Natural Language Engineering **3** (2/3): 123–145. Printed in the United Kingdom

© 1997 Cambridge University Press

Uniform knowledge representation for language processing in the B2 system

SUSAN W. MCROY*, SYED S. ALI†

*Department of Electrical Engineering and Computer Science, †Department of Mathematical Sciences, University of Wisconsin–Milwaukee, Milwaukee, WI 53201, USA

SUSAN M. HALLER

Computer Science and Engineering Department, University of Wisconsin–Parkside, Kenosha, WI 53141, USA

(*Received 25 April 1997*)

Abstract

We describe the natural language processing and knowledge representation components of B2, a collaborative system that allows medical students to practice their decision-making skills by considering a number of medical cases that differ from each other in a controlled manner. The underlying decision-support model of B2 uses a Bayesian network that captures the results of prior clinical studies of abdominal pain. B2 generates story-problems based on this model and supports natural language queries about the conclusions of the model and the reasoning behind them. B2 benefits from having a single knowledge representation and reasoning component that acts as a blackboard for intertask communication and cooperation. All knowledge is represented using a propositional semantic network formalism, thereby providing a *uniform* representation to all components. The natural language component is composed of a generalized augmented transition network parser/grammar and a discourse analyzer for managing the natural language interactions. The knowlege representation component supports the natural language component by providing a uniform representation of the content and structure of the interaction, at the parser, discourse, and domain levels. This uniform representation allows distinct tasks, such as dialog management, domain-specific reasoning, and meta-reasoning about the Bayesian network, to all use the same information source, without requiring mediation. This is important because there are queries, such as *Why?*, whose interpretation and response requires information from each of these tasks. By contrast, traditional approaches treat each subtask as a "black-box" with respect to other task components, and have a separate knowledge representation language for each. As a result, they have had much more difficulty providing useful responses.

1 Introduction

Tools for medical decision analysis offer doctors a systematic way to interpret new diagnostic information or to select the most appropriate diagnostic test. These tools support a doctor's practical experience with quantitative information about how diagnostic tests affect the probability that the patient has a certain disease, according to studies of similar patients.

https://doi.org/10.1017/S1351324997001745 Published online by Cambridge University Press

Building decision support systems involves the collection and representation of a large amount of medical knowledge. It also involves providing mechanisms for reasoning over this knowledge efficiently. To make the best use of these efforts, our project group, which involves researchers at the University of Wisconsin-Milwaukee, the University of Wisconsin-Parkside and the Medical College of Wisconsin, is working on a system to redeploy our decision support tools to build new systems for educating medical students. Our aim is to give medical students an opportunity to practice their decision making skills by considering a number of medical cases that differ from each other in a controlled manner. We also wish to give students the opportunity to ask the system to explain what factors most influenced the system.

The explanation of statistical information such as conditional probabilities presents unique problems for explanation generation. Although probabilities provide a good model of uncertain information, the reasoning that they support differs significantly from how people think about uncertainty (Kahneman *et al.* 1982). Probabilistic models are composed of a large number of numeric relationships that interact in potentially non-intuitive ways. Each state-value pair in the model (e.g. gallstones is present) may at once serve as possible conclusion to be evaluated and as evidence for some other conclusion. What emerges are chains of influence, corresponding to systems of conditional probability equations through which changes to probability values will propagate. Another difficulty in understanding probability models is the fact that the numeric relations alone provide no information about their origin (e.g. whether they reflect causation, constituency, or arbitrary co-occurrence). An explanation system thus needs to explain the local relations that comprise the model, the global dependencies that arise, and must be able to explain the relationship between the numerical data and the world knowledge that underlies it.

Natural language interactions can facilitate a fine-grained understanding of statistical models by allowing users to describe or select components of the model and to ask questions about their numeric or symbolic content. Natural language interactions can also facilitate a global understanding of such models, by providing summaries of important results or by allowing the user to describe events or results and ask questions about them. Lastly, an interactive system can adapt to different users' abilities to assimilate new information by presenting information in a conversational manner, and by tailoring the interaction to the users' concerns and apparent level of understanding. This paper describes the natural language and knowledge representation components of B2, a tutoring system that helps medical students learn a statistical model for medical diagnosis. B2 does this by generating story problems and supporting natural language dialogs about the conclusions of the model and the reasoning behind them.

2 Background

2.1 The need for uniform knowledge representation

The B2 system is comprised of three distinct, but interrelated tasks that rely on a variety of information sources. The tasks are:

https://doi.org/10.1017/S1351324997001745 Published online by Cambridge University Press

- Managing the interaction between the user and B2, including the interpretation of context-dependent utterances.
- Reasoning about the medical domain, including the relation between components of a medical case history and diseases that might occur.
- Meta-reasoning about the Bayesian reasoner and its conclusions, including an ability to explain the conclusions by identifying the factors that were most significant.

The tasks interact by addressing and handling queries to each other. However, the knowledge underlying these queries and the knowledge needed to generate a response can come from a variety of knowledge sources. Translating between knowledge sources is not an effective solution.

The information sources that B2 uses include:

- Linguistic knowledge – knowledge about the meanings of utterances and plans for expressing meanings as text.
- Discourse knowledge – knowledge about the intentional, social, and rhetorical relationships that link utterances.
- Domain knowledge – factual knowledge of the medical domain and the medical case that is under consideration.
- Pedagogy – knowledge about the tutoring task.
- Decision-support – knowledge about the statistical model and how to interpret the information that is derivable from the model.

In B2, the interaction between the tasks is possible because the information for all knowledge sources is represented in a uniform framework. The knowledge representation component serves as a central 'blackboard' for all other components.

2.2 Lessons from prior work: The need for a discourse model

The first prototype of our current system is Banter (Haddawy *et al.* 1996). Banter is a tutoring shell that generates word problems and short-answer questions on the basis of stored information about a particular medical situation, such as a patient who sees her doctor complaining of abdominal pains. This information comprises statistical relations among known aspects of a patient's medical history, findings from physical examinations of the patient, results of previous diagnostic tests and the different candidate diseases. The information is represented as a Bayesian belief network. The Banter shell has been designed to be general enough to be used with any network having nodes of hypotheses, observations and diagnostic procedures.

The output of Banter includes the prior and posterior probabilities (i.e. before and after any evidence such as symptoms or tests are taken into consideration) of a candidate disease, and the best test for ruling out or ruling in a disease, given the details of a case. It also includes a facility for explaining the system's reasoning to the student, showing her the paths in the belief network that were most significant in determining the probability calculations.

https://doi.org/10.1017/S1351324997001745 Published online by Cambridge University Press

```
After presenting the evidence, the probability of GALLSTONES is 0.135.

The tests for ruling GALLSTONES in, in order of effectiveness, are:
  A positive CT test results in a post-test probability of 0.976.
  A positive ULTRASOUND FOR GALLSTONES test results in a post-test
    probability of 0.445.
  A positive HIDA test results in a post-test probability of 0.307.
  A positive ULTRASOUND FOR CHOLECYSTITIS test results in a post-test
    probability of 0.247.

Their influence flows along the following paths:
  GALLSTONES is seen by CT
  GALLSTONES is seen by ULTRASOUND FOR GALLSTONES
  GALLSTONES causes CHOLECYSTITIS, which is detected by HIDA
  GALLSTONES causes CHOLECYSTITIS, which is detected by ULTRASOUND
    FOR MURPHY'S SIGN, which is a result of ULTRASOUND FOR CHOLECYSTITIS
  GALLSTONES causes CHOLECYSTITIS, which is detected by ULTRASOUND FOR
    THICK GB WALL, which is a result of ULTRASOUND FOR CHOLECYSTITIS
```

Fig. 1. An example explanation from Banter.

