ELSEVIER

Information Processing and Management 40 (2004) 829–847

INFORMATION PROCESSING & MANAGEMENT

www.elsevier.com/locate/infoproman

Using context information in structured document retrieval: an approach based on influence diagrams ☆

Luis M. de Campos, Juan M. Fernández-Luna *, Juan F. Huete

Departamento de Ciencias de la Computación e Inteligencia Artificial, E.T.S.I. Informática, Universidad de Granada, C.P. 18071, Granada, Spain

Available online 4 June 2004

Abstract

In this paper we present an Information Retrieval System (IRS) which is able to work with structured document collections. The model is based on the influence diagrams formalism: a generalization of Bayesian Networks that provides a visual representation of a decision problem. These offer an intuitive way to identify and display the essential elements of the domain (the structured document components and their usefulness) and also how these are related to each other. They have also associated quantitative knowledge that measures the strength of the interactions. By means of this approach, we shall present structured retrieval as a decision-making problem. Two different models have been designed: SID (Simple Influence Diagram) and CID (Context-based Influence Diagram). The main difference between these two models is that the latter also takes into account influences provided by the context in which each structural component is located.

Keywords: Bayesian networks; Influence diagrams; Structured documents; Retrieval model; Decision theory

1. Introduction

Documents such as textbooks, scientific articles, technical manuals, etc. have two main characteristics: on the one hand, the set of terms used to describe their contents, and on the other, a well-defined structure to organize these contents comprehensibly and improve readability for the user. Since both characteristics are quite important and must be taken into account when writing a document, it is natural to develop tools capable of representing these documents properly. New formalisms to manage documents, such as HTML, XML or MPEG-7, have been designed to tackle this problem, being capable of representing both the content and the structure of a document.

☆ This work has been jointly supported by the Spanish Fondo de Investigación Sanitaria and Consejería de Salud de la Junta de Andalucía, under Projects PI021147 and 177/02, respectively.

* Corresponding author.

E-mail addresses: lci@decsai.ugr.es (L.M. de Campos), jmfluna@decsai.ugr.es (J.M. Fernández-Luna), jhg@decsai.ugr.es (J.F. Huete).

0306-4573/$ - see front matter © 2004 Elsevier Ltd. All rights reserved.
doi:10.1016/j.ipm.2004.04.014

In general, when a document is represented by means of content and also structure, it is called a *structured document*, in contrast to the (plain) classical representation of a document. Nowadays, these formalisms are widely used to manage electronic documents and, as a result, more and more structured documents have become available. This structural knowledge can be exploited when designing an Information Retrieval System (IRS) (Chiaramella, 2001) in order to obtain a more intelligent performance and therefore one which better satisfies the user's needs. The aim of the system is therefore to retrieve the set of document components which are most relevant to a query (not necessarily an entire document). In order to achieve this aim, it is necessary to design and implement models and tools to index, retrieve, and present documents according to the given structure.

Classical IRSs consider each document as an atomic item of information, which is handled in isolation. For example, a probabilistic IRS ranks the documents by considering their probability of relevance to a given query. These values are usually computed without taking into account the rest of the documents in the collection. Therefore, the action of presenting (or not presenting) a document to the user is independent of the action of presenting (or not presenting) any other document.

Although this procedure is valid for classical document collections, it cannot be applied properly when dealing with structured documents. Now, the situation is different: *rather than being interested in retrieving a whole document, we might be interested in retrieving document units or components.* Once the relevance probability of each structural unit has been computed and a ranking with all of these has been generated, the user can retrieve redundant information. For example, let us suppose that the top of the ranking of units is composed of the three subsections of a section from the same article, and the fourth item is the section itself. The system should detect this situation and decide to show either these three subsections or only the section, but not the four units. In this case, it would probably be better to retrieve only the whole section. It is clear that the decision to show a document unit affects the retrieval of other units.

One possible solution to these problems is to make a decision about what to retrieve, depending not only on the probability of relevance of the units but also in terms of the *usefulness* of these units for the user and what has been previously retrieved. For instance, if a user is interested in the Runge–Kutta method for numerical integration of ordinary differential equations, they would not be interested in either a whole book being retrieved or only the five equations defining the method being returned. What the user would be interested in, however, would be certain parts of the book where the formulas themselves were accompanied by additional information so that they could be understood (notation, description of the method, etc.). In other words, what it is being pursued is to provide an IRS with the capacity to automatically select those more appropriate units (the *best entry points* to the relevant documents).

In an attempt to solve these problems, we propose a model which is capable of making decisions, i.e. the system should be able to determine those units to be retrieved, not only depending on the probability of relevance, but also on the *utility* of these units for the user (possibly taking into account the user's own preferences). This will be carried out by means of an *Influence diagram* (Jensen, 2001), a generalization of the well founded Bayesian network formalism (Pearl, 1988). Bayesian networks have been successfully applied for dealing with classical Information Retrieval (IR) (de Campos, Fernández-Luna, & Huete, 2003a; Ribeiro-Neto & Muntz, 1996; Turtle & Croft, 1990) and also for structured Information Retrieval (Crestani, de Campos, Fernández-Luna, & Huete, 2003; Graves & Lalmas, 2002; Myaeng, Jang, Kim, & Zhoo, 1998; Piwowarski, Faure, & Gallinari, 2002). Influence diagrams can show the structure of a decision problem, since they are an intuitive, effective formalism for representing, understanding and explaining the decision models, and have a high expressive power.

In this paper, Section 2 introduces basic concepts about influence diagrams and the type of documents that we are going to work with. In Section 3, we present the two proposed influence diagram models: SID (simple influence diagram) and CID (context-based influence diagram). Section 4 shows how an influence diagram can be solved and how to apply the results obtained after this process to structured IR. The experimental results will be discussed in Section 5 and finally, Section 6 presents the conclusions of the paper.

2. Preliminaries

2.1. Influence diagrams

An influence diagram (Jensen, 2001; Shachter, 1988) provides a simple notation for creating decision models by clarifying the qualitative issues of the factors which need to be considered and how they are related, i.e. an intuitive representation of the model. They also have associated an underlying quantitative representation in order to measure the strength of the relationships: we can quantify uncertain interactions between random variables and also the decision maker's options and preferences. The model is used to determine the optimal decision policy.

More formally, an influence diagram is an acyclic directed graph containing three types of nodes (decision, chance, and value nodes) and two types of arcs (influence and informative arcs).

Nodes in an influence diagram represent various types of variables.

- *Decision nodes*. Usually drawn as rectangles, these represent variables that the decision maker controls directly. These variables model the decision alternatives available for the decision maker.
- *Chance nodes*. Usually drawn as circles, these represent random variables, i.e. uncertain quantities that are relevant to the decision problem and cannot be controlled directly. They are quantified by means of conditional probability distributions, identical to those used in Bayesian networks. [1] Predecessors (parents) of chance nodes that are decision nodes act in exactly the same way as those predecessors that are chance nodes—they index the conditional probability tables of the child node.
- *Utility nodes*. Usually drawn as diamonds, these represent utility, i.e. they express the profit or the preference degree of the consequences derived from the decision process. They are quantified by the utility of each of the possible combinations of outcomes of their parent nodes.

There are also different types of arcs in an influence diagram, which generally represent influence. The arcs between chance nodes represent probabilistic dependences (as occurs in Bayesian networks). The arcs from a decision node to a chance node or to a utility node establish that the future decision will affect the value of the chance node or the profit obtained, respectively. Arcs between a chance node and a decision node (also called *informative*) only say that the value of the chance node will be known at the moment of making the decision. Finally, arcs from a chance node to a utility node will represent the fact that the profit depends on the value that this chance node takes. The absence of an arc between two nodes specifies (conditional) independence relationships. It should be noted that the absence of an arc is a stronger statement than the presence of an arc, which only indicates the possibility of dependence.

Some arcs in influence diagrams clearly have a causal meaning. In particular, a directed path from a decision node to a chance node means that the decision will influence that chance node, in the sense of changing its probability distribution.

A simple example of an influence diagram appears in Fig. 1. It has two chance nodes, F and W, representing, the weather forecast in the morning (sunny, cloudy or rainy), and whether it actually rains during the day (rain or no-rain), respectively. It has one decision node U, take an umbrella (with possible values true or false). The utility node measures the decision maker's satisfaction.

With each chance node X in the graph, the quantitative part of an influence diagram associates a set of conditional probability distributions $p(X \mid pa(X))$, one for each *configuration* $pa(X)$ from the *parent set* of X in the graph, $Pa(X)$, i.e. for each allocation of values to all the variables in the parent set of X. If X has no

[1] In fact, the subset of an influence diagram that consists only of chance nodes is a Bayesian network, i.e., an influence diagram can also be viewed as a Bayesian network enlarged with decision and utility nodes.

