AAAI-88; Proceedings of the Seventh National Conference on Artificial Intelligence, 1988

Evidential Reasoning Using DELIEF

Debra Zarley, Yen-Teh Hsia and Glenn Shafer
School of Business, University of Kansas
Lawrence, Kansas

Abstract

The Dempster-Shafer theory of belief functions [Shafer 1976] is an intuitively appealing formalism for reasoning under uncertainty. Several AI implementations have been undertaken [e.g., Lowrance et al. 1986, Biswas and Anand 1987], but the computational complexity of Dempster's rule has limited the usefulness of such implementations. With the advent of efficient propagation schemes in Markov trees [Shafer et al. 1987], the time is ripe for more powerful systems.

This paper discusses DELIEF (Design of bELIEFs), an interactive system that allows the design of belief function arguments via a simple graphical interface. The user of DELIEF constructs a graph, with nodes representing variables and edges representing relations among variables. This graph serves as a default knowledge schema. The user enters belief functions representing evidence pertinent to the individual variables in a specific situation, and the system combines them to obtain beliefs on all variables. The schema may be revised and re-evaluated until the user is satisfied with the result. The Markov tree used for belief propagation is displayed on demand. The system handles Bayesian causal trees [Pearl 1986] as a special case, and it has a special user interface for this case.

1 Introduction

Reasoning about real-world situations is a process often beset with uncertainty, contradictions and ignorance. Information or evidence may come from many sources: from experience, from sensory data, from context. Such evidence is rarely clear-cut. Often it is incomplete, ambiguous, or misleading.

Uncertain evidence is not easily represented by logical formalisms. Classical probability measures provide an alternative, but they require that evidence be complete in a different sense; in order for probabilities to be well-founded, we need statistical data on many similar cases. Because of these difficulties, existing expert systems often deal with uncertainty using heuristic methods that can lead to unintuitive or hard-to-interpret results.

The theory of belief functions provides another way of dealing with some of these difficulties. It provides a formally consistent method for interpreting and pooling uncertain evidence, and it allows us to get meaningful answers to questions with only partial evidence. Costly evidence is gathered only when necessary. As we shall see, it also allows us to use knowledge schemas that are flexible enough to accomodate unexpected evidence.

In most successful applications, the Bayesian approach to evidential reasoning utilizes probability measures that are extracted from statistical data. The knowledge schemas used by Bayesian analyses in such applications are fixed by the data available. From the user's point of view, these schemas are essentially hardwired. This is an advantage inasmuch as the user needs only input evidence, without worrying about improving the structure of the schema. But it is a disadvantage in terms of flexibility and breadth of relevance.

The belief function approach embodies a different philosophy. Knowledge schemas constructed in the belief function framework can be supported by educated guesses that are simpler in structure than probability measures. The reasoner starts with hunches formalized as a default knowledge schema, gathers and evaluates individual items of evidence for some (but not necessarily all) of the questions considered by this schema, uses the system to combine this evidence and examines resulting joint beliefs on answers to particular questions, and revises the evaluation of the individual items of the evidence or even the structure of the schema as the investigation moves along. This resembles the kind of reasoning process in which an expert (e.g. an auditor) assimilates his or her knowledge into the context (e.g. a firm that is being audited). We think this kind of reasoning process is common in many domains. The DELIEF system aids such processes by facilitating the design of the knowledge schemas and propagating probability judgments within these schemas.

2 Theoretical Background

2.1 Belief Function Models as Knowledge Schemas

A belief function model consists of a set of variables, a set of joint variables, and zero or more belief functions for each of the individual variables and joint variables. Individual variables represent questions for which we would like an answer. A joint variable involves two or more individual variables and is used to specify how these individual variables are related. Associated with each variable is a frame of discernment (henceforth a frame), an exhaustive set of mutually exclusive answers to a question. The simplest kind of variable is a Boolean variable, or a proposition; its frame is the set {true, false}. The frame for a joint variable is the

Cartesian product of the frames of its individual variables. If, for example, A and B are Boolean variables, then the frame of the joint boolean variable $A \times B$ is the Cartesian product {(yes, no) (yes, yes) (no, no) (no, yes)}. Abstractly, the set of variables and joint variables may be thought of as a hypergraph, where each variable is a vertex and each joint variable is a hyperedge containing only vertices that are member variables. This concept leads naturally to a graphical representation of the model [Kong 1986].

