) since for two queries (query 98 and 100) the ratio was 0 for BN-2 and above 100 for query 121 and BN-1. Numbers on the x-axis for the plots correspond to the query identifiers used in INEX (integers between 90 and 125). The same information appears in a more synthetic way with the boxplots computed over all ratio values.

with the view developed in this paper (no nested highly specific highly exhaustive doxels with the same exhaustivity). Learning will depend on the quality of the assessments and BN statistics will be learnt in accordance with the regularities observed in the majority of the assessments. Assessment variability which is still high for this collection causes variability in BNs scores.

We also performed a paired Student t-test for comparing Okapi and BN scores after 9000 learning iterations. With a 90% confidence interval, ERR@50 for BN-1 outperforms Okapi-D-T. For the other measures or models, there was no significant improvement at 90%. Paired t-test results should however be taken with care here since the number of queries is relatively small. In order to get a significant difference, results ought to be particularly stable.

To summarise the results obtained with these experiments, the proposed discriminative criterion is effective for training the BN model. The RP_{hs} measure is increased by learning in all cases, although the small number of queries and the variability of the scores do not allow matching the t-test. The BN performance significantly outperformed the baseline model with one of the proposed model according to the measure ERR@50 with a 90% confidence level. Performance does not significantly differ when using one or two baselines models except on a few queries.

8. 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 data. The training criterion is a discriminant criterion: cross-entropy between desired and target distributions of node values. The learning algorithm is an iterative gradient descent. We have focused here on the parameter learning problem for these models which appeared to be a non trivial machine learning problem. We have shown on the INEX collection that it was possible to improve retrieval performance through learning. More investigations are however still needed for the robust training of these models on large collections like INEX. However, these results are encouraging since traditional BN learning via Estimation Maximisation did not allow this type of performance increase.

Besides ranking doxels in their structural context and integrating different sources of evidence, the BN framework has additional advantages: 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 of its descendants; it can use simultaneously different baseline models: we can therefore use specific models for non textual media (image, sound, etc.) as another source of evidence. Another advantage of the BN framework is that we have more information on the relevance of each doxel than with any other model tested at INEX: we have a score which is composed of three probabilities (exact, too big and irrelevant) that can be translated back onto the INEX scale and that can also be used to filter the resulting list so as to avoid doxel overlap.

There are still several directions to explore with this model. The training criterion should better reflect the ranking task than the classification criterion which was used here. Preliminary small scale experiments with ranking criteria led to promising results, but this has still to be explored exhaustively. New evaluation criteria, better suited to SIR will

also influence the choice of a training criterion and the evaluation of SIR systems. The robustness of training algorithms with respect to the different training conditions (training corpus, number and quality of the assessments, number of queries needed to reach a reasonable performance) will also need further experiments. In order to get better targets for training, assessments should be pre-processed in order to remove incoherent assessments and the mapping from assessments to distributions *should depend* on neighbouring assessments. The method will have to be validated on other structured corpora when they will become available. The training algorithms themselves have to be refined for increasing the convergence speed and the robustness of training. The BN model should benefit from using different local baseline scorers with different behaviours. We only performed preliminary experiments in this direction using two local scorers. This also needs further investigations.

Appendix

We give here the derivation of the updating formula (7) for learning the parameters of the BN. Notations are the same as in Section 6. The parameter to be updated is θ. In the following, only BN variables associated to doxels are considered. Let us assume that for any node j the ancestors of X_j are the variables X_l where $l \in anc(j)$. We will denote $pa(k)$ the parent of the node k and $ANC(j)$ the set of ancestors including j, that is $ANC(j) = anc(j) \cup \{j\}$. We will also use the abreviation v_* for $X_* = v_*$ within probabilities. The P_Θ derivative in (6) decomposes as:

$$
\begin{aligned}
\frac{\partial P_\Theta(X_j = v_j \mid q)}{\partial \theta} &= \frac{\partial}{\partial \theta} \sum_{v_l \in \mathbf{V}, l \in anc(j)} \prod_{k \in ANC(j)} P_\Theta\big(v_k \mid v_{pa(k)}, q\big) \\
&= \sum_{\{v_l \in \mathbf{V}\}_{l \in anc(j)}} \sum_{l \in ANC(j)} \left(\prod_{k \in ANC(j)} P_\Theta(v_k \mid v_{pa(k)}, q) \right) \frac{\frac{\partial P_\Theta(v_l \mid v_{pa(l)}, q)}{\partial \theta}}{P_\Theta\big(v_l \mid v_{pa(l)}, q\big)} \\
&= \sum_{l \in ANC(j)} \sum_{v_l, v_{pa(l)} \in \mathbf{V}} \frac{\frac{\partial P_\Theta(v_l \mid v_{pa(l)}, q)}{\partial \theta}}{P_\Theta\big(v_l \mid v_{pa(l)}, q\big)} \sum_{\substack{\{v_l \in \mathbf{V}\}_l \\ l \in anc(j) \setminus \{l, pa(l)\}}} \prod_{k \in ANC(j)} P_\Theta\big(v_k \mid v_{pa(k)}, q\big) \\
&= \sum_{l \in ANC(j)} \sum_{v_l, v_{pa(l)} \in \mathbf{V}} \frac{\frac{\partial P_\Theta(v_l \mid v_{pa(l)}, q)}{\partial \theta}}{P_\Theta\big(v_l \mid v_{pa(l)}, q\big)} P(v_j, v_l, v_{pa(l)} \mid q) \qquad (9)
\end{aligned}
$$

And using Eq. (2), we get:

$$
\begin{aligned}
\frac{\partial P_\Theta\big(X_l = v_l \mid X_l = v_{pa(l)}\big)}{\partial \theta} = \sum_{m_1, \ldots, m_n} & \frac{\partial F_X\big(\Theta, v_l, v_{pa(l)}, m_1, \ldots, m_n\big)}{\partial \theta} \\
& \times \textstyle\prod_{s=1}^{n} P(M_s(X) = m_s \mid q) \qquad (10)
\end{aligned}
$$

with

$$\frac{\partial F_X(\Theta, v_l, v_m, m_1, \ldots, m_n)}{\partial \theta_{i(c,w_l,w_m,m'_1,\ldots,m'_n)}} = \begin{cases} 0 & \text{if } c_X \neq c \vee w_m \neq v_m \vee m_i \neq m'_i \\ F_X(\Theta, v_l, v_m, m_1, \ldots, m_n) & \\ \quad \times (1 - F_X(\Theta, v_l, v_m, m_1, \ldots, m_n)) & \text{if } w_l = v_l \\ -F_X(\Theta, v_l, v_m, m_1 \ldots m_n) & \\ \quad \times F_X(\Theta, w_l, v_m, m_1, \ldots, m_n) & \text{else} \end{cases} \tag{11}$$

The updating formula for θ is then easily obtained from (9), (10) and (11). The probability $P_\Theta(X_j = v_j, X_l = v_l, X_{pa(l)} = v_{pa(l)} \mid q)$ can be easily computed by our BN.

Acknowledgment

This work was supported in part by the IST Programme of the European Community, under the PASCAL Network of Excellence, IST-2002-506778. This publication only reflects the authors' views.

Notes

1. http://inex.is.informatik.uni-duisburg.de
2. ACM SIGIR 200 Workshop on XML and Information Retrieval, Athens, Greece, July 2000.
3. Workshop on XML and Information Retrieval, Tampere, Finland, August 2002.
4. Joint workshops on XML, Information Retrieval and Databases, Sheffield, UK, 2004.
5. More precisely, the probability that the document represents the query.
6. Only highly specific elements are taken into account with this measure; this measure is simply the classical recall-precision curve where relevant documents are highly specific doxels.
7. We used the default values for the different Okapi parameters (k_3 is 1.2, k_3 is 7 and b is 0.75).
8. These parameters were estimated on INEX 2002, i.e. on a corpus different from the INEX 2003 corpus which has been used in our experiments described in Section 7.
9. The boxplots synthesise a set of real values using 5 values: the min value, the three quartiles and the max value. The five vertical lines from left to right (figures 8 and 9) correspond respectively to these 5 values. The two extreme lines are min and max, the box lines are the quartiles. We used here modified boxplots where outliers appear separately and not in the boxplot.

References

Callan JP, Croft WB and Harding SM (1992) The INQUERY retrieval system. In: Min Tjoa A and Isidro Ramos, Eds., Database and Expert Systems Applications, Proceedings of the International Conference, Valencia, Spain. Springer-Verlag, pp. 78–83.

Crestani F, de Campos LM, Fernández-Luna JM and Huete JF (2003) A multi-layered bayesian network model for structured document retrieval. In: Nielsen TD and Zhang NL, Eds., Symbolic and Quantitative Approaches to Reasoning with Uncertainty: 7th European Conference, ECSQARU 2003, Aalborg, Denmark, Springer-Verlag, pp. 74–86.

Crestani F, de Campos LM, Fernández-Luna JM and Huete JF (2003a) Ranking structured documents using utility theory in the bayesian network retrieval model. In: Nascimento MA, de Moura ES and Oliveira AL, Eds.