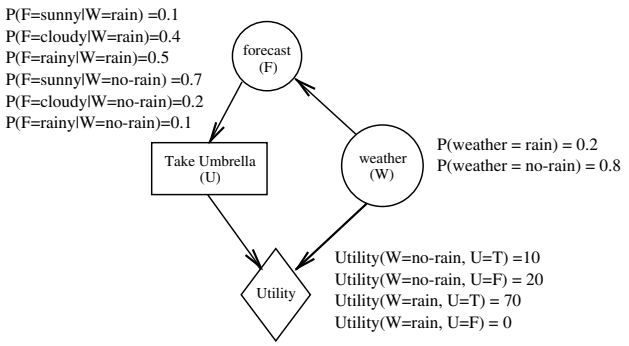

Fig. 1. An example of an influence diagram.

parents ($Pa(X) = \emptyset$), then $p(X \mid pa(X))$ equals $p(X)$. For each utility node V, a set of utility values $v(pa(V))$ is associated, specifying for each combination of values for the parents of V, a number expressing the desirability of this combination for the decision maker.

The goal of influence diagram modeling is to choose the decision alternative that will lead to the highest expected gain (utility), i.e. to find the *optimal policy* (Shachter, 1986; Zhang, 1998). In order to compute the solution, for each sequence of decisions, the utilities of its uncertain consequences are weighted with the probabilities that these consequences will occur.

2.2. The document sources

In this section, we introduce the type of structured documents that we shall consider. We start with a document collection comprising N documents, $\mathscr{D} = \{D_1, \ldots, D_N\}$, and the set $\mathscr{T} = \{T_1, \ldots, T_M\}$ of the M terms or concepts used to index these documents (the glossary of the collection). $A(D_i)$ will denote the subset of terms in $\mathscr{T}$ that are used to index the document D_i.

We shall assume that each document D is composed of a hierarchical structure of ℓ_D abstraction *levels*, $\mathscr{L}_1, \mathscr{L}_2, \ldots, \mathscr{L}_{\ell_D}$, each representing a structural association of elements in the document. For instance, sections, subsections, and paragraphs in the context of a collection of structured scientific articles, or scenes, shots, and frames in MPEG-7 videos. Each level contains a set of *structural units* of a given degree of specificity (such as Sections 3, 3.2, or paragraph 3.2.15). The level in which the document itself is included will be denoted by level 1 ($\mathscr{L}_1$), and the more specific level by $\mathscr{L}_{\ell_D}$. In order to simplify the notation, we assume that the number of levels, ℓ_D, is the same for all the documents in the collection ($\ell_D = \ell$).

Each structural unit will be denoted by $U_{i,j}$, where i is the identifier of that unit in the level j. The number of structural units contained in each level $\mathscr{L}_j$ is represented by $|\mathscr{L}_j|$. Therefore, $\mathscr{L}_j = \{U_{1,j}, \ldots, U_{|\mathscr{L}_j|,j}\}$. The units are organized according to the current structure of the document: every unit $U_{i,j}$ at level j, except the unit at level $j = 1$ (i.e. the complete document $D_i = U_{i,1}$), is related to only one unit $U_{z(i,j),j-1}$ of the lower level $j - 1$. [2] As the text (the whole set of terms) associated to $U_{i,j}$ is part of the text associated to $U_{z(i,j),j-1}$, using notation, we shall note this relation as $U_{i,j} \subseteq U_{z(i,j),j-1}$.

Each term $T_k \in A(D_i)$, originally indexing a document D_i, will be assigned to those units in level $\mathscr{L}_\ell$ containing it which are associated with D_i. Therefore, only the units in level $\mathscr{L}_\ell$ will be indexed, having associated several terms describing their content. Consequently, each structured document may be represented as a tree (Fig. 2 shows an example).

[2] $z(i,j)$ is a function that returns the index of the unit in level $j - 1$ to which the unit with index i in level j belongs.

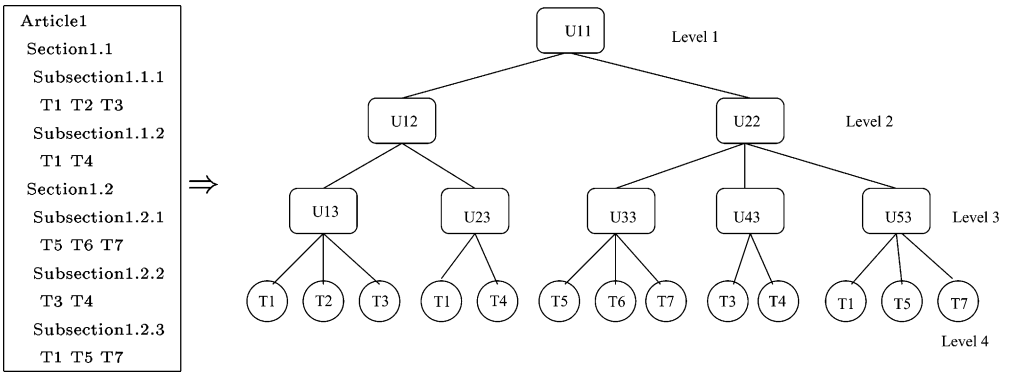

Fig. 2. Structured representation of a document.

3. The influence diagram-based retrieval model for structured documents

An influence diagram consists of a qualitative and a quantitative part. The qualitative part, represented by a directed acyclic graph, with nodes representing the variables of the decision problem to solve, and arcs indicating causality, relevance, dependence or influence relationships between the variables. The quantitative component is encoded by means of both conditional probability distributions and utility values. In order to describe our model, we shall therefore present each of these parts.

3.1. The qualitative component: representing variables and influences

First, the different types of nodes to be used:

- *Chance nodes*. Our influence diagram will contain two different sets of chance nodes: those associated to structural units, $\mathscr{U} = \cup_{j=1}^{\ell} \mathscr{L}_j$; and those related to terms, $\mathscr{T}$. Each node represents a binary random variable: $U_{i,j}$ takes its values in the set $\{u_{i,j}^-, u_{i,j}^+\}$, representing that the unit is not relevant and is relevant, respectively; [3] T_i takes its values from the set $\{t_i^-, t_i^+\}$, where in this case t_i^- stands for 'the term T_i is not relevant', and t_i^+ represents 'the term T_i is relevant'. [4] In order to denote a generic, unspecified value of a term variable T_i or a unit variable $U_{i,j}$, we shall use lower-case letters, t_i and $u_{i,j}$. It should be noted that we employ the notation T_i ($U_{i,j}$, respectively) to refer to the term (unit, respectively) and also to its associated variable and corresponding node.
- *Decision nodes*. These nodes model the decision variables, representing the possible alternatives available to the decision maker. In our case, we consider one decision node, $R_{i,j}$, for each structural unit $U_{i,j} \in \mathscr{L}_j$, $\forall j = 1, \ldots, \ell$, $\forall i = 1, \ldots, |\mathscr{L}_j|$. $R_{i,j}$ represents the decision variable related to whether or not to show the content of $U_{i,j}$ to the user. The two different values for $R_{i,j}$ are $r_{i,j}^+$ and $r_{i,j}^-$, meaning 'retrieve $U_{i,j}$' and 'do not retrieve $U_{i,j}$', respectively.
- *Utility nodes*. We shall also consider one utility node, $V_{i,j}$, for each structural unit $U_{i,j} \in \mathscr{L}_j$, $\forall j = 1, \ldots, \ell$, $\forall i = 1, \ldots, |\mathscr{L}_j|$. $V_{i,j}$ will measure the value of utility of the corresponding decision.
  We shall also consider a utility node that represents the joint utility of the whole model. This node will be denoted by $\sum$, representing the fact that we are assuming an additive behavior of the model.

[3] A unit is relevant for a given query if it satisfies the user's information need expressed by means of this query.
[4] A term is relevant in the sense that the user believes that this term will appear in relevant documents.

In order to complete the qualitative component, it is necessary to describe the topology of the model, i.e. the set of arcs used to represent the relationships between the variables described above.

- *Arcs pointing to chance nodes*. These arcs encode the dependences between the represented statistical variables. The absence of an arc between two nodes means that the corresponding variables do not influence each other directly and hence are (conditionally) independent.
  In our case, we shall consider that there is an arc from a given chance node (either a term or structural unit) to the particular structural unit node it belongs to. In this case, we are expressing the fact that the probability of relevance of a given structural unit to the query will depend on the relevance values of the different elements (units or terms) that comprise it. It should be noted that with this criteria, term nodes have no parents, i.e. they are root nodes, expressing the fact that they are the most specific items of information in the model. We do not include arcs from decision nodes to chance nodes, since these arcs express an influence on the chance node exerted by the decision maker. In our case, the degree of relevance for a structural unit is independent of the possible decision about whether this unit should be retrieved or not.
  The skeleton representing the structured collection therefore has a graph topology with $\ell + 1$ layers, where the arcs go from term nodes to structural units in level ℓ, and from units in level j to units in level $j-1$, $j = 2, \ldots, \ell$. More formally, the network is characterized by the following parent sets, $Pa(\cdot)$, for each type of node:
  - $\forall T_k \in \mathscr{T}, Pa(T_k) = \emptyset$.
  - $\forall U_{i,\ell} \in \mathscr{L}_\ell, Pa(U_{i,\ell}) = \{T_k \in \mathscr{T} \mid U_{i,\ell} \text{ is indexed by } T_k\}$.
  - $\forall U_{i,j} \in \mathscr{L}_j, j = 1, \ldots, \ell - 1, Pa(U_{i,j}) = \{U_{h,j+1} \in \mathscr{L}_{j+1} \mid U_{h,j+1} \subseteq U_{i,j}\}$.