A belief function BEL_X bearing on a variable X can be stored as a mapping m_X ("the m-values") from the set of all subsets of X's frame to the real interval [0, 1]. If we let the set of all subsets of X's frame be $\{S_1, S_2, ..., S_k\}$, then the sum of $m_X(S_i)$, where $1 \le i \le k$, is one. The belief function BEL_X itself is computed from m_X. Specifically, for any subset S of X's frame, $BEL_X(S)$ is the sum of all $m_X(S')$, where S' is a subset of S. We call any subset S of X's frame for which $m_X(S)$ is non-zero a focal element of BEL_X. Individual items of evidence are entered into DELIEF in terms of m-values for focal elements.

The simplest kind of belief function is the vacuous belief function, whose only focal element is the entire frame (with m-value 1). Next come the logical belief functions. A logical belief function has only one proper subset as a focal element (with m-value not necessarily equal to 1), and this focal element may be represented by a logical assertion. DELIEF initially associates a vacuous belief function with each node in a knowledge schema. It also provides shortcuts for specifying logical belief functions.

2.2 Propagation

The evaluation of a belief function model consists of the combination of all belief functions and the projection of the resulting joint belief function to the frame of each individual variable or joint variable. At the heart of it is Dempster's rule of combination, the main inference mechanism for belief functions. As Dempster's rule is exponential in computational complexity, care must be taken to trim down the size of the frames involved. So instead of combining all belief functions at once, we try to combine only a few belief functions at a time using local computation. In order for such a procedure to give the same result as combining all the belief functions on an overall frame, the original hypergraph structure of the belief function model must be embedded in a hypergraph that can be arranged as a Markov tree [Shafer et al. 1987]. So to combine belief functions, we first look for a hypergraph embedding, and then we combine belief functions locally and propagate the results through an associated Markov tree structure. See [Kong 1986] and [Mellouli 1987] for more details.

3 The System

Performing evidential reasoning in DELIEF consists of three major steps: creating a model as a graphical network, providing evidence, and propagating the evidence in the model. The user has the option of performing either a Bayesian or a belief function analysis. Bayesian analysis is implemented as a special case of belief function analysis. While interfaces for providing evidence differ, and the inputs greater in the Bayesian analysis, the algorithms for propagating evidence are exactly the same for either case, since probability transition matrices are translated to and stored internally as belief functions. For a formal discussion, see [Shafer et al. 1987].

Throughout this section we will illustrate the system by considering the following hypothetical knowledge schema from [Lauritzen and Spiegelhalter 1988]:

> Shortness-of-breath (dyspnoea) may be due to tuberculosis, lung cancer or bronchitis, or none of them, or more than one of them. A recent visit to Asia increases the chances of tuberculosis, while smoking is known to be a risk factor for both lung cancer and bronchitis. The result of chest x-rays do not discriminate between lung cancer and tuberculosis, nor does the presence or absence of dyspnoea.

3.1 Creating a Model

Variable nodes are created where desired in the graph by clicking the mouse over any blank area of the graph display pane. A window then pops up (Figure 1) over the graph pane and prompts the user to "formalize" the variable by providing a name and a frame for it. Variables are displayed as circular nodes (Figure 3).

Creation of a joint variable is accomplished by first indicating which variable nodes the joint variable conjoins (Figure 2) and then indicating the location of the new joint variable node. Joint variables appear as rectangular nodes with edges to their member variables (Figure 3). The system automatically calculates the joint variable's frame as the Cartesian product of the frames of its member variables. Figure 3 shows the resulting network for our hypothetical knowledge base.

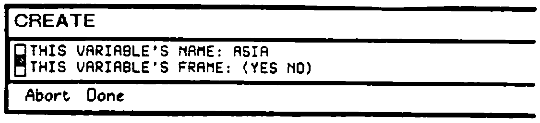

Figure 1: Formalizing a variable node. The variable has been named ASIA, and its frame is the set {YES, NO}.

3.2 Providing Evidence - The Belief-Function Case

Until explicitly defined by the user, all variables and joint variables have a vacuous belief function defined for them by the system.

Evidence for a variable is represented by a belief function on the variable's frame. For example, we wish to indicate that we are 80% sure that the patient in question has visited Asia recently. Figure 4 illustrates how this evidence can be expressed as a belief function on the variable ASIA. First, we select the subset of the variable's frame for which we have evidence, namely "YES", and then indicate that we wish to add this subset to the belief function. We are then prompted to specify the m-value (degree of belief) for this subset, and we enter 0.8 (Figure 4a). Figure 4b shows the resulting belief function for the variable. Twenty percent of our belief is still uncommitted and therefore remains on the entire frame. Joint variables are handled in the same way.