A preliminary (and informal) user study of the Banter system with students at the Medical College of Wisconsin revealed two important facts: First, students like the idea of being able to set up hypothetical cases and witness how different actions might (or might not!) affect the statistical likelihood of a candidate diagnosis. Second, students do not like, and will not use, a system that overwhelms them with irrelevant information or that risks misleading them because it answers questions more narrowly than a teacher would.

The problem is that the explanations that Banter provides mirror the structure of the chains of influence that produced the answer, including small steps that people find irrelevant and confusing. For example, Banter produces the explanation shown in figure 1 for why a CT scan would be the best test for ruling in gallstones, given the evidence of the case.

Our new work with this system focuses on improving its usability and usefulness as an educational tool. We began by generating a series of mockups for (informal) consideration by students and faculty at the Medical College of Wisconsin. The feedback that we received indicated that students preferred explanations that highlighted the most significant pieces of evidence. Consistent with empirical studies (Carbonell 1983), they preferred being able to ask brief context-dependent questions, such as *Why CT?* or *What about ultrasound?* and they preferred to give brief, context-dependent responses. Moreover, they liked explanations that were tailored to their needs – sometimes only a single word answer, sometimes the answer along with its justification. The new system, B2, can provide this customization by generating explanations incrementally, over a sequence of exchanges, while at the same time making it easier for students to request other types of clarifying information.

https://doi.org/10.1017/S1351324997001745 Published online by Cambridge University Press

2.3 Lessons from other systems: The need for an explanation component

Early attempts at explaining the reasoning produced by decision support systems focused on determining the types of queries that were possible and, for each type, writing rules to access appropriate information in the reasoning chain (Kukich 1985). More recent work on explaining Bayesian networks has been similar, focusing on generating verbal descriptions of the local relations that comprise the network (Elsaesser 1989; Norton 1988; Sember and Zukerman 1989), describing the generalizations of the numerical information qualitatively (Druzdel 1996), presenting the information in the context of a (template-based) scenario (Druzdel and Henrion 1990; Henrion and Druzdel 1991), or depicting numerical quantities graphically (Cole 1989; Madigan *et al.* 1994). The problem is that these systems analyze and answer carefully formulated queries the same way each time. The explanations produced are stiff and are closely mapped to the reasoning trace that produced the recommendation, which might be very different from how a person would conceptualize the problem (Slotnick and Moore 1995).

Another common problem that we found is that the explanations provided by decision support systems violate people's expectations for co-operative interaction Grice 1975. For example, they might fail to distinguish old information from new or typical information from exceptional. Thus, methods from natural language processing and human–computer interaction are needed to improve computer-generated explanations. These methods require simultaneous access to linguistic, discourse, domain, pedagogical and decision-support knowledge.

3 The B2 system

The new system under development, B2, extends Banter with the ability to generate well-structured, natural-language answers and to produce them in a manner that is consistent with principles for co-operative communication. The revised application also integrates multiple modalities so that students can refer to sequences of actions as well as to elements of the ongoing verbal communication. In figure 2, we include a dialogue from B2.

Our approach to the problems that we have described is to augment the reasoning chains produced by the Bayesian network with two types of knowledge. First, we have added static knowledge about the medical domain that Banter reasons about. For example, B2 represents that gallstones is a disease and that ultrasound is a diagnostic test. Second, we have added a discourse model. Using the discourse model, we can represent the content and the structure of the system's and the user's utterances from other modalities (such mouse-clicks), rather than simply devising mechanisms for producing or interpreting them. Thus, B2 can interpret questions and answers, such as *Why?* or *HIDA*, that assume knowledge of previous discourse.

3.1 The domain model

The domain knowledge in the B2 system is made up of the Bayesian decision model and the domain-dependent facts, including the *medical cases*, *tests*, *diseases*

https://doi.org/10.1017/S1351324997001745 Published online by Cambridge University Press

Doc: ⟨*Selected STORY PROBLEM from ACTION menu of the GUI*⟩
B2: Mr. Jones is of unknown age. He presents with diarrhea, denies anorexia and obstipation, and had no similar symptoms previously. His temperature is 39.04. His white-blood count is unknown. Physical examination reveals rebound-tenderness, but no evidence of abnormal bowel sounds.

What is the best test to rule in gallstones?
Doc: HIDA
B2: No, it is not the case that HIDA is the best test to rule in gallstones. CT is the best test to rule in gallstones.

Doc: Why?
B2: After presenting the evidence in this case, the probability of gallstones is 0.135. A positive CT test would increase the probability to 0.976. A positive HIDA would only increase the probability to 0.307.

Doc: What about ultrasound?
B2: A positive ultrasound for gallstones would result in a post-test probability of 0.445.
Doc: What if there had been abnormal bowel sounds?
B2: The posterior probability would be 0.130
Doc: ok

Fig. 2. A B2 dialogue.

and *outcomes*. The domain-dependent facts that we represent include both general information such as *HIDA, Ultrasound and CT are tests* as well as more specific information based on specific clinical cases (a *case history*) that are used as examples to test students' diagnostic skills. A case history consists of patient medical history items, the results of a physical examination, and the results of various medical tests.

The Bayesian network is specified as a sequence of state names (e.g. Gallstones, Sex or HIDA) and a table of posterior probabilities. For each possible value of a state, and each possible value of the states directly connected to it, the table indicates the posterior probability of the state-value combination. This information is provided to the probabilistic reasoner. In addition, B2 converts the specification into a propositional representation that captures the connectivity of the original Bayesian network. Such a representation allows B2 to answer questions such as *Why is Gallstones suspected?*, which would require B2 to identify paths among nodes in the network and identify those that were most influential in the probability calculations. (Such a question is answered by asking the probabilistic reasoner to evaluate the posterior probability of Gallstones, given only the nodes of a particular path under consideration: Suermondt 1992; Haddawy *et al.* 1996.)

3.2 The discourse model

The discourse model combines information about the discourse level actions performed by B2, as well as B2's interpretation of the user's utterances. The content

https://doi.org/10.1017/S1351324997001745 Published online by Cambridge University Press

B2: *What is the best test to rule in Gallstones?*
Doc: `HIDA.`
B2: *No, CT.*
Doc: `Ok.`

Fig. 3. A dialogue between B2 and a medical student.