  An example of this multi-layer topology is depicted in Fig. 3, for $\ell = 3$, where for clarity reasons, we only display chance nodes. It is interesting to note that the set of chance nodes and the arcs into these nodes constitute a Bayesian network model representing the dependence in a structured document collection. In Crestani et al. (2003), we studied the application of this model, called the BNR-SD model, to the problem of structured document retrieval.
- *Arcs pointing to decision nodes*. These arcs would indicate that the value of the source node is available when the decision is made, and thus may affect the decision. But in our case, it is impossible to know the specific relevance values for the nodes in the collection before the decision is made, and therefore there are no arcs pointing to decision nodes.
- *Arcs pointing to utility nodes*. These arcs will be used to indicate which variables have a direct influence on the desirability of a given decision, i.e. the profit obtained will depend on the value of these variables.

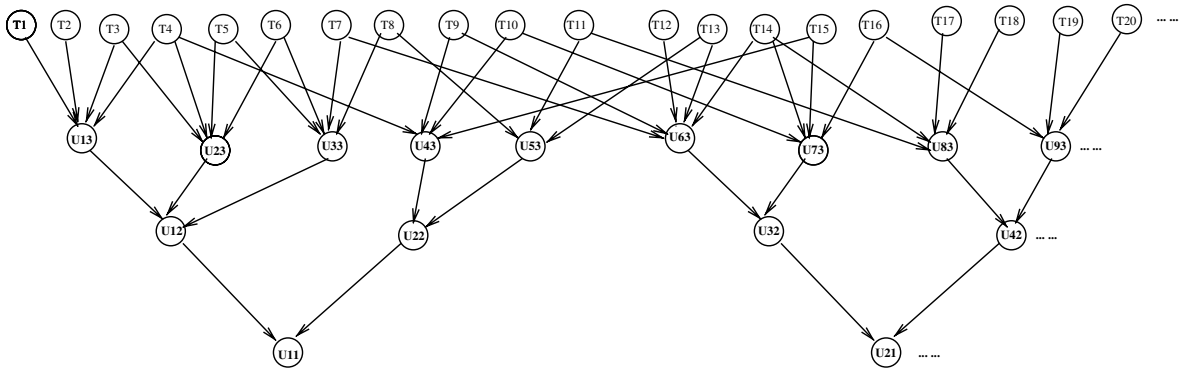

Fig. 3. Arcs into chance nodes.

We shall consider two different set of arcs, which will consistently generate two different influence diagrams models:

(1) *Simple influence diagram (SID)*. We shall only take into account arcs from chance nodes $U_{i,j}$ and decision nodes $R_{i,j}$ to the utility nodes $V_{i,j}$, $\forall j = 1, \ldots, \ell$, $\forall i = 1, \ldots, |\mathscr{L}_j|$. These arcs mean that the utility function of $V_{i,j}$ obviously depends on the decision made and the relevance value of the structural unit considered.
Finally, the utility node $\sum$ has all the utility nodes $V_{i,j}$ as its parents. These arcs represent the fact that the joint utility of the model will depend on the values of the individual utilities of each structural unit.

(2) *Context-based influence diagram (CID)*. In this case, the model inherits all the arcs from the SID, but also includes new arcs to consider context information. In particular, arcs from $U_{z(i,j),j-1}$ (the unique structural unit containing $U_{i,j}$) to $V_{i,j}$, $\forall j = 2, \ldots, \ell$, $\forall i = 1, \ldots, |\mathscr{L}_j|$ will be added. These arcs mean that the utility of the decision as to whether or not to retrieve a unit $U_{i,j}$ also depends on the relevance of the unit which contains it. The units in level 1 (the whole documents), which are not contained in any other unit, are an exception.
This kind of arc is very important since it allows us to represent the context-based information and can avoid redundant information being shown to the user. For instance, we could express the fact that on the one hand, if $U_{i,j}$ is relevant and $U_{z(i,j),j-1}$ is not, then the utility of retrieving should be large (and the one of not retrieving null). On the other hand, if $U_{z(i,j),j-1}$ is relevant, even if $U_{i,j}$ were also relevant, the utility of retrieving $U_{i,j}$ should be small, because in this case, it would be preferable to retrieve the largest unit as a whole, instead of retrieving each of its components separately.
The utility node $\sum$ will have the same set of parents as in the SID model.
Fig. 4 shows an example of both influence diagram models: the SID (left-hand side) and the CID (right-hand side).

Once the topology of the influence diagram has been established, the quantitative information must be specified. We need to assess the set of (conditional) probabilities for chance nodes (terms and structural units) and the utility functions associated to each utility node.

3.2. The quantitative component: probabilities in chance nodes

For each chance node X in the graph, we need to define a set of conditional probability distributions $p(x|pa(X))$, one for each configuration $pa(X)$ of the parent set of X in the graph, $Pa(X)$.

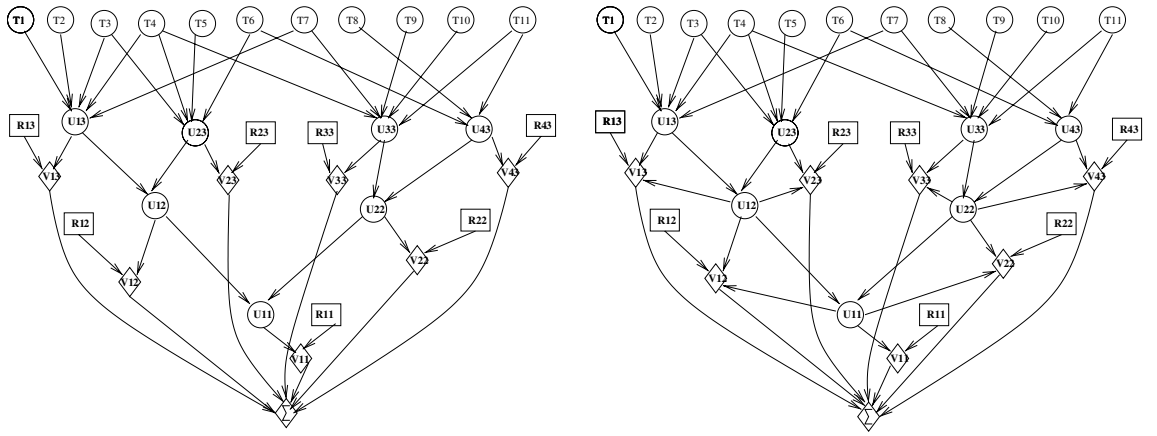

Fig. 4. Influence diagrams for the SID and CID models.

- Term nodes T_k. Since term nodes do not have parents, $Pa(T_k) = \emptyset$, hence $p(t_k \,|\, pa(T_k))$ equals $p(t_k)$. Therefore, they store marginal probabilities. We assume that these probabilities are identical for all the term nodes, $p(t_k^+) = p_0$ and $p(t_k^-) = 1 - p_0$, $\forall T_k \in \mathscr{T}$. The parameter p_0 used in this paper is $p_0 = 0.5$.
- Structural units $U_{i,j}$. We must consider two different situations: the structural units at level ℓ, having a subset of terms as their parents, and the structural units at level j, $j \neq \ell$, having other structural units as their parents. We must therefore assess $p(u_{i,\ell} \,|\, pa(U_{i,\ell}))$ and $p(u_{i,j} \,|\, pa(U_{i,j}))$, $j \neq \ell$. For each situation, the following canonical model is considered:

$$p(u_{i,\ell}^+ \,|\, pa(U_{i,\ell})) = \sum_{T_k \in R(pa(U_{i,\ell}))} w(T_k, U_{i,\ell}), \tag{1}$$

$$p(u_{i,j}^+ \,|\, pa(U_{i,j})) = \sum_{U_{h,j+1} \in R(pa(U_{i,j}))} w(U_{h,j+1}, U_{i,j}), \tag{2}$$

where $w(T_k, U_{i,\ell})$ is a weight associated to each term T_k indexing the unit $U_{i,\ell}$, $w(U_{h,j+1}, U_{i,j})$ is a weight measuring the importance of the unit $U_{h,j+1}$ within $U_{i,j}$, with $w(T_k, U_{i,\ell}) \geqslant 0$ and $w(U_{h,j+1}, U_{i,j}) \geqslant 0$. In either case, $R(pa(U))$ is the subset of parents of U (terms for $j = \ell$, units in level $j+1$ for $j \neq \ell$) that are instantiated as relevant in the configuration $pa(U)$, i.e. $R(pa(U_{i,\ell})) = \{T_k \in Pa(U_{i,\ell}) \,|\, t_k^+ \in pa(U_{i,\ell})\}$ and $R(pa(U_{i,j})) = \{U_{h,j+1} \in Pa(U_{i,j}) \,|\, u_{h,j+1}^+ \in pa(U_{i,j})\}$. Therefore, the greater the number of relevant parents in U, the greater the relevance probability of U. For example, for the unit $U_{1,3}$ in the model displayed in Fig. 4, $Pa(U_{1,3}) = \{T_1, T_2, T_3, T_4, T_7\}$; if the configuration $pa(U_{1,3}) = \{t_1^+, t_2^-, t_3^+, t_4^-, t_7^-\}$, then $p(u_{1,3}^+ \,|\, pa(U_{1,3})) = w(T_1, U_{1,3}) + w(T_3, U_{1,3})$.