SPIRE(String Processing and Information Retrieval) 2003, volume 2857 of Lecture Notes in Computer Science, Brazil, Springer-Verlag Heidelberg, pp. 168–182.

Culioli J-C (1994) Introduction à l'optimisation. Ellipses.

de Campos LM, Fernández-Luna JM and Huete JF (2003b) The BNR model: Foundations and performance of a bayesian network-based retrieval model. International Journal of Approximate Reasoning, 34(2):265–285.

de Campos LM, Fernández-Luna JM and Huete JF (2003) Improving the efficiency of the bayesian network retrieval model by reducing relationships between terms. International Journal of Uncertainty Fuzziness Knowledge-Based Systems, 11(Supplement):101–116.

Dempster AP, Laird NM and Rubin DB (1977) Maximum likelihood from incomplete data via de EM algorithm. The Journal of Royal Statistical Society, 39:1–37.

Fuhr N and Malik S (2003) Overview of the initiative for the evaluation of XML retrieval (INEX 2003). In: INitiative for the Evaluation of XML Retrieval (INEX). Proceedings of the Second INEX Workshop.

Fuhr N and Rölleke T (1998) Hyspirit—a probabilistic inference engine for hypermedia retrieval in large databases. In: Schek H-J, Saltor F, Ramos I and Alonso G, Eds., Proceedings of the 6th International Conference on Extending Database Technology (EDBT), Valencia, Spain, Springer, Berlin.

Gövert N and Kazai G (2002) Overview of the initiative for the evaluation of XML retrieval (INEX 2002). In: Proceedings of the First Annual Workshop of the Initiative for the Evaluation of XML retrieval (INEX), DELOS workshop, Dagstuhl, Germany, ERCIM.

Gövert N, Kazai G, Fuhr N and Lalmas M (2003) Evaluating the effectiveness of content-oriented XML retrieval. Technical report, University of Dortmund, Computer Science 6.

Indrawan M, Ghazfan D and Srinivasan B (1994) Using bayesian networks as retrieval engines. In: ACIS 5th Australasian Conference on Information Systems, Melbourne, Australia, pp. 259–271.

Jensen FV (1996) An Introduction to Bayesian Networks. UCL Press, London, England.

Kazai G (2003) Report on the INEX 2003 metrics group. In: INitiative for the Evaluation of XML Retrieval (INEX). Proceedings of the Second INEX Workshop.

Kazai G, Lalmas M and Vries AP (2004) The overlap problem in content-oriented XML retrieval evaluation. In: INitiative for the Evaluation of XML Retrieval (INEX). Proceedings of the Second INEX Workshop.

Krause PJ (1998) Learning probabilistic networks. The Knowledge Engineering Review, 13(4):321–351.

Lalmas M (1997) Dempster-shafer's theory of evidence applied to structured documents: Modelling uncertainty. In: Proceedings of the 20th Annual International ACM SIGIR, Philadelphia, PA, USA, ACM, pp. 110–118.

Myaeng SH, Jang D-H, Kim M-S and Zhoo Z-C (1998) A flexible model for retrieval of SGML documents. In: Croft WB, Moffat A, van Rijsbergen CJ, Wilkinson R and Zobel J, Eds., Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Melbourne, Australia, ACM Press, New York, pp. 138–140.

Pearl J (1988) Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference. Morgan Kaufmann.

Piwowarski B and Gallinari P (2003) Expected ratio of relevant units: A measure for structured information retrieval. In: Fuhr N, Lalmas M and Malik S, Eds., INitiative for the Evaluation of XML Retrieval (INEX). Proceedings of the Second INEX Workshop, Dagstuhl, France.

Piwowarski B and Lalmas M (2004) Providing consistent and exhaustive relevance assessments for XML retrieval evaluation. In: Proceedings of the Thirteenth Conference on Information and Knowledge Management (CIKM 2004), Washington D.C., USA.

Ribeiro BAN and Muntz R (1996) A belief network model for IR. In: Proceedings of the 19th ACM-SIGIR Conference, pp. 253–260.

Robertson SE (2002) Threshold setting and performance optimization in adaptive filtering. Information Retrieval, 5(2/3):239–256.

Walker S and Robertson SE (1999) Okapi/keenbow at TREC-8. In: Voorhees EM and Harman DK, Eds., NIST Special Publication 500-246: The Eighth Text REtrieval Conference (TREC-8), Gaithersburg, Maryland, USA.

Wilkinson R (1994) Effective retrieval of structured documents. In: Croft WB and van Rijsbergen CJ, Eds., Proceedings of the 17th Annual International Conference on Research and Development in Information Retrieval, Dublin, Ireland: Springer-Verlag, pp. 311–317.