What was said
u1 The system asked *What is the best test to rule in gallstones?.*
u2 The user said *HIDA.*
u3 The system said *No.*
u4 The system said *CT.*
u5 The user said *Ok.*

How B2 interprets what was said
i1 The system asks *What is the best test to rule in gallstones?.*
i2 The user answers *HIDA is the best test to rule in gallstones.*
i3 The system disconfirms *No, HIDA is not the best test to rule in gallstones*
i4 The system informs *CT is the best test to rule in gallstones.*
i5 The user agrees that *CT is the best test to rule in gallstones.*

Relations
r1 i1 is the interpretation of u1
r2 i2 is the interpretation of u2
r3 i3 is the interpretation of u3
r4 i4 is the interpretation of u4
r5 i5 is the interpretation of u5
r6 i2 accepts i1
r7 i3 accepts i2
r8 i5 accepts i4
r9 i4 justifies i3

Fig. 4. Propositions that would be added to B2's model of the discourse.

of this model combines information that is used by the system to plan utterances (based on Haller 1996) with information that is inferred by the system as a result of its interpreting the user's utterances (based on McRoy 1995*b*; McRoy and Hirst 1995*a*). The B2 system provides a uniform representation for these two types of information. Consider the dialogue shown in figure 3.

In the discourse model, this dialogue leads to the assertion of a number of propositions about what was said by each participant, how the system interpreted what was said as an action, and how each discourse action relates to prior ones. For the exchange above, B2's knowledge representation would include representations of facts that can be glossed as shown in figure 4. (The form of the representation is discussed in section 3.3; a more detailed example is given in section 5.)

This model of the discourse is used to both interpret users' utterances and to generate B2's responses. When a user produces an utterance, the parser will generate a representation of its surface content (e.g. a word, phrase, or sentence) and force (e.g. ask, request, say). B2 then uses its model of the discourse to build an

https://doi.org/10.1017/S1351324997001745 Published online by Cambridge University Press

interpretion of the utterance that captures both its complete propositional content and its relationship to the preceding discourse. Having this discourse model allows B2 greater flexibility than previous explanation systems, because it enables the system to judge whether:

- The student understood the question that was just asked and has produced a response that can be evaluated as an answer to it; or
- The student has rejected the question and is asking a question of her own; or
- The student has misunderstood the question and has produced a response that can be analyzed to determine how it might repair the misunderstanding.

Conversely, B2 uses the knowledge representation to select an appropriate response (e.g. confirm, disconfirm) and to realize the response as a natural language utterance. The utterance will include rhetorical devices that emphasize important details and will omit information that would be redundant or irrelevant given the preceding discourse. After B2 generates its utterance, the discourse model will be augmented to include a representation of the response and its relation to the preceding discourse, as well as a representation of the surface content and force used to express it. Both representations are useful; they allow the system to produce a focused answer to a question like *Why?*, yet still be able to respond to requests for more information, such as *What about ultrasound?*. Below we discuss these mechanisms and the overall architecture of B2 in greater detail.

3.3 The knowledge representation blackboard

B2 represents both domain knowledge and discourse knowledge in a uniform framework as a propositional semantic network. A propositional semantic network is a framework for representing the concepts of a cognitive agent who is capable of using language (hence the term *semantic*). The information is represented as a graph composed of nodes and labeled directed arcs. In a *propositional* semantic network, the propositions are represented by the nodes, rather than the arcs; arcs represent only non-conceptual binary relations between nodes. The particular systems that are being used for B2 are SNePS and ANALOG (Shapiro and Group 1992; Ali 1994*a*; Ali 1994*b*). These systems satisfy the following additional constraints:

1. Each node represents a unique concept.
2. Each concept represented in the network is represented by a unique node.
3. The knowledge represented about each concept is represented by the structure of the *entire network* connected to the node that represents that concept.

These constraints allow efficient inference when processing natural language. For example, such networks can represent complex descriptions (common in the medical domain), and can support the resolution of ellipsis and anaphora, as well as general reasoning tasks such as subsumption (Ali 1994*a*; Ali 1994*b*; Maida and Shapiro 1982; Shapiro and Rapaport 1987; Shapiro and Rapaport 1992).

We term a knowledge representation *uniform* when it allows the representation of different kinds of knowledge in the same knowledge base using the same inference

https://doi.org/10.1017/S1351324997001745 Published online by Cambridge University Press

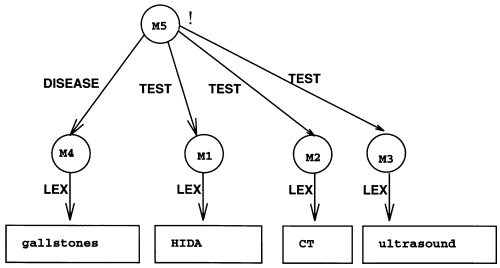


Fig. 5. A simple network representing the proposition *HIDA, CT, and ultrasound can be used to test for gallstones.*

processes. The knowledge representation component of B2 is uniform because it provides a representation of the discourse knowledge, domain knowledge, and probabilistic knowledge (from the Bayesian net). This supports intertask communication and cooperation for interactive processing of tutorial dialogs.

To achieve this uniform representation, the knowledge representation uses four types of nodes: *base*, *molecular*, *variable* and *pattern.*

Base nodes are nodes that have no arcs emanating from them. They are used to represent atomic concepts.

Molecular nodes have arcs emanating from them. They represent propositions, rules, and structured concepts.

Variable nodes represent arbitrary individuals. Like base nodes, variable nodes have no arcs emanating from them. They correspond to variables in predicate logic.

Pattern nodes represent arbitrary propositions. They correspond to open sentences in predicate logic.

Propositions are represented using molecular nodes. Case frames are conventionally agreed upon sets of arcs emanating from a node used to express a proposition. For example, to express that A isa B we use the MEMBER-CLASS case frame which is a node with a MEMBER arc and a CLASS arc. Shapiro *et al.* (1994) provides a dictionary of standard case frames. Additional case frames can be defined as needed.

Figure 5 is an example of a network that uses base nodes and molecular nodes to represent the system's knowledge that *HIDA, CT, and ultrasound can be used to test for gallstones.* Node `M5` is the molecular node that represents this proposition using the DISEASE-TEST case frame. The assertion flag (exclamation mark beside the node) indicates that the system believes that this proposition is true. The system represents all propositions that are believed to be true as *asserted molecular nodes.* `HIDA` and `gallstones` are base nodes, representing atomic concepts.

Figure 6, a somewhat more complex example, shows a network that uses variable nodes and pattern nodes. It illustrates a text plan for describing a medical case to the user. In the knowledge representation, text plans are represented as rules. Rules are general statements about objects in the domain; they are represented as molecular nodes that have `FORALL` or `EXISTS` arcs to variable nodes (these variable nodes correspond to the quantified variables of the rule.)

https://doi.org/10.1017/S1351324997001745 Published online by Cambridge University Press

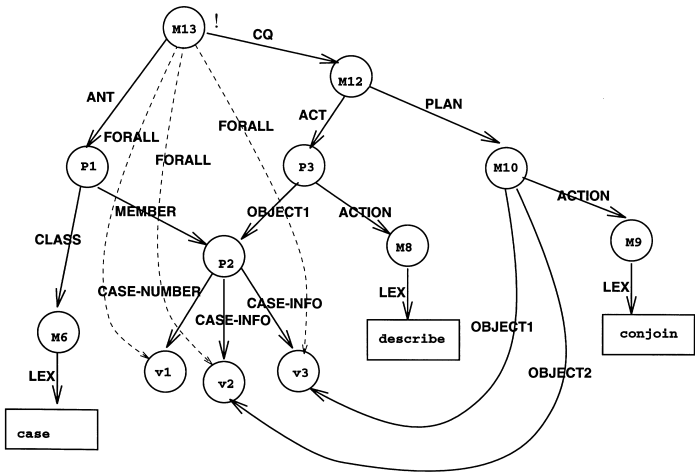


Fig. 6. A rule stating that if V1 is the case number of a case, and V2 and V3 are two pieces of case information, then a plan for generating a description of the case will present the two pieces of information in a coordinating conjunction.