Before defining the weights $w(T_k, U_{i,\ell})$ and $w(U_{h,j+1}, U_{i,j})$ in Eqs. (1) and (2), let us introduce some additional notation: for any unit $U_{i,j} \in \mathscr{U}$, let $A(U_{i,j}) = \{T_k \in \mathscr{T} \,|\, T_k \text{ is an ancestor of } U_{i,j}\}$, i.e., $A(U_{i,j})$ is the set of terms that are included in the unit $U_{i,j}$. [5] For example, for the model displayed in Fig. 4, $A(U_{1,2}) = \{T_1, T_2, T_3, T_4, T_5, T_6, T_7\}$ and $A(U_{2,3}) = \{T_3, T_4, T_5, T_6\}$. Let $tf_{k,C}$ be the *frequency* of the term T_k (number of times that T_k occurs) in the set of terms C and idf_k be the *inverse document frequency* of T_k in the whole collection. We shall use the weighting scheme $\rho(T_k, C) = tf_{k,C} \cdot idf_k$. We define

$$\forall U_{i,\ell} \in \mathscr{L}_\ell,\ \forall T_k \in Pa(U_{i,\ell}), \quad w(T_k, U_{i,\ell}) = \frac{\rho(T_k, A(U_{i,\ell}))}{\sum_{T_h \in A(U_{i,\ell})} \rho(T_h, A(U_{i,\ell}))}, \tag{3}$$

$$\forall j = 1, \ldots, \ell - 1,\ \forall U_{i,j} \in \mathscr{L}_j,\ \forall U_{h,j+1} \in Pa(U_{i,j}),$$
$$w(U_{h,j+1}, U_{i,j}) = \frac{\sum_{T_k \in A(U_{h,j+1})} \rho(T_k, A(U_{h,j+1}))}{\sum_{T_k \in A(U_{i,j})} \rho(T_k, A(U_{i,j}))}. \tag{4}$$

It should be observed that the weights in Eq. (3) are only the classical tfidf weights, normalized to sum one up. The weights $w(U_{h,j+1}, U_{i,j})$ in Eq. (4) measure, to a certain extent, the proportion of the content of the unit $U_{i,j}$ which can be attributed to each one of its components.

3.3. The quantitative component: utilities in utility nodes

For each node $V_{i,j}$, the associated utility functions must be defined. We shall always consider normalized utility values for these nodes, i.e. all the values will belong to the interval $[0, 1]$. The reason for this is that

[5] Two things should be noted: first, that although a unit $U_{i,j}$ in level $j \neq \ell$ is not connected directly to any term, it does contain all the terms indexing structural units in level ℓ that are included in $U_{i,j}$; and secondly, $A(U_{i,\ell}) = Pa(U_{i,\ell})$.

the result of evaluating an influence diagram is invariable with respect to changes in the scale of the utilities. Previously, we shall discuss the utility values at node $\sum$, common to both models. Since we have assumed that the joint utility of the model is additive, this value shall be computed as the sum of the individual utilities associated to each node $V_{i,j}$.

(1) *Utility nodes in SID*. For each node $V_{i,j}$, we need to assess a numeric value that represents the utility for the corresponding combination of the decision node $R_{i,j}$ and the chance node representing the structural component $U_{i,j}$. Table 1 displays the four values required to specify the utility function for $V_{i,j}$.
We shall present general guidelines that can be used to assign these utility values. For a given unit $U_{i,j}$, the best situation is clearly for a relevant unit to be retrieved, and the worst situation, for a relevant unit to be hidden. We therefore fix $v(r^+_{i,j}\,|\,u^+_{i,j}) = 1$ and $v(r^-_{i,j}\,|\,u^+_{i,j}) = 0$. If $U_{i,j}$ is not relevant, it is obvious that not showing it is better than showing it. Then $v(r^-_{i,j}\,|\,u^-_{i,j}) \geqslant v(r^+_{i,j}\,|\,u^-_{i,j})$. Therefore, a complete ordering for the utilities is

$$1 = v(r^+_{i,j}\,|\,u^+_{i,j}) \geqslant v(r^-_{i,j}\,|\,u^-_{i,j}) \geqslant v(r^+_{i,j}\,|\,u^-_{i,j}) \geqslant v(r^-_{i,j}\,|\,u^+_{i,j}) = 0. \tag{5}$$

(2) *Utility nodes in CID*. We shall distinguish between the levels $s \neq 1$ and level 1. Focusing our attention on any node $V_{i,s}$ from level $s \neq 1$, the utility of decision $R_{i,s}$ for the corresponding combination of relevance values of $U_{i,s}$ and $U_{z(i,s),s-1}$, must be determined (a total of eight numerical values are required to specify the utility function of $V_{i,s}$). The utility nodes $V_{i,1}$, associated to the units in level 1 (representing the whole documents), have only two parents $R_{i,1}$ and $U_{i,1}$, so that in this situation, we shall proceed in a similar way to the one used in the SID model. Table 2 displays the values defining the utilities in the CID model.

In the following paragraphs, some guidelines to assign numerical values to the utilities, for units $U_{i,s}$ that do not belong to level 1, are presented. For utilities in level 1, we can proceed as in the SID model. In order to do so, let us determine the most and the least desirable situations. In what follows, $U_{j,s-1}$ will denote the unit that contains $U_{i,s}$ (i.e., $j = z(i,s)$). It seems obvious that the most preferable situation is to show a unit $U_{i,s}$ when it is relevant but the unit that contains it, $U_{j,s-1}$, is not. In this case, relevant information in a context which is not, must be retrieved without showing this context. The decision not to show $U_{i,s}$ in these conditions, on the other hand, is the least preferable since it hides relevant information from the user. Therefore, we shall fix

$$v(r^+_{i,s}\,|\,u^+_{i,s}, u^-_{j,s-1}) = 1 \quad \text{and} \quad v(r^-_{i,s}\,|\,u^+_{i,s}, u^-_{j,s-1}) = 0.$$

Let us now consider the case where $U_{i,s}$ is not relevant; we must distinguish between two different situations: $U_{j,s-1}$ is relevant, or $U_{j,s-1}$ is not relevant. In both cases, it is clear that it is more useful not to retrieve $U_{i,s}$ than it is to retrieve it, but is it more preferable not to show $U_{i,s}$ when the context in which it is contained is also irrelevant or when it is relevant? We believe that not retrieving $U_{i,s}$ in the first case is more justified than in the second, as this unit is not even placed near the relevant information. Therefore

$$v(r^-_{i,s}\,|\,u^-_{i,s}, u^-_{j,s-1}) \geqslant v(r^-_{i,s}\,|\,u^-_{i,s}, u^+_{j,s-1}) \geqslant v(r^+_{i,s}\,|\,u^-_{i,s}, u^+_{j,s-1}) \geqslant v(r^+_{i,s}\,|\,u^-_{i,s}, u^-_{j,s-1}).$$

Table 1
Utility table for $V_{i,j}$ in the SID model

$V_{i,j}$	$R_{i,j}$	
$U_{i,j}$	$r^+_{i,j}$	$r^-_{i,j}$
$u^+_{i,j}$	$v(r^+_{i,j}\,\|\,u^+_{i,j})$	$v(r^-_{i,j}\,\|\,u^+_{i,j})$
$u^-_{i,j}$	$v(r^+_{i,j}\,\|\,u^-_{i,j})$	$v(r^-_{i,j}\,\|\,u^-_{i,j})$

Table 2
Utility tables for $V_{i,s}(s \neq 1)$ and $V_{i,1}$ in the CID model

$V_{i,s}$, $s \neq 1$		$R_{i,s}$	
$U_{i,s}$	$U_{j,s-1}$	$r^+_{i,s}$	$r^-_{i,s}$
$u^+_{i,s}$	$u^+_{j,s-1}$	$v(r^+_{i,s} \mid u^+_{i,s}, u^+_{j,s-1})$	$v(r^-_{i,s} \mid u^+_{i,s}, u^+_{j,s-1})$
$u^+_{i,s}$	$u^-_{j,s-1}$	$v(r^+_{i,s} \mid u^+_{i,s}, u^-_{j,s-1})$	$v(r^-_{i,s} \mid u^+_{i,s}, u^-_{j,s-1})$
$u^-_{i,s}$	$u^+_{j,s-1}$	$v(r^+_{i,s} \mid u^-_{i,s}, u^+_{j,s-1})$	$v(r^-_{i,s} \mid u^-_{i,s}, u^+_{j,s-1})$
$u^-_{i,s}$	$u^-_{j,s-1}$	$v(r^+_{i,s} \mid u^-_{i,s}, u^-_{j,s-1})$	$v(r^-_{i,s} \mid u^-_{i,s}, u^-_{j,s-1})$
$V_{i,1}$	$R_{i,1}$		
$U_{i,1}$	$r^+_{i,1}$	$r^-_{i,1}$	
$u^+_{i,1}$	$v(r^+_{i,1} \mid u^+_{i,1})$	$v(r^-_{i,1} \mid u^+_{i,1})$	
$u^-_{i,1}$	$v(r^+_{i,1} \mid u^-_{i,1})$	$v(r^-_{i,1} \mid u^-_{i,1})$	

$U_{j,s-1}$ denotes the unit that contains $U_{i,s}(j = z(i,s))$.