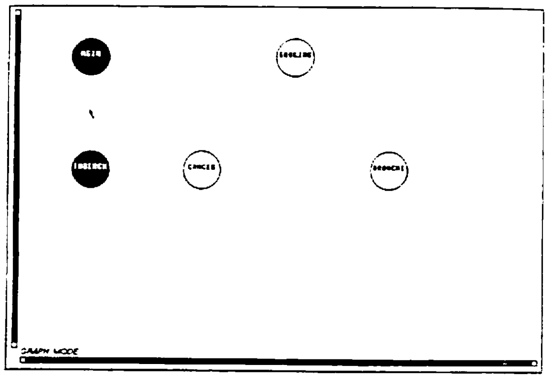

Figure 2. Indication of variable nodes to be included in a joint variable. The mouse is positioned where the user wishes to place the joint variable node.

An additional feature of the system is the option to define belief functions on a joint variable using logical expressions involving the individual variables. We can express the meaning of ASIA-TUBERCULOSIS as the statement "if ASIA = YES then TUBERCULOSIS = YES cf 0.5". Such expressions are translated to and stored internally as belief functions by the system.

We can also define multiple belief functions (or logical expressions) for a single node. To express the constraint "if A then B else C", we define the two belief functions representing "if A then B" and "if NOT A then C", and the system combines them using Dempster's rule to yield the intended meaning.

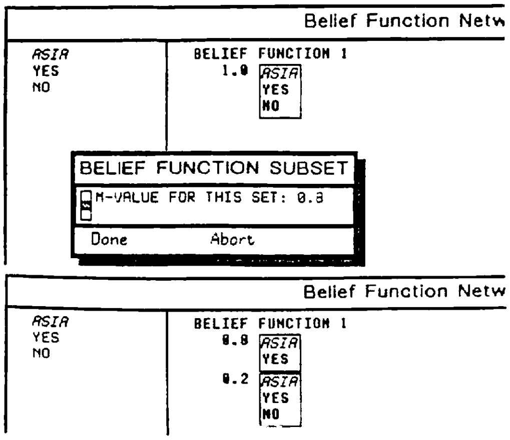

Figure 4. Providing evidence for the variable ASIA. a) The frame subset {YES} has been highlighted and an m-value of 0.8 has been specified for the subset; b) the resulting belief function.

To complete our example, Figure 5 shows the belief functions which were defined on the joint variables and on the variable DYSPNOEA. The remaining variables represent questions which we wish to answer but for which we have no direct evidence. The input for these variables remains the default - the vacuous belief function.

3.3 Providing Evidence - The Bayesian Case

For a Bayesian analysis, inputting evidence means providing prior probabilities for some individual variables and probability transition matrices for joint variables. In

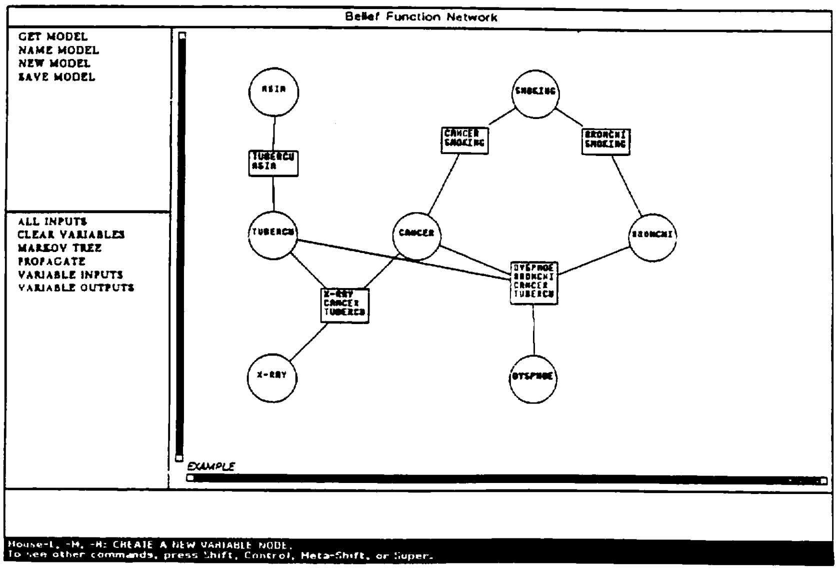

Figure 3. The completed model for our hypothetical example.