In figure 6, node M13 is a molecular node representing a rule with three universally quantified variables (at the end of the FORALL arcs), an antecedent (at the end of the ANT arc), and a consequent (at the end of the CQ arc). This means that if an instance of the antecedent is believed, then a suitably instantiated instance of the consequent is believed. M13 states that if V1 is the case number of a case, and V2 and V3 are two pieces of case information, then a plan to describe the case will conjoin[1] the two pieces of case information. Node V1 is a variable node. Node P1 represents the concept that something is a member of the class case and P2 represents the concept that the case concept has a case number and case information.

The rule in figure 6 is a good example of how the uniform representation of information in the semantic network allows us to relate domain information (a medical case) to discourse planning information (a plan to describe it).

In addition to knowledge about the domain and about the discourse, there is an explicit representation of the connectivity of the Bayesian network. The discourse analyzer uses this connectivity information to identify chains of reasoning that underly the system's diagnostic recommendations. These chains will be needed by the discourse analyzer to explain the system's probability calculations. For example, to answer a question such as *Why is gallstones suspected?* or *Why does a positive CT test support gallstones?* the discourse analyzer must find sequences of conditionally dependent nodes that terminate at the node corresponding to gallstones. Then the discourse analyzer queries the Bayesian reasoner to determine the significance of each such reasoning chain. Because the connectivity information is represented declaratively in a uniform framework, B2 will be able to relate the probabilistic

[1] 'Conjoin' is a technical term from Rhetorical Structure Theory (Mann and Thompson 1986); it refers to a co-ordinate conjunction of clauses.

https://doi.org/10.1017/S1351324997001745 Published online by Cambridge University Press

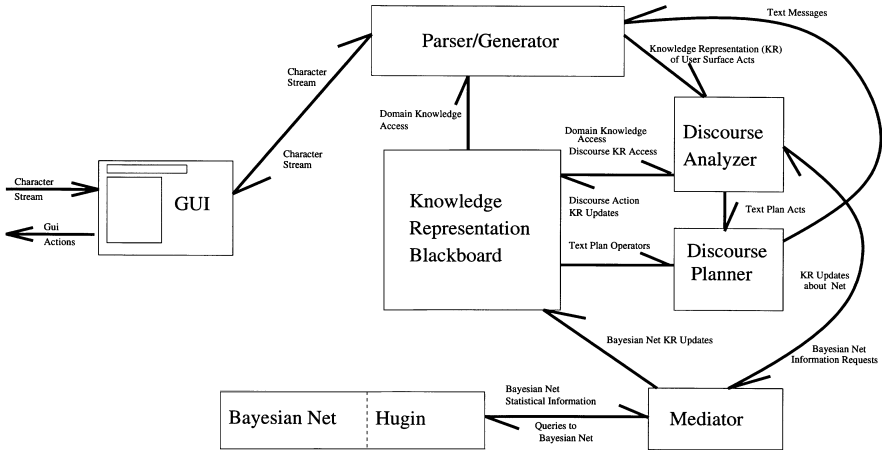


Fig. 7. The B2 architecture.

information to other information that it has about the medical domain (such as that cholecystitis is a type of inflammation), allowing the discourse analyzer to formulate appropriate generalizations when generating an explanation.

4 The B2 architecture

The B2 system consists of seven components (see figure 7). In the diagram, solid, directed arrows indicate the direction of information flow between components. The system gets the user's input using a graphical user interface that supports both natural language interaction and mouse inputs. The *Parser* component of the *Parser/Generator* performs the first level of processing on the user input using its grammar and the domain information from the *Knowledge Representation Blackboard*. The Parser interprets the user's inputs to form propositional representations of surface-level utterances for the Discourse Analyser. The *Generator* produces natural language outputs from the text messages (propositional descriptions of text) that it receives from the *Discourse Planner*.

The system as a whole is controlled by a module called the *Discourse Analyser*. The Discourse Analyser determines an appropriate response to the user's actions on the basis of a model of the discourse and a model of the domain, within the knowledge representation component. The Analyser invokes the *Discourse Planner* to select the content of the response and to structure it. The Analyser relies on a component called the *Mediator* to interact with the Bayesian network processer, *Hugin*. This Mediator processes domain level information, such as ranking the effectiveness of alternative diagnostic tests. The Mediator also handles the information interchange between the propositional information that is used by the Analyzer and the probabilistic data that is used by Hugin. All phases of this process are recorded in the *Knowledge Representation Component* resulting in a complete history of the discourse. Thus, the knowledge representation component serves as a central 'blackboard' for all other components.

https://doi.org/10.1017/S1351324997001745 Published online by Cambridge University Press

```
PROCESS-DIALOG
1. LOOP
2.    Get a surface-level interpretation of the user's input
3.    Identify the user's action (as ASK, REQUEST, or SAY)
4.    IF input is a REQUEST, THEN
5.       PROCESS-REQUEST
6.    ELSE IF input is a ASK, THEN
7.       PROCESS-QUESTION
8.    ELSE IF input is a SAY, THEN
9.       PROCESS-UTTERANCE
```

Fig. 8. The top-level discourse analyser algorithm.

During the initialization of the system, there is one-time a transfer of information from a file that contains a specification of the Bayesian network both to Hugin and to the Knowledge Representation Component. B2 converts the specification into a propositional representation that captures the connectivity of the original Bayesian network.

In the remainder of this section, we will consider these components in greater detail.

4.1 The discourse analyser and the discourse planner

All interaction between the user and the system is controlled by the Discourse Analyser. The Analyser calls upon the Parser and the Discourse Planner to interpret the user's surface-level utterances and respond to them. The analysis algorithm takes as input the propositional representations of the user's actions (that have been produced by the parser, see figure 8). Given the parsed input, the Analyser interprets it as either a request, a question, or a statement, taking both surface form and contextual information into account. The resulting interpretation and its relations to the context are then added to the knowledge representation blackboard. The last step of the algorithm is to call one of the system's discourse planning modules to formulate an appropriate response.

Discourse planning is handled by three independent modules: a request-processor, a question-processor, and a general utterance-processor.[2] The first two processors handle utterances in which the user is taking primary control of the dialogue. (See figures 9 and 10.) The third handles all utterances for which the system has control. (See figure 11.) The request-processor encodes text plans for two domain tasks: the presentation of a story problem (based on a newly selected case) and the presentation of quiz questions (based on a given case). The question-processor encodes a text plan for presenting a justification for inferences that can be drawn from a given

[2] Although B2 does not use it, the knowledge representation component does provide a framework for representing discourse plans declaratively and for building and executing such plans. In the next phase, these processing modules will be replaced by a single module that uses this planning and acting framework. At that point, the pedagogical knowledge will be part of the same uniform representation.

https://doi.org/10.1017/S1351324997001745 Published online by Cambridge University Press

PROCESS-REQUEST

1. IF the user is requesting a story, THEN
2. Build a story in the network:
3. Tell the story
4. **PROCESS-REQUEST** for a quiz question about the story
5. ELSE IF the user is requesting a quiz question about a story, THEN
6. Formulate a question about the story
7. Ask the question

Fig. 9. The request processing algorithm.

