
User Modeling and User-Adapted Interaction (2005) 15: 5–53
DOI 10.1007/s11257-004-5660-7

A Probabilistic Approach for Argument Interpretation

I. ZUKERMAN and S. GEORGE
School of Computer Science and Software Engineering, Monash University, Clayton, Victoria 3800, Australia. e-mail: {ingrid, sarahg}@csse.monash.edu.au

(Received: 6 November 2003; accepted in revised form 30 October 2004)

Abstract. We describe a probabilistic approach for the interpretation of user arguments, and investigate the incorporation of different models of a user's beliefs and inferences into this mechanism. Our approach is based on the tenet that the interpretation intended by the user is that with the highest posterior probability. This approach is implemented in a computer-based detective game, where the user explores a virtual scenario, and constructs an argument for a suspect's guilt or innocence. Our system receives as input an argument entered through a web interface, and produces an interpretation in terms of its underlying knowledge representation – a Bayesian network. This interpretation may differ from the user's argument in its structure and in its beliefs in the argument propositions. We conducted a synthetic evaluation of the basic interpretation mechanism, and a user-based evaluation which assesses the impact of the different user models. The results of both evaluations were encouraging, with the system generally producing argument interpretations our users found acceptable.