Let us now assume that $U_{j,s-1}$ is not relevant. Is it more desirable to show the relevant information (retrieving $U_{i,s}$ when it is relevant), or not to show the irrelevant information (not retrieving $U_{i,s}$ when it is not relevant)? We believe that it is more important not to lose information which is useful for the user than it is to show useless information, [6] so we shall suppose that

$$v(r^+_{i,s} \mid u^+_{i,s}, u^-_{j,s-1}) \geqslant v(r^-_{i,s} \mid u^-_{i,s}, u^-_{j,s-1}) \quad \text{and} \quad v(r^+_{i,s} \mid u^-_{i,s}, u^-_{j,s-1}) \geqslant v(r^-_{i,s} \mid u^+_{i,s}, u^-_{j,s-1}).$$

Let us imagine that $U_{j,s-1}$ is relevant (i.e. the context is relevant). In this case, it seems clear that showing relevant information is more useful than showing irrelevant information, and hiding relevant information is less useful than hiding irrelevant information. This means that

$$v(r^+_{i,s} \mid u^+_{i,s}, u^+_{j,s-1}) \geqslant v(r^+_{i,s} \mid u^-_{i,s}, u^+_{j,s-1}) \quad \text{and} \quad v(r^-_{i,s} \mid u^-_{i,s}, u^+_{j,s-1}) \geqslant v(r^-_{i,s} \mid u^+_{i,s}, u^+_{j,s-1}).$$

Finally, we shall deal with the case where both the context and the considered unit are relevant. Is it more preferable to retrieve $U_{i,s}$ than not to do so? If we made the first decision, the same reasoning would force us to also retrieve $U_{j,s-1}$, (regardless of whether the unit containing $U_{j,s-1}$ is relevant or not), thereby showing redundant information. It therefore seems reasonable under these circumstances not to show $U_{i,s}$, and to show $U_{j,s-1}$ instead. Therefore, it shall be assumed that

$$v(r^-_{i,s} \mid u^+_{i,s}, u^+_{j,s-1}) \geqslant v(r^+_{i,s} \mid u^+_{i,s}, u^+_{j,s-1}).$$

Combining all these previous inequalities, a complete ordering of all the utilities is obtained:

$$\begin{aligned} 1 = v(r^+_{i,s} \mid u^+_{i,s}, u^-_{j,s-1}) &\geqslant v(r^-_{i,s} \mid u^-_{i,s}, u^-_{j,s-1}) \geqslant v(r^-_{i,s} \mid u^-_{i,s}, u^+_{j,s-1}) \geqslant v(r^-_{i,s} \mid u^+_{i,s}, u^+_{j,s-1}) \geqslant v(r^+_{i,s} \mid u^+_{i,s}, u^+_{j,s-1}) \\ &\geqslant v(r^+_{i,s} \mid u^-_{i,s}, u^+_{j,s-1}) \geqslant v(r^+_{i,s} \mid u^-_{i,s}, u^-_{j,s-1}) \geqslant v(r^-_{i,s} \mid u^+_{i,s}, u^-_{j,s-1}) = 0. \end{aligned} \tag{6}$$

Setting up an ordering of all the possible alternatives is useful but insufficient, since it is essential to assign numerical values to each of the parameters. The results will be different depending on the specific values used and the users' preferences.

3.3.1. Assessing the utility values

One easy way to simplify the task of assessing the utility values is to assume that these values do not depend on the specific structural unit being considered, i.e.

[6] Recall is being rewarded against precision.

$$v(r_{i,j}\,|\,u_{i,j}) = v(r_{i',j'}\,|\,u_{i',j'}),\ \forall i,i',\ \forall j,j' = 1,\ldots,\ell \tag{7}$$

for the SID model, and

$$\begin{aligned} v(r_{i,1}\,|\,u_{i,1}) &= v(r_{i',1}\,|\,u_{i',1})\ \forall i,i', \\ v(r_{i,s}\,|\,u_{i,s},u_{z(i,s),s-1}) &= v(r_{i',s'}\,|\,u_{i',s'},u_{z(i',s'),s'-1})\ \forall i,i',\ \forall s,s' = 2,\ldots,\ell \end{aligned} \tag{8}$$

for the CID model. The only objective of this assumption is to reduce the number of parameters that must be assessed. In this way, only four parameters are required in the SID model and eight for the CID. In the experiments in Section 5, we shall try to determine values of these parameters offering a good retrieval performance. Another more complex option, for example, would be to use different utility values for different levels (reflecting user preferences about the desirability of more or less complex structural units) and the same values within each level.

A different proposal (one which we shall also explore experimentally in Section 5) is to consider a different information source to define the utility values: the query itself. We could say that a given structural unit $U_{i,s}$ will be more useful (with respect to a query Q) as more terms indexing $U_{i,s}$ also belong to Q. More formally, let us consider the sum of the inverted document frequencies of those terms indexing a unit $U_{i,s}$ that also belong to the query Q, normalized by the sum of the idfs of the terms contained in the query:

$$nidf_Q(U_{i,s}) = \frac{\sum_{T_k \in A(U_{i,s}) \cap Q} idf_k}{\sum_{T_k \in Q} idf_k}. \tag{9}$$

These values $nidf_Q(U_{i,s})$ will be used as a correction factor of the previously defined utility values, for each utility node $V_{i,s}$

$$\begin{aligned} v^*(r_{i,s}\,|\,u_{i,s}) &= v(r_{i,s}\,|\,u_{i,s}) \cdot nidf_Q(U_{i,s}), \\ v^*(r_{i,s}\,|\,u_{i,s},u_{j,s-1}) &= v(r_{i,s}\,|\,u_{i,s},u_{j,s-1}) \cdot nidf_Q(U_{i,s}). \end{aligned} \tag{10}$$

4. Inference: evaluation of the influence diagram

In order to solve an influence diagram, the expected utility of each possible decision (for those situations of interest) is determined, in order to make decisions which maximize the expected utility. In our context, there is only one of these situations, corresponding to the query Q formulated by a user to the IRS. In this case, each term T_i occurring in the query is instantiated to t_i^+ (relevant), [7] and we then wish to compute the expected utility of each decision given the query. As we have assumed a global additive utility model, and the different decision variables $V_{i,s}$ are not directly linked to each other, we can process each independently.

4.1. Inference with the SID model

The expected utility for each structural unit $U_{i,j}$ in the SID model can be computed by means of

$$\begin{aligned} EU(r_{i,j}^+\,|\,Q) &= \sum_{u_{i,j} \in \{u_{i,j}^+, u_{i,j}^-\}} v(r_{i,j}^+\,|\,u_{i,j}) p(u_{i,j}\,|\,Q), \\ EU(r_{i,j}^-\,|\,Q) &= \sum_{u_{i,j} \in \{u_{i,j}^+, u_{i,j}^-\}} v(r_{i,j}^-\,|\,u_{i,j}) p(u_{i,j}\,|\,Q). \end{aligned} \tag{11}$$

[7] A similar approach has been used with the BNR-SD model (Crestani et al., 2003).

In order to compute the expected utility, we therefore need to compute the posterior probabilities $p(u_{i,j}|Q)$. Since in the designed influence diagram, a chance node only has as its parent set some chance nodes (terms or structural units), the computation of these probabilities can be performed using ordinary inference in Bayesian networks, known as *evidence propagation*.

There are different algorithms (Jensen, 2001; Pearl, 1988) that perform the propagation process efficiently, although generally speaking, the problem of evidence propagation is NP-Hard (Cooper, 1990). However, in view of the large number of variables involved and the complex topology of the influence diagram, general purpose inference algorithms cannot be applied. In our case, considering that (1) all the conditional probabilities have been assessed using a specific canonical model, (2) the multi-layered topology of the network (arcs only go from nodes in one level to nodes in the previous one), and (3) that only term nodes are instantiated (so that only a top-down inference is required), we can use a specific inference process designed for a non-structured document Bayesian network retrieval model (de Campos et al., 2003a), which has also been applied to structured document collections (see Crestani et al., 2003). With this methodology, the inference process can be carried out very efficiently in the following way:

- For the structural units in level $\mathscr{L}_\ell$

$$P(u^+_{i,\ell}|Q) = \sum_{T_k \in Pa(U_{i,\ell}) \cap Q} w(T_k, U_{i,\ell}) + p_0 \sum_{T_k \in Pa(U_{i,\ell}) \setminus Q} w(T_k, U_{i,\ell}) \tag{12}$$

  or equivalently (taking into account the fact that the weights $w(T_k, U_{i,\ell})$ are normalized)

$$p(u^+_{i,\ell}|Q) = p_0 + (1-p_0) \sum_{T_k \in Pa(U_{i,\ell}) \cap Q} w(T_k, U_{i,\ell}). \tag{13}$$

- For the structural units in level $\mathscr{L}_j$, $j \neq \ell$

$$P(u^+_{i,j}|Q) = \sum_{U_{h,j+1} \in Pa(U_{i,j})} w(U_{h,j+1}, U_{i,j}) \cdot p(u^+_{h,j+1}|Q). \tag{14}$$

  If we define the weight $w(T_k, U_{i,j})$ of a term T_k in a structural unit of level $j \neq \ell$ analogously to the weight of T_k in a unit of level ℓ (Eq. (3)), i.e.

$$w(T_k, U_{i,j}) = \frac{\rho(T_k, A(U_{i,j}))}{\sum_{T_h \in A(U_{i,j})} \rho(T_h, A(U_{i,j}))}, \tag{15}$$

  then an expression equivalent to Eq. (14) is

$$p(u^+_{i,j}|Q) = p_0 + (1-p_0) \sum_{T_k \in A(U_{i,j}) \cap Q} w(T_k, U_{i,j}). \tag{16}$$

We can therefore compute the required probabilities on a level-by-level basis, starting from level ℓ and going down to level 1.