PROCESS-QUESTION

1. IF the user is asking "Why?" or "Why not?', THEN
2. Determine the proposition that is being questioned
3. Find the reasoning behind the proposition
4. Invoke the Discourse Planning to explain the reasoning
5. ELSE IF the user is asking "What about" *something*?, THEN
6. Determine the alternative proposition that is being questioned
7. **PROCESS-QUESTION** "Why" about this alternative proposition

Fig. 10. The question processing algorithm

case; questions can be posed directly (e.g. as a *why*-question) or indirectly (e.g. as a *what-if* or *what-about* question). The system's general utterance processor encodes text plans for handling answers and acknowledgements that have been produced by the user; presently, all other user actions are rejected, resulting in the system making a request for clarification.

4.2 The parser and generator

This component provides a morphological analyzer, a morphological synthesizer and an interpreter/compiler for generalized augmented transition network (GATN) grammars (Shapiro 1982). The parser and generator are integrated and use the same grammar. These tools are used in B2 to perform syntactic and semantic analysis of the user's natural language inputs and to realize the system's own natural language outputs. For all inputs, the parser produces a propositional representation of its surface content and force; for example, figure 12 shows the parser's output when B2 gets the input *HIDA*. Node `M103` represents the proposition that the agent `user` did `say HIDA`. Note that this is all the informaton that is available to the parser; within the discourse analyser, this action will be interpreted as an `answer` to a previous question by the system. Both actions (`say` and `answer`) will be included in B2's representation of the discourse history.

https://doi.org/10.1017/S1351324997001745 Published online by Cambridge University Press

PROCESS-UTTERANCE

1. IF the system just asked a question and the utterance is a potential answer to it, THEN,
2. IF this is a legitimate answer, on the basis of static domain information, THEN
3. Query the mediator for the correct answer.
5. Evaluate the answer against the correct answer.
6. IF correct answer THEN CONFIRM, ELSE DISCONFIRM answer and tell the correct answer
7. ELSE IF this is an acknowledgement, THEN
8. Continue processing at previous level
9. ELSE SEEK-CLARIFICATION

Fig. 11. The general utterance processing algorithm.

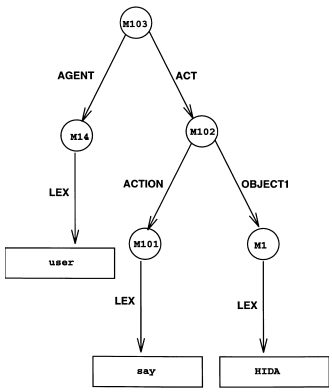


Fig. 12. Node M103 is the representation produced by the parser for the utterance *HIDA*. M103 represents the proposition *The user said HIDA*.

4.3 The mediator and hugin

The underlying Bayesian belief network was developed using Hugin, a commercial system for reasoning with probabilistic networks. Hugin allows one to enter and propagate evidence to compute posterior probabilities. The Mediator component of B2 translates information needs of the discourse planner into the command language of Hugin and translates the results into propositions that are added to the domain model in the knowledge representation component. For example, to assess the significance of the evidence, the mediator will instantiate alternative values for the different random variables and compare the results. After analysing and sorting the results, the mediator will generate a set of propositions (e.g. to assert which test was the most informative and what the reason was).

5 An example exchange

In this section, we provide a detailed example that shows how B2 processes a dialogue, showing the knowledge representations for the discourse history and the domain knowledge. The system uses this knowledge to analyse the user's response

https://doi.org/10.1017/S1351324997001745 Published online by Cambridge University Press

B2	`What is the best test to rule in gallstones?`	1
User:	`HIDA`	2
B2:	`No, it is not the case that HIDA is the best test to rule in gallstones.`	3
	`CT is the best test to rule in gallstones.`	4

Fig. 13. The example exchange.

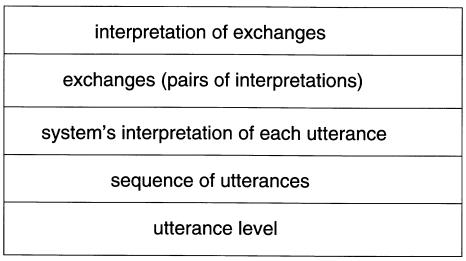


Fig. 14. Five levels of representation.

to a system-generated question and to determine that she has provided a reasonable (if not correct) answer. The actual dialogue is shown in figure 13.

In this section, we will discuss the representation of the discourse and how it is constructed one utterance at a time.

5.1 Representation of the discourse

The discourse has five levels of representation, shown in figure 14. We will consider each of these levels in turn, starting with the utterance level, shown at the bottom of the figure. For the user's utterances, the utterance level representation is the output of the parser (an event of type ASK, REQUEST or SAY). The content of the user's utterance is always represented by what she said literally. Figure 12 shows the representation of the user's first utterance, line 2 in figure 13. The content of the utterance is represented by the node at the end of an OBJECT1 arc, which is node `M1`. In figure 12, node `M103` represents the event of the user saying HIDA.

For the system's utterances, the utterance level representation corresponds to a text generation event (this contains much more fine-grained information about the system's utterance, such as mode and tense.) The content of the system's utterance is the text message that is sent to the language generator. In figure 15, node `M119` represents the event of the system making utterance 3 in figure 13. The content of the utterance is represented by node `M105`, a present tense sentence in declarative mode expressing the proposition that *HIDA is not the best test to rule in gallstones.*

The second level corresponds to the sequence of utterances. (This level is comparable to the linguistic structure in the tripartite model of Grosz and Sidner (1986).) In the semantic network, we represent the sequencing of utterances explicitly, with asserted propositions that use the BEFORE-AFTER case frame Shapiro *et al.* (1994).

https://doi.org/10.1017/S1351324997001745 Published online by Cambridge University Press

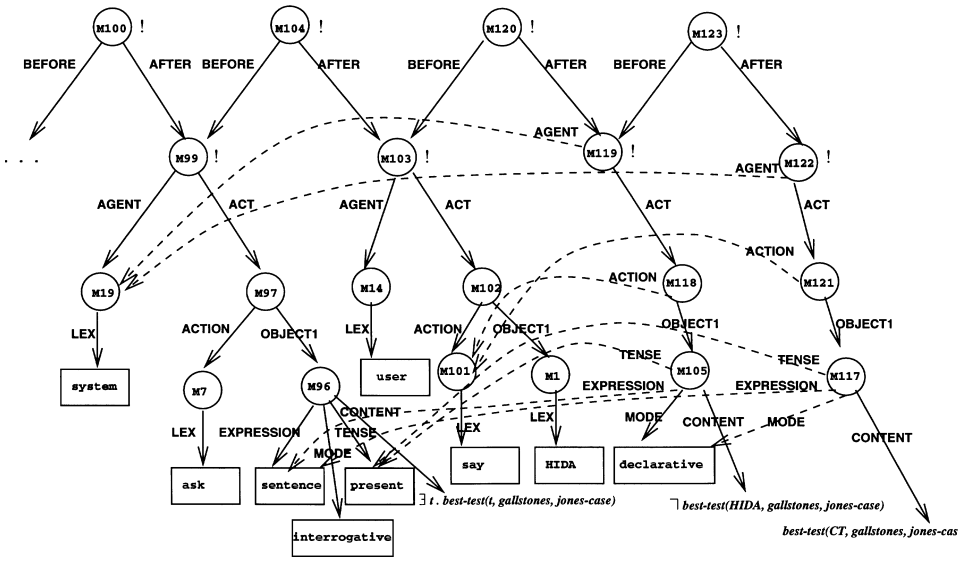


Fig. 15. Nodes M100, M104, M120, and M123 represent the sequence of utterances produced by the system and the user, shown in Figure 13. For example, Node M104 represents the proposition that the event M103 immediately followed event M99.