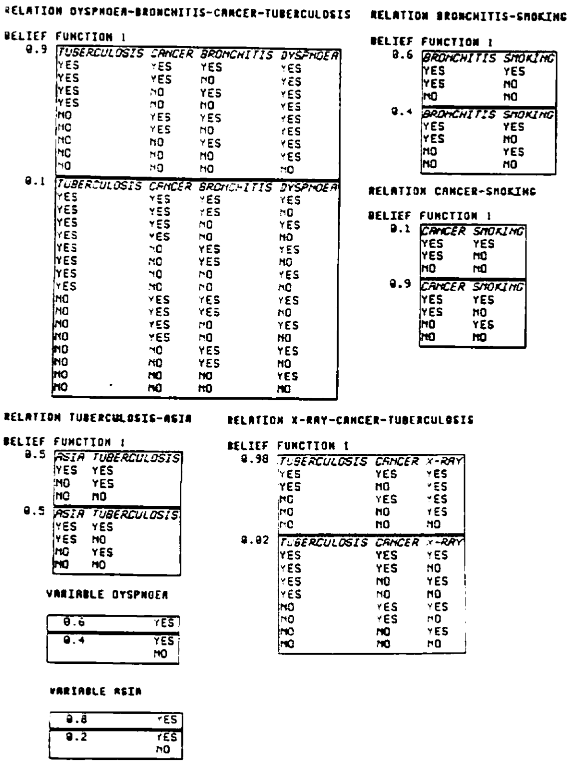

Figure 5. Evidence for the belief function example.

specifying the probability transition matrix, the user must specify which of the individual variables is the effect variable (Figure 6). The others are treated as causes. The system presents the user with a transition matrix, the values of which may be modified as desired. The user must provide evidence for *every* element of the space frame(cause) × frame(effect).

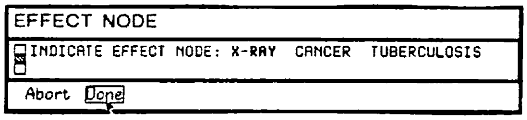

Figure 6. Specification of the effect node X-RAY for joint variable X-RAY-CANCER-TUBERCULOSIS via a pop-up window .

The system translates the transition matrices into belief functions and stores them in that form. In Figure 7, we have provided transition matrices for all joint variables and prior probabilities for the variables ASIA and SMOKING.

3.4 Propagating the Evidence

To analyze the model, the user simply selects the menu item 'Propagate'. The system creates a Markov tree of variables representing the network and propagates the evidence in the Markov tree. The tree and the results of the propagation for the belief-function example are shown in Figure 8. Propagation results for the Bayesian example are shown in Figure 9.

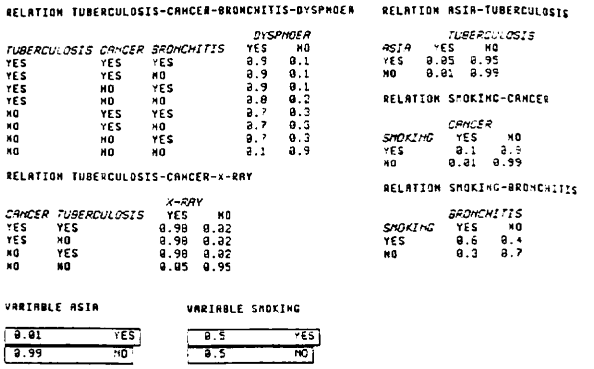

Figure 7. Evidence for the Bayesian example.

3.5 Other Operations

All aspects of a model and its associated evidence may be freely modified at any time. Variable nodes may be moved, deleted, renamed and reframed. Joint variable nodes may be moved and deleted. Evidence for any graph node may be 'cleared', i.e. the belief function for the node defaults to a vacuous one. These operations are defined on mouse buttons when the mouse is over a graph node.

Deleting an individual variable can have one of two effects on the joint variables of which it is a member. If deletion of the variable leaves the joint variable with only one remaining member, then the joint variable is also deleted. Otherwise, the joint variable's belief function is projected to the frame of its remaining member variables. If a variable's frame is changed, the value of all adjacent joint variable nodes and the variable node is projected to the new frame, if possible. Otherwise, the values for those nodes becomes vacuous. For Bayesian analysis, variable node deletion is somewhat more complicated. Deletion of a variable that is the effect variable of a joint variable causes such a joint variable to be deleted also, regardless of how many remaining member variables it may have.

4 Conclusion

DELIEF provides a flexible and easy-to-use interface for use in modeling and analyzing real-world problems. Knowledge schemas are easily represented as graphical networks of variables, for which both the structure and corresponding evidence can be easily modified.

The system implements both belief function and Bayesian analysis of knowledge schemas using one formalism, the Dempster-Shafer theory of evidence. The choice of a Bayesian analysis implies that the user has at hand all evidence

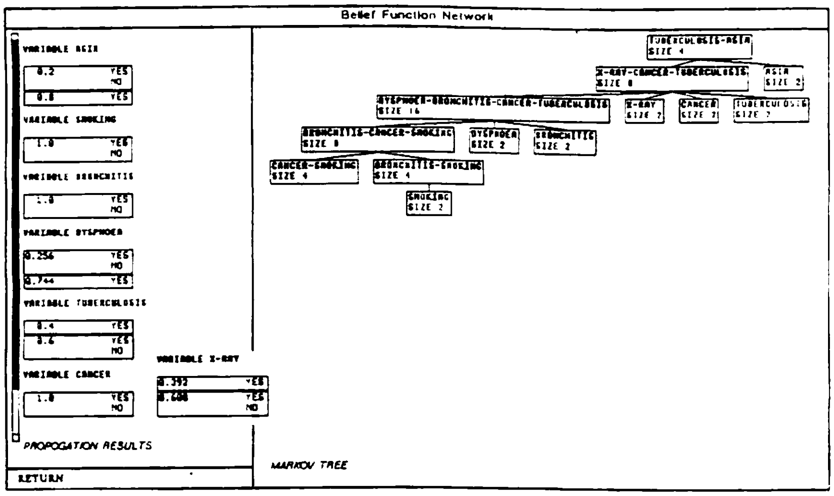

Figure 8. Markov tree for the model and results of propagation of the evidence shown in Figure 5 (belief function example).

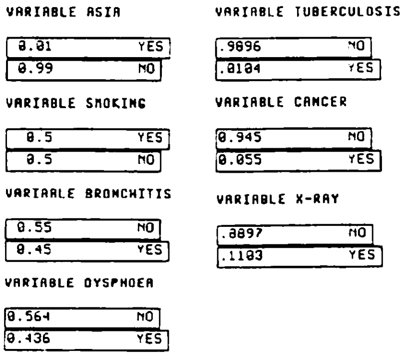

Figure 9. Results of propagation of evidence shown in Figure 7 (Bayesian example).

necessary to define a joint probability distribution for all relevant variables. The choice of a belief function analysis implies that the user has limited access to evidence, or is delaying getting costly evidence, but wishes to get meaningful answers to questions nevertheless.

References

[Biswas and Anand 1987] G. Biswas and T. S. Anand. Using the Dempster-Shafer scheme in a diagnostic expert system shell. *Proceedings of the Third Workshop on Uncertainty in Artificial Intelligence*, Seattle, WA, pages 98-105, 1987.

[Kong 1986] A. Kong. Multivariate belief functions and graphical models. Doctoral dissertation, Department of Statistics, Harvard University, 1986.

[Lauritzen and Spiegelhalter 1988] S. L. Lauritzen and D. J. Spiegelhalter. Local computations with probabilities on graphical structures and their application to expert systems. To appear in *Journal of the Royal Statistical Society*, Series B, 50, 1988.

[Lowrance et al. 1986] J. D. Lowrance, T. D. Garvey, and T. M. Strat. A framework for evidential-reasoning systems. *Proceedings of the fifth National Conference on AI*, AAAI-86, Philadelphia, PA, pages 896-903, 1986.

[Mellouli 1987] K. Mellouli. On the propagation of beliefs in networks using the Dempster-Shafer theory of evidence. Doctoral dissertation, School of Business, University of Kansas, 1987.

[Pearl 1986] J. Pearl. On the logic of probabilistic dependencies. *Proceedings of the fifth National Conference on AI*, AAAI-86, Philadelphia, PA, pages 339-343, 1986.

[Shafer 1976] G. Shafer. *A Mathematical Theory of Evidence*. Princeton University Press, Princeton, NJ, 1976.

[Shafer et al. 1987] G. Shafer, P. P. Shenoy, and K. Mellouli. Propagating belief functions in qualitative Markov trees. *International Journal of Approximate Reasoning*, 1:349-400, 1987.