4.2. Inference with the CID model

In this case, we need to differentiate between the nodes in level $s \neq 1$ and the nodes in level 1. For each level $s \neq 1$, as well as for each structural unit belonging to that level, $U_{i,s} \in \mathscr{L}_s$, let $U_{j,s-1} \in \mathscr{L}_{s-1}$ be the unit from level $s-1$ which contains it (i.e. $U_{i,s} \in Pa(U_{j,s-1})$ and $j = z(i,s)$). The expected utility of retrieving the structural unit $U_{i,s}$, given a query Q, can be computed by means of the following expression:

$$EU(r_{i,s}^{+}\,|\,Q) = \sum_{\substack{u_{i,s}\in\{u_{i,s}^{+},u_{i,s}^{-}\}\\ u_{j,s-1}\in\{u_{j,s-1}^{+},u_{j,s-1}^{-}\}}} v(r_{i,s}^{+}\,|\,u_{i,s},u_{j,s-1})p(u_{i,s},u_{j,s-1}\,|\,Q). \tag{17}$$

Analogously, the expected utility of not retrieving the same unit can be calculated as follows:

$$EU(r_{i,s}^{-}\,|\,Q) = \sum_{\substack{u_{i,s}\in\{u_{i,s}^{+},u_{i,s}^{-}\}\\ u_{j,s-1}\in\{u_{j,s-1}^{+},u_{j,s-1}^{-}\}}} v(r_{i,s}^{-}\,|\,u_{i,s},u_{j,s-1})p(u_{i,s},u_{j,s-1}\,|\,Q). \tag{18}$$

Regarding the expected utilities of units $U_{i,1}$ from level 1, these may be computed using

$$\begin{aligned} EU(r_{i,1}^{+}\,|\,Q) &= \sum_{u_{i,1}\in\{u_{i,1}^{+},u_{i,1}^{-}\}} v(r_{i,1}^{+}\,|\,u_{i,1})p(u_{i,1}\,|\,Q),\\ EU(r_{i,1}^{-}\,|\,Q) &= \sum_{u_{i,1}\in\{u_{i,1}^{+},u_{i,1}^{-}\}} v(r_{i,1}^{-}\,|\,u_{i,1})p(u_{i,1}\,|\,Q). \end{aligned} \tag{19}$$

To put this model into practice, it is therefore necessary to assess the bi-dimensional posterior probabilities, corresponding to each structural unit $U_{i,s}$ from level s and the unit $U_{j,s-1}$ where it is contained, for each level $s \neq 1$, $p(u_{i,s},u_{j,s-1}\,|\,Q)$. To obtain these values may be a time consuming process, because of the great amount of calculations required on retrieval time. This is the reason because we propose to use a first approximation assuming that both units are independent given the query, i.e.,

$$p(u_{i,s},u_{j,s-1}\,|\,Q) = p(u_{i,s}\,|\,Q)p(u_{j,s-1}\,|\,Q). \tag{20}$$

Therefore, we only have to compute the relevance values for each structural unit given a query Q, using Eqs. (12) and (14).

4.3. Decision-making

In the context of a typical decision-making problem, once the expected utilities have been computed, the decision with the greatest utility is chosen. In our case, however, this would mean retrieving the structural unit $U_{i,s}$ (i.e. to show it to the user) if $EU(r_{i,s}^{+}\,|\,Q) \geqslant EU(r_{i,s}^{-}\,|\,Q)$, and not retrieving it otherwise. Yet our purpose is not only to make decisions about what to retrieve but also to rank these units, showing them in decreasing order of utility. Consequently, one very important question is what technique to use in order to sort the units by taking into account their expected utilities.

We shall now discuss three different alternatives in order to rank each unit. These measures will be generically called *Re-ranking Utility Measures* (RUM)

RUM-u: The most natural way is to rank the units according to the expected utility of retrieving each unit, $RUM_u(U_{i,s}) = EU(r_{i,s}^{+}\,|\,Q)$, although this method does not consider the utilities of not retrieving the units, $EU(r_{i,s}^{-}\,|\,Q)$.

RUM-q: Another reasonable choice is to use $RUM_u(U_{i,s}) = EU(r_{i,s}^{+}\,|\,Q)/EU(r_{i,s}^{-}\,|\,Q)$, i.e. the quotient between utilities.

RUM-d: The difference between both expected utilities, $RUM_u(U_{i,s}) = EU(r_{i,s}^{+}\,|\,Q) - EU(r_{i,s}^{-}\,|\,Q)$, can also be applied.

It is interesting to note that the CID model can mimic the behavior of the SID if the utilities $v(r_{i,s}\,|\,u_{i,s},u_{j,s-1})$ are defined appropriately. More precisely, if these utilities are defined in such a way that $v(r_{i,s}^{+}\,|\,u_{i,s}^{+},u_{j,s-1}^{+}) = v(r_{i,s}^{+}\,|\,u_{i,s}^{+},u_{j,s-1}^{-})$, $v(r_{i,s}^{+}\,|\,u_{i,s}^{-},u_{j,s-1}^{+}) = v(r_{i,s}^{+}\,|\,u_{i,s}^{-},u_{j,s-1}^{-})$, $v(r_{i,s}^{-}\,|\,u_{i,s}^{+},u_{j,s-1}^{+}) = v(r_{i,s}^{-}\,|\,u_{i,s}^{+},u_{j,s-1}^{-})$ and

$v(r^-_{i,s}\,|\,u^-_{i,s}, u^+_{j,s-1}) = v(r^-_{i,s}\,|\,u^-_{i,s}, u^-_{j,s-1})$, then the expected utilities $EU(r^+_{i,s}\,|\,Q)$ and $EU(r^-_{i,s}\,|\,Q)$ computed by the SID and the CID are the same, and therefore the corresponding RUM measures are also equal.

Let us now consider the situation where the utility values are the same for all the structural units. In this case, it can be easily seen that if the utility values verify the ordering established in Eq. (5), then the ranking of the structural units obtained by the SID model using any of the three RUM measures is exactly the same as the one produced by the BNR-SD model (which ranks the units in decreasing order of their posterior probabilities of relevance), i.e. $RUM(U_{i,s}) \geqslant RUM(U_{i',s'}) \Longleftrightarrow p(u^+_{i,s}\,|\,Q) \geqslant p(u^+_{i',s'}\,|\,Q)$. We can therefore say that the SID subsumes the BNR-SD model.

This observation is important because ranking structural units using only the probabilities of relevance has one shortcoming: if we analyze the expressions that compute these probabilities (Eq. (14)), we can observe that the probability of relevance of a unit $U_{i,s}$ at level $s \neq \ell$ is always less than or equal to the relevance probability of one of the units in level $s+1$ being contained in $U_{i,s}$ (since the weights $w(U_{h,s+1}, U_{i,s})$ are normalized). There is therefore a tendency to consider that the structural units which are more specific and reduced (such as paragraphs) are more relevant than the larger units (such as chapters or complete documents). This is due to the fact that the relevance probability of a structural unit $U_{i,s}$ in our model is essentially defined in terms of the number of terms that $U_{i,s}$ and the query Q have in common, as Eqs. (13) and (16) clearly reveal. As the weights used in these equations are normalized with respect to the number of terms in $U_{i,s}$, the relevance probabilities do not take into account the number of terms that belong to Q but do not belong to $U_{i,s}$. In this way, a unit with a small number of terms, [8] most of which appear in Q, may become more relevant than a larger unit which shares many more terms with Q.

This leads us to introduce the modified utility values v^* defined in Eq. (10). These values use information about the terms that belong to the query and do not belong to the structural unit $U_{i,s}$, by means of the factor $nidf_Q(U_{i,s})$. It should be noted that the larger the number of terms in Q that do not belong to $U_{i,s}$, the small the value $nidf_Q(U_{i,s})$. It should also be observed that if $U_{j,s-1}$ is the unit containing $U_{i,s}$, then $nidf_Q(U_{i,s}) \leqslant nidf_Q(U_{j,s-1})$, and therefore the bias of the BNR-SD model in favoring small structural units may be compensated.