In figure 15, asserted propositional nodes M99, M103, M119, and M122 represent the events of utterances 1, 2, 3 and 4, respectively.[3] For example, node M103 asserts that the user said *HIDA*. Asserted nodes M100, M104, M120, and M123 encode the order of the utterances. For example, node M120 asserts that the event of the user saying *HIDA* (M103) occurred just before the system said *HIDA is not the best test to rule in gallstones* (M119).

In the third level, we represent the system's interpretation of each utterance. Each utterance event (from level 1) will have an associated system interpretation, which is represented using the INTERPRETATION_OF-INTERPRETATION case frame. Figure 16 gives a semantic network representation of utterance 2 and its interpretation. In the figure, node M103 corresponds to the proposition that *The user said HIDA* (the utterance event). Node M108 is the system's interpretation of the utterance event, that *The user answered that the best test to rule in gallstones is HIDA*. Node M109 represents the system's belief that M103) is interpreted as node M108.

The fourth and fifth levels of representation in our discourse model are exchanges and intepretations of exchanges, respectively. A *conversational exchange* is a pair of interpreted events that fit one of the conventional structures for dialog (e.g. QUESTION-ANSWER). Figure 17 gives the network representation of a conversational exchange and its interpretation. Node M113 represents the exchange in

[3] Propositional expressions that are written in italics, for example, ∃*t.best-test(t, gallstones, jones-case)*, represent subgraphs of the semantic network that have been omitted from the figure due to space restrictions.

https://doi.org/10.1017/S1351324997001745 Published online by Cambridge University Press

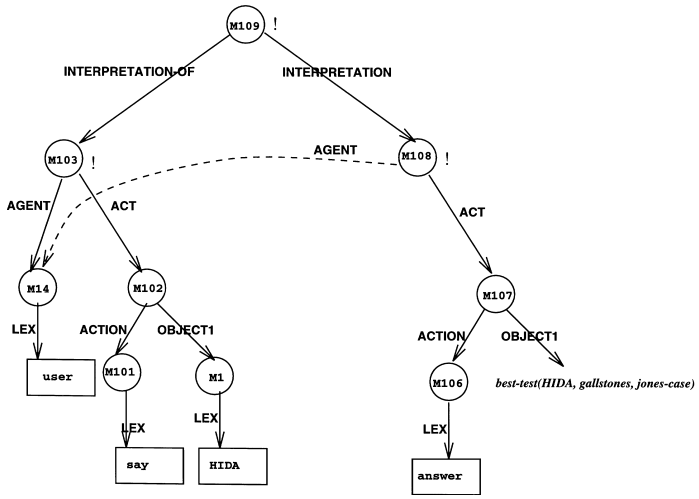


Fig. 16. Node M109 represents the system's interpretation of event M103, *The user said HIDA*. M109 is the proposition that the system's interpretation of M103 is M108. M108 is the proposition that *The user answered 'HIDA is the best test to rule in Gallstones'*.

which the system has asked a question and the user has answered it. Using the MEMBER-CLASS case frame, propositional node M115 asserts that the node M113 is an exchange. Propositional node M112 represents the system's interpretation of this exchange: that the user has accepted the system's question (i.e. that the user has understood the question and requires no further clarification). Finally, propositional node M116 represents the system's belief that node M112 is the interpretation of the exchange represented by node M113.

A major advantage of the network representation is the knowledge sharing between these five levels. We term this knowledge sharing *associativity*. This occurs because the representation is uniform and every concept is represented by a unique node (see section 3.3). As a result, we can retrieve and make use of information that is represented in the network implicitly, by the arcs that connect propositional nodes. For example, if the system needed to explain why the user had said HIDA, it could follow the links from node M103 (shown in figure 17) to the system's interpretation of that utterance, node M108, to determine that

- The user's utterance was understood as the answer within an exchange (node M113), and
- The user's answer indicated her acceptance and understanding of the discourse, up to that point M112.

This same representation could be used to explain why the system believed that the user had understood the system's question. This associativity in the network is vital if the interaction starts to fail.

https://doi.org/10.1017/S1351324997001745 Published online by Cambridge University Press

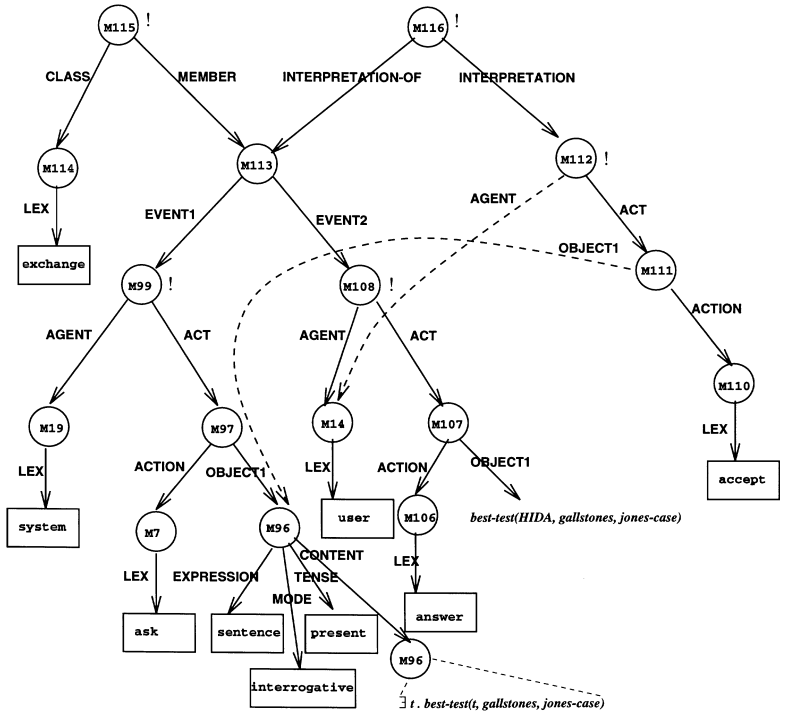


Fig. 17. Node M115 represents the proposition that node M113 is an exchange comprised of the events M99 and M108. Additionally, node M116 represents the proposition that the interpretation of M113 is event M112. M112 is the proposition that the user has accepted M96. (M96 is the question that the system asked in event M99.)

5.2 *Construction of the discourse model*

Now we will consider how the discourse model for the dialogue shown in figure 12 is built up one utterance at a time. The discussion will focus on:

- The generation of the system's question;
- The interpretation of the user's reply;
- The evaluation of the user's answer; and
- The generation of the system's response.