The computation of the new expected utilities, EU^*, obtained by using the utilities v^*, is very simple:

$$EU^*(r^+_{i,s}\,|\,Q) = EU(r^+_{i,s}\,|\,Q) \cdot nidf_Q(U_{i,s}), \tag{21}$$

$$EU^*(r^-_{i,s}\,|\,Q) = EU(r^-_{i,s}\,|\,Q) \cdot nidf_Q(U_{i,s}). \tag{22}$$

The corresponding RUM measures can also easily be computed: [9]

$$\begin{aligned} RUM^*_u(U_{i,s}) &= RUM_u(U_{i,s}) \cdot nidf_Q(U_{i,s}),\\ RUM^*_d(U_{i,s}) &= RUM_d(U_{i,s}) \cdot nidf_Q(U_{i,s}),\\ RUM^*_q(U_{i,s}) &= RUM_q(U_{i,s}). \end{aligned} \tag{23}$$

5. Experimentation

The model has been tested using a collection of structured documents, marked up in XML, containing the 37 plays of William Shakespeare (Kazai, Lalmas, & Reid, 2001). A play is considered to be structured in acts, scenes and speeches (so that $\ell = 4$), and may also contain epilogues and prologues. Speeches have been

[8] In the experimental collection that we shall use in Section 5 there is a large number of units which are indexed by only one or two terms.

[9] It should be noted that RUM^*_q does not represent anything new, because the factor $nidf_Q$ cancels out.

the only structural units indexed using the Lemur Retrieval Toolkit (available at http://www-2.cs.cmu.edu/~lemur/). The total number of unique terms contained in these units is 14 019, and the total number of structural units taken into account is 32 022. With respect to the queries, the collection is distributed with 43 queries, with their corresponding relevance judgements. From these 43 queries, the 35 which are content-only queries were selected for our experiments.

Our aim in this section is to determine the contribution of the used utility values to the ranking of structural units. In each experiment, all the structural units have been simultaneously considered. Thus, after a first stage where the posterior probabilities of each unit, $p(u^+_{i,s} \mid Q)$, are computed, in a second phase the expected utilities are calculated and ranked according to each RUM. In order to measure the effectiveness of the retrieval, our methodology will be based on *recall* and *precision* estimates (van Rijsbergen, 1979; Salton & McGill, 1983); more precisely, we use the mean precision for the 11 standard recall points (Salton & McGill, 1983), AVP-11.

5.1. Experimentation with the SID model

In this case we need to discuss two alternatives: the first where we use the same utility values for all the structural unit (Eq. (7)), and the second where we use the utilities v^* (Eq. (10)). In the first case, and assuming the ordering of the utility values displayed in Eq. (5), the SID model obtains the same results as the BNR-SD model (Crestani et al., 2003) with an AVP-11 equal to 0.0653. This value will be used as a baseline for all the experimentation, so that in the subsequent tables of results we shall also display the percentage of change (%C) with respect to this baseline.

We shall now discuss the results obtained when using the utility values $v^* = v \cdot nidf_Q$. Since given a query Q, the $nidf_Q$ value is automatically computed, we shall study the effects on the performance when considering different v values. Following the discussion in Section 3.3, we set the utility of retrieving a relevant document to the maximum, i.e. $v(r^+ \mid u^+) = 1$, and the utility of not retrieving a relevant document to the minimum, $v(r^- \mid u^+) = 0$. In this way, we only need to experiment with the utility values of $v(r^+ \mid u^-)$ and $v(r^- \mid u^-)$. The results are displayed in Table 3.

The panel A presents the results obtained when we use RUM_u. In this case, we only need to consider the value $v(r^+ \mid u^-)$, i.e. the utility of retrieving a non-relevant document. We consider values in the interval $[0, 1]$, with a 0.1 step. The best results are obtained when using $v(r^+ \mid u^-) = 0.0$, although the performance is quite similar in all cases (it only starts to degrade for values greater than 0.4). These results are coherent

Table 3
Results of the experiments carried out with the SID model

RUM_u (Panel A)			RUM_d: Max (Panel B)			RUM_d: fixed v^+_- (Panel C)		
v^+_-	AVP-11	%C	v^+_-	v^-_-	AVP-11	v^-_-	$v^+_- = 0.0$	$v^+_- = 0.5$
0.0	0.1808	176.8	0.0	0.0	0.1808	0.0	0.1808	0.1795
0.1	0.1798	175.3	0.1	0.1	0.1808	0.1	0.1790	0.1802
0.2	0.1794	174.7	0.2	0.2	0.1808	0.2	0.1770	0.1805
0.3	0.1805	176.4	0.3	0.3	0.1808	0.3	0.1729	0.1794
0.4	0.1802	175.9	0.4	0.4	0.1808	0.4	0.1700	0.1798
0.5	0.1795	174.9	0.5	0.5	0.1808	0.5	0.1646	0.1808
0.6	0.1790	174.1	0.6	0.6	0.1808	0.6	0.1534	0.1790
0.7	0.1786	173.5	0.7	0.7	0.1808	0.7	0.1332	0.1770
0.8	0.1782	172.9	0.8	0.8	0.1808	0.8	0.1143	0.1729
0.9	0.1781	172.7	0.9	0.9	0.1808	0.9	0.0984	0.1700
1.0	0.1677	156.8	1.0	1.0	0.1808	1.0	0.0811	0.1646

$v^+_- = v(r^+ \mid u^-)$ and $v^-_- = v(r^- \mid u^-)$.

with the interpretation of the parameter $v(r^+ \mid u^-)$. The high increment in retrieval performance obtained in comparison with the model that uses fixed utilities (more than 176%) is particularly remarkable.

The panel B, labeled with 'RUM_d: Max' presents the maximum values obtained when using RUM_d. As the number of alternatives is small, we have considered all the possible combinations of the values for $v(r^+ \mid u^-)$ and $v(r^- \mid u^-)$, taking values from 0 to 1 with an increment of 0.1. In this case, the system performance is better when these two values are equal (a situation where RUM_d reduces to RUM_u with $v(r^+ \mid u^-) = 0$).

The panel C is used to illustrate the performance of RUM_d when we fix $v(r^+ \mid u^-)$. In this case, the behavior is quite similar for all the experiments (we only show the results for the values $v(r^+ \mid u^-) = 0.0$ and $v(r^+ \mid u^-) = 0.5$). It is interesting to note the decrease in performance that occurs when $v(r^- \mid u^-)$ separates from $v(r^+ \mid u^-)$.

Finally, with respect to the system's time efficiency, the mean time used by the SID to process the 35 queries was 7.3 s.

5.2. Experimentation with the CID model

Our objective here is also to determine, if possible, patterns that offer a good performance, using several combinations of *RUM*s and utilities.

We will start with the case where the utilities are the same for all the structural units. In this case, we need to assess eight utility values. Once again following the ideas set out in Section 3.3, two of these will be fixed: $v(r^+ \mid u^+, w^-) = 1$ and $v(r^- \mid u^+, w^-) = 0$, where u denotes any structural unit and w the single structural unit containing u. In any case, even considering six utility values, the number of possible combinations is huge. Consequently, we have used a greedy approach to search for the best combination of values. Roughly speaking, this approach consists of a series of steps where in each one we fix five parameters and look for the best utility value [10] for the sixth.

Table 4 shows the results of these experiments. The first three rows show the maximum AVP-11 values obtained when the utilities are forced to satisfy the ordering suggested by Eq. (7), combined with the three proposed *RUM*s. For RUM_u, we only present the four values $v(r^+ \mid u, w)$ (those necessary to compute $EU(r^+ \mid Q)$). In general, we can observe an improvement in the system's performance with respect to the corresponding experiment with the SID model (AVP-11 = 0.0653), with percentages of change greater than 100%.

The last three rows in Table 4 show the results obtained when we do not take into account the restriction in the ordering imposed by Eq. (7). The best performance for RUM_u is obtained by the same previous utility values. The value $v(r^+ \mid u^+, w^+) = 0.5$ for RUM_u seems to point towards a slightly conservative strategy (probably recall-enhancing), where it is useful to retrieve a relevant unit even if its context is also relevant. The other measures, RUM_q and RUM_d, scarcely improve the previous results (an increase in the average precision of around 1%). In these cases, only one of the order restrictions in Eq. (7) is violated: $v(r^- \mid u^+, w^+) \not\leqslant v(r^- \mid u^-, w^+)$.

Finally, we have also carried out experiments with the CID model using the utility values $v^* = v \cdot idf_Q$. The first two rows in Table 5 show the best combination of utility values v found verifying the ordering in Eq. (7), using RUM_u and RUM_d (it should be remembered that in this case it is pointless to use RUM_q). The last two rows show the results when this restriction is removed. Once again, we observe a considerable increase in retrieval performance with respect to the same model using fixed utilities. We believe that some of the 'anomalous' utility values found in the experiments (in the sense of not completely verifying the ordering established in Eq. (7)) may be due to the relevance judgments provided with the collection: these are not oriented to determine the best entry points to the documents. Therefore, if a relevant unit comprises

[10] The values are in the range $[0, 1]$, with an increment of 0.005.