5.2.1 Generating the question

The decision to quiz the user by asking her a question is embedded in the **PROCESS-REQUEST** algorithm (section 4.1) as part of the system's response to a request for a story. That is, whenever, the system receives a request to tell a story, responding to the request involves telling the story and then asking the user a question about it. In figure 2, the user has requested that the system tell a story. The system selects a medical case for the story, describes it and asks the user a question about diagnostic tests based on the story. After generating the question, the discourse representation includes the proposition that the system has asked the question *What is the best*

https://doi.org/10.1017/S1351324997001745 Published online by Cambridge University Press

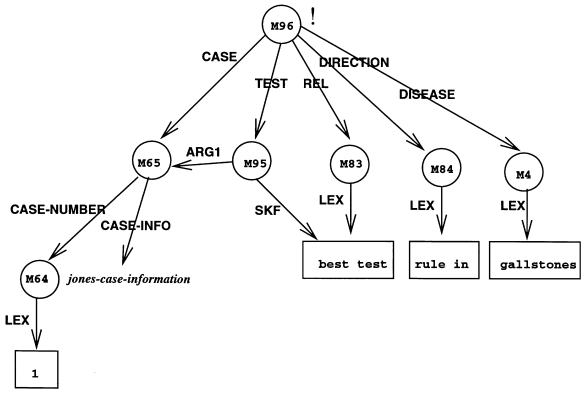


Fig. 18. Node M96 is the representation of a fact that has been selected by the discourse analyser to use as the basis for a question to quiz the student. M96 is the proposition that, for case 1, there is a best test to rule in gallstones.

test to rule in gallstones? (node M99, figure 15). In the network representation, the content of this question is represented using a skolem constant for the existentially quantified variable. Figure 18 is a more detailed representation of this content, where node M96 corresponds to the skolemized sentence

best-test(•t, gallstones, jones-case)

Node M95 represents the skolem constant •t as a function of the case in question (node M65).[4]

5.2.2 Interpreting 'HIDA'

When the user says *HIDA* (utterance 2), the system first checks to see whether the utterance is a request or a question. As it is neither, the Analyser must call the **PROCESS-UTTERANCE** procedure to interpret it. The linguistic model of the discourse at this point (node M99, figure 15) indicates that the system has just asked a question. The analyser attempts to interpret the user's utterance as an answer to the question. To do this, the node that represents the content of the question is retrieved from the network. Within this node, the skolem constant indicates the item being queried (in this example, the missing TEST item). To determine whether the user's utterance is reasonable (if not correct), the analyser searches the network for a concept or proposition that has the user's answer at the end of a TEST arc. (In other words, the system verifies that the content of the user's utterance is a TEST.) Such a node is found: node M5, figure 5.

Once the system has established that the user's utterance constitutes a reasonable answer to the question, the analyser builds a full representation of the answer by replacing the skolem constant (node M95 in figure 18) with node M1 from figure 5. The

[4] The complete structure of the case is not shown; we have abbreviated the subnetwork corresponding to the case information as *jones-case-information.*

https://doi.org/10.1017/S1351324997001745 Published online by Cambridge University Press

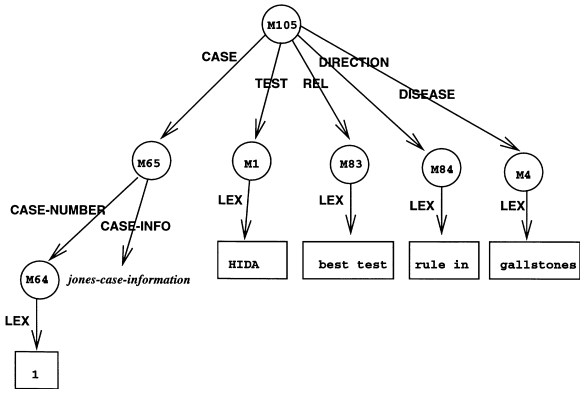


Fig. 19. Node M105 is the representation of the user's answer to the quiz question. M105 is the proposition that, for case 1, the best test to rule in gallstones is HIDA.

system's representation of the user's answer to the question is the final result, and is shown as node M105 in figure 19. In addition, the analyser adds to the discourse model that the user has answered *HIDA is the best test to rule in Gallstones* (node M108 figure 16). The Analyser also asserts (node M109 in figure 16) that this knowledge (node M108) is the system's interpretation of the user's literal utterance (node M103 figure 16). Thus, *HIDA* has two interpretations – at the utterance level, *The user said HIDA is the best test to rule in gallstones*, and at the intepretation level, *The user answered the system's question, 'What is the best test to rule in gallstones'.*

5.2.3 Evaluating the user's answer

The system queries the Bayesian net to obtain domain information. This information is necessary to build a propositional representation that contains the correct answer to the question (node M86, figure 20). The correct answer to the system's question is deduced using the question (figure 18) and the domain knowledge, and is asserted as node M117 (figure 21). Due to the uniqueness of nodes, if the user's answer were correct it would be this same node. However, the user's answer (node M105, figure 19) is different, meaning that the user gave an incorrect answer.

5.2.4 Producing the response

When the user's answer is incorrect, the system plans text to disconfirm the user's answer and state the correct answer. Planning these acts results in two surface-level utterance events shown in figure 15. Node M119 represents the system's statement that HIDA is not the best test for gallstones. Node M122 represents the system's subsequent statement that CT is the best test to rule in gallstones.

6 Current status of B2

B2 is being developed using the Common LISP programming language. We are using the SNePS 2.1 and ANALOG 1.1 tools to create the lexicon, parser, generator and

https://doi.org/10.1017/S1351324997001745 Published online by Cambridge University Press

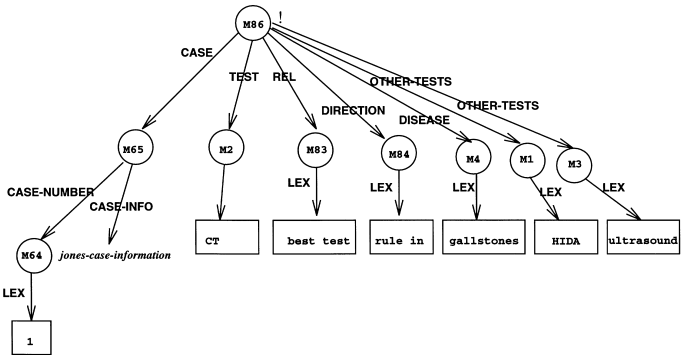


Fig. 20. Node `M86` is the fact that the discourse analyser uses when evaluating the user's answer to the quiz question (based on `M96`). `M86` is the proposition that, for case 1, the best test to rule in gallstones (considering only CT, HIDA, and Ultrasound) is CT.

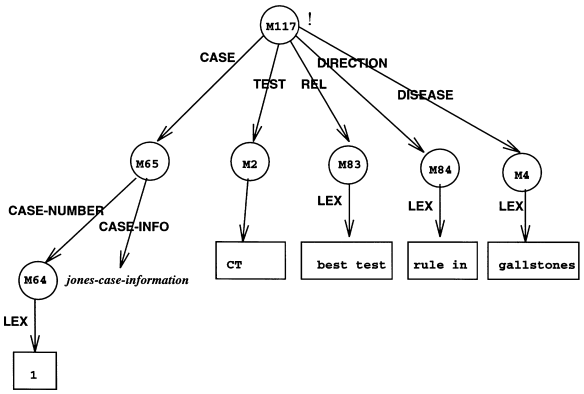


Fig. 21. Node `M117` is the answer to the quiz question that the system deduces from the relevant domain knowledge (Node `M96`). `M117` is the proposition that CT is the best test to rule in gallstones.

underlying knowledge representations of domain and discourse information (Shapiro and Group 1992; Shapiro and Rapaport 1992; Ali 1994*a*; Ali 1994*b*). Developed at the State University of New York at Buffalo, SNePS (Semantic Network Processing System) provides tools for building and reasoning over nodes in a propositional semantic network.

7 Conclusions

The goal of the B2 project is to give medical students an opportunity to practice their decision making skills. This is done by considering a number of medical cases that differ from each other in a controlled manner. We also want to give students the opportunity to ask the system to explain what factors were most influential to its decision and why. The primary means for these interactions is natural language.

Our prior experience suggests that it is necessary for a tutoring system to obey

https://doi.org/10.1017/S1351324997001745 Published online by Cambridge University Press

the conventions of natural language interaction. In particular, B2 takes into account the changing knowledge of the student when interpreting her actions, to avoid presenting redundant or irrelevant information that might be confusing. The B2 system also allows students to direct the system's explanation to their own needs, making it easier for them to learn and remember the information.

The knowlege representation component supports natural language interaction. It does so by providing a uniform representation of the content and structure of the interaction at the parser, discourse, and domain levels. This uniform representation allows distinct tasks, such as dialog management, domain-specific reasoning, and meta-reasoning about the Bayesian network, to cooperatively use the same information source.

We believe that a uniform knowledge representation framework, such as that used in B2, is essential for effective natural language interaction with computers.

Acknowledgements

This work was partially funded by the National Science Foundation, under grants IRI-9523646 and IRI-9523666 and by a gift from the University of Wisconsin Medical School, Department of Medicine.

References

Ali, S. S. (1994*a*) A Logical Language for Natural Language Processing. *Proceedings of the 10th Biennial Canadian Artificial Intelligence Conference*, Banff, Alberta, Canada, pp. 187–196.

Ali, S. S. (1994*b*) A 'Natural Logic' for Natural Language Processing and Knowledge Representation. *PhD thesis*, State University of New York at Buffalo.

Carbonell, J. G. (1983) Discourse pragmatics and ellipsis resolution in task-oriented natural language interfaces. *Proceedings of the 21st Annual Meeting of the Association for Computational Linguistics*.

Cole, W. G. (1989) Understanding Bayesian reasoning via graphical displays. *Proceedings of the Annual Meeting of the Special Interest Group on Computer and Human Interaction (*SIGCHI*)*, Austin, TX, pp. 381–386.

Druzdel, M. J. and Henrion, M. (1990) Using scenarios to explain probabilistic inference. *Working Notes of the AAAI-90 Workshop on Explanation*, Boston, MA, pp. 133–141. AAAI.

Druzdel, M. J. (1996) Qualitative verbal explanations in Bayesian belief networks. *Artificial Intelligence and Simulation of Behavior Quarterly*, **94**: 43–54.

Elsaesser, C. (1989) Explanation of probabilistic inferences. In: L. N. Kanal, T. S. Levitt and J. F. Lemmer (eds.), *Uncertainty in Artificial Intelligence 3*, pp. 387–400. Elsevier.

Grice, H. P. (1975) Logic and conversation. In: P. Cole and J. L. Morgan (eds.), *Syntax and Semantics 3: Speech Acts*. Academic Press.

Grosz, B. J. and Sidner, C. L. (1986) Attention, intentions, and the structure of discourse. *Computational Linquistics*, **12**.

Haddawy, P., Jacobson, J. and Kahn Jr., C. E. (1996) An educational tool for high-level interaction with Bayesian networks. *Artificial Intelligence and Medicine* (to appear).

Haller, S. (1996) Planning text about plans interactively. *International Journal of Expert Systems*, 85—112.

Henrion, M. and Druzdel, M. J. (1991) Qualitative propagation and scenario-based approaches to explanation of probabilistic reasoning. In: *Uncertainty in Artificial Intelligence 6*, pp. 17–32. Elsevier.

https://doi.org/10.1017/S1351324997001745 Published online by Cambridge University Press

Kahneman, D., Slovic, P. and Tversky, A. (eds.), *Judgement Under Uncertainty: Heuristics and Biases.* Cambridge University Press.

Kukich, K. (1985) Explanation structures in XSEL. *Proceedings of the Annual Meeting of the Asscociation for Computational Linguistics.*

Madigan, D., Mosurski, K. and Almond, R. G. (1994) Explanation in belief networks. (Manuscript.)

Maida, A. S. and Shapiro, S. C. (1982) Intensional concepts in propositional semantic networks. *Cognitive Science*, **6**(4): 291–330. (Reprinted in R. J. Brachman and H. J. Levesque (eds.), *Readings in Knowledge Representation*, Morgan Kaufmann, 1985, pp. 170–189.)

Mann, W. and Thompson, S. (1986) Rhetorical structure theory: Description and construction of text structures. In: G. Kempen (ed.), *Natural Language Generation*, pp. 279–300. Kluwer Academic.

McRoy, S. W. and Hirst, G. (1995*a*) The repair of speech act misunderstandings by abductive inference. *Computational Linguistics*, **21**(4): 435–478.

McRoy, S. W. (1995*b*) Misunderstanding and the negotiation of meaning. *Knowledge-Based Systems*, **8**(2–3): 126–134.

Norton, S. W. (1988) An explanation mechanism for Bayesian inferencing systems. In: L. N. Kanal, T. S. Levitt and J. F. Lemmer (eds.), *Uncertainty in Artificial Intelligence 2*, pp. 165–174. Elsevier.

Sember, P. and Zukerman, I. (1989) Strategies for generating micro explanations for bayesian belief networks. *Proceedings of the 5th Workshop on Uncertainty in Artificial Intelligence*, Windsor, Ontario, pp. 295–302.

Shapiro, S. C. (1982) Generalized augmented transition network grammars for generation from semantic networks. *American Association of Computational Linguistics*, **8**.

Shapiro, S. C. and The SNePS Implementation Group. (1992) *SNePS-2.1 User's Manual.* Department of Computer Science, SUNY at Buffalo.

Shapiro, S. C. and Rapaport, W. J. (1987) SNePS considered as a fully intensional propositional semantic network. In: N. Cercone and G. McCalla (eds.), *The Knowledge Frontier*, pp. 263–315. Springer–Verlag.

Shapiro, S. C. and Rapaport, W. J. (1992) The SNePS family. *Computers & Mathematics with Applications*, **23**(2–5).

Shapiro, S. C., Rapaport, W. J., Cho, S.-H., Choi, J., Feit, E., Haller, S., Kankiewicz, J. and Kumar, D. (1994) A dictionary of SNePS case frames.

Slotnick, S. A. and Moore, J. D. (1995) Explaining quantitative systems to uninitiated users. *Expert Systems with Applications*, **8**(4): 475–490.

Suermondt, H. J. (1992) Explanation of Bayesian Belief Networks. *PhD thesis*, Department of Computer Science and Medicine, Stanford University.

https://doi.org/10.1017/S1351324997001745 Published online by Cambridge University Press