Table 4
Results of the experiments with the CID model using the same utilities for all the structural units

RUM	v^-_{+-}	v^+_{--}	v^+_{-+}	v^+_{++}	v^-_{++}	v^-_{-+}	v^-_{--}	v^+_{+-}	AVP-11	%*C*
RUM_u	–	0.0	0.1	0.5	–	–	–	1.0	0.1387	112.4
RUM_q	0.0	0.0	0.48	0.95	0.95	0.95	0.95	1.0	0.1402	114.7
RUM_d	0.0	0.0	0.0	0.0	0.65	0.70	0.72	1.0	0.1372	110.1
RUM_u	–	0.0	0.1	0.5	–	–	–	1.0	0.1387	112.4
RUM_q	0.0	0.0	0.20	0.81	0.90	0.57	1.0	1.0	0.1412	116.2
RUM_d	0.0	0.0	0.0	0.0	0.65	0.35	0.70	1.0	0.1386	112.3

$v^-_{+-} = v(r^-|u^+, w^-)$, $v^+_{--} = v(r^+|u^-, w^-)$, $v^+_{-+} = v(r^+|u^-, w^+)$, $v^+_{++} = v(r^+|u^+, w^+)$, $v^-_{++} = v(r^-|u^+, w^+)$, $v^-_{-+} = v(r^-|u^-, w^+)$, $v^-_{--} = v(r^-|u^-, w^-)$ and $v^+_{+-} = v(r^+|u^+, w^-)$.

Table 5
Results of the experiments with the CID model using the utility values v^*

RUM	v^-_{+-}	v^+_{--}	v^+_{-+}	v^+_{++}	v^-_{++}	v^-_{-+}	v^-_{--}	v^+_{+-}	AVP-11	%*C*
RUM_u	–	0.9	0.9	0.9	–	–	–	1.0	0.1878	187.6
RUM_d	0.0	0.0	0.0	0.0	0.0	0.01	0.025	1.0	0.1650	152.7
RUM_u	–	0.0	0.9	0.1	–	–	–	1.0	0.1984	203.8
RUM_d	0.0	0.0	0.15	0.30	0.0	0.0	0.70	1.0	0.1899	190.8

units which are also relevant, all of these are considered equally by the evaluation process (although in this case, as we discussed, it would be preferable to retrieve the larger unit). With respect to the time efficiency of the CID, the mean time required to process all the queries was 15.2 s.

Two clear conclusions emerge from the analysis of the experimental results. The first is that the use of the utility values v^*, which combine a fixed component v, equal for all the structural units, and a variable component $nidf_Q$, specific for each unit and query, is much better than using only the uniform and static utility values v, in terms of retrieval effectiveness. The second conclusion is that the CID model obtains better results than the SID, which supports the idea that context information is important and can be used to improve the performance of a structured document retrieval system. Nevertheless, the differences between the CID and the SID are not very pronounced (the increment in the average precision of the CID with respect to the SID is less than 10%). We believe that this may be due to the extremely simplistic (and unrealistic) assumption of independence between structural units made (Eq. (20)), in order to efficiently compute the bi-dimensional probabilities of relevance that the CID requires. We speculate that a more precise estimation of these probabilities would produce greater differences between these two models.

6. Conclusions and further research

In this paper, we have introduced a new retrieval model for structured documents based on influence diagrams, a generalization of Bayesian networks. Not only does this model present a solid foundation but it is also computationally tractable. Although a lot of work has been done on IR based on Bayesian networks, some of this dealing with structured documents, as far as we know the approach to structured document retrieval as a decision-making problem and, in particular, the application of influence diagrams, is totally new. The two models proposed, the SID and the CID, show promising experimental results. The SID is more efficient, although the CID (in the simplified form discussed in this paper) is somewhat more effective.

There are many ways in which this work could be improved. For instance, instead of simplifying the CID model by assuming that each unit is (conditionally) independent on the one where it is contained, a method

could be developed to compute the exact posterior probability of a unit and its container, given the query. This task could be very time and resource consuming, so perhaps the search for another approximate computation might also be interesting, thereby reducing the required time without losing too much accuracy in the results. After this stage, the model will be tested with other larger, more complex collections, such as INEX (Initiative for the Evaluation of XML Retrieval). [11]

We must also improve the selection of utility functions. Using different utility configurations for each type of structural unit would produce a more flexible model (this could be adapted to user preferences and collection types), and would not cause a problem from either the theoretical or the implementational point of view.

Among the possible model modifications or extensions that we plan to put in practice, the following should be mentioned:

- Until now, the structure of documents has been very rigid, with each unit from level $s \neq 1$ being contained in another in level $s-1$. However, we could consider less homogeneous documents, where it is possible to establish relationships between units from different (non-consecutive) levels. For instance, a book divided into chapters, where some are divided into sections which, in turn, are divided into paragraphs, and other chapters containing no structure but paragraphs. It is not difficult to deal with this kind of structure since all the formulas developed in this paper could be applied directly.
- Another limitation of the model is that only the units from level ℓ have indexing terms. Nevertheless, it would be extremely easy to consider other units from different levels that could have indexing terms assigned, such as for example, the title of a chapter. The modification of the model to deal with this situation is also very easy if we allow relationships between non-consecutive structural units.
- Certain structural units (e.g. title or abstract) could be more representative of the content of a document than another. Therefore, it might be interesting to give different weights to the structural units that comprise another larger unit, which would reflect the relative importance of each type of unit. In this way, the same term appearing in two different units would contribute to the relevance of these units differently (e.g. a term in the title of a paper may be more important than the same term appearing in a section).
- Regarding the kind of query that our model can deal with, in this paper content-only queries have been considered. We also plan to extend the model to accept structure-only and content-and-structure queries.
- Finally, another feature which we would like to introduce into our model is the direct relationship between terms, as in de Campos et al. (2003a) and de Campos, Fernández-Luna, and Huete (2003b) for non-structured retrieval. Direct relationships between documents (Acid, de Campos, Fernández-Luna, & Huete, 2003) are also interesting, and are particularly important when applying our model to the Web.

References

Acid, S., de Campos, L. M., Fernández-Luna, J. M., & Huete, J. F. (2003). An information retrieval model based on simple Bayesian networks. *International Journal of Intelligent Systems, 18*, 251–265.

Chiaramella, Y. (2001). Information retrieval and structured documents. In *Lecture notes in computer science* (Vol. 1980, pp. 291–314).

Cooper, G. (1990). The computational complexity of probabilistic inference using Bayesian belief networks. *Artificial Intelligence, 42*, 393–405.

Crestani, F., de Campos, L. M., Fernández-Luna, J. M., & Huete, J. F. (2003). A multi-layered Bayesian network model for structured document retrieval. In *Lecture notes in artificial intelligence* (Vol. 2711, pp. 74–86).

de Campos, L. M., Fernández-Luna, J. M., & Huete, J. F. (2003a). The BNR model: foundations and performance of a Bayesian network-based retrieval model. *International Journal of Approximate Reasoning, 34*, 265–285.

[11] http://www.is.informatik.uni-duisburg.de/projects/inex/index.html.en

de Campos, L. M., Fernández-Luna, J. M., & Huete, J. F. (2003b). Two term-layers: an alternative topology for representing term relationships in the Bayesian network retrieval model. In J. Benítez, O. Cordón, F. Hoffmann, & R. Roy (Eds.), *Advances in soft computing—engineering, design and manufacturing* (pp. 213–224). Berlin: Springer Verlag.

Graves, A., & Lalmas, M. (2002). Video retrieval using an MPEG-7 based inference network. In *Proceedings of the 25th ACM–SIGIR conference* (pp. 339–346). New York: ACM.

Jensen, F. (2001). *Bayesian networks and decision graphs*. Berlin: Springer Verlag.

Kazai, G., Lalmas, M., & Reid, J. (2001). *The Shakespeare test collection*.

Myaeng, S., Jang, D., Kim, M., & Zhoo, Z. (1998). A flexible model for retrieval of SGML documents. In *Proceedings of the 21st ACM–SIGIR conference* (pp. 138–145). New York: ACM.

Pearl, J. (1988). *Probabilistic reasoning in intelligent systems: Networks of plausible inference*. San Mateo, CA: Morgan and Kaufmann.

Piwowarski, B., Faure, G., & Gallinari, P. (2002). Bayesian networks and INEX. In *Proceedings of the INEX workshop* (pp. 7–12).

Ribeiro-Neto, B. A., & Muntz, R. R. (1996). A belief network model for IR. In H. Frei, D. Harman, P. Schäble, & R. Wilkinson (Eds.), *Proceedings of the 19th ACM–SIGIR conference* (pp. 253–260). New York: ACM.

Salton, G., & McGill, M. J. (1983). *Introduction to modern information retrieval*. New York: McGraw-Hill.

Shachter, R. (1986). Evaluating influence diagrams. *Operations Research, 34*, 871–882.

Shachter, R. (1988). Probabilistic inference and influence diagrams. *Operations Research, 36*(5), 527–550.

Turtle, H. R., & Croft, W. B. (1990). Inference networks for document retrieval. In *Proceedings of 13th international ACM–SIGIR conference* (pp. 1–24). New York: ACM.

van Rijsbergen, C. (1979). *Information retrieval* (2nd ed.). London, UK: Butter Worths.

Zhang, N. (1998). Probabilistic inference in influence diagrams. *Computational Intelligence, 14*, 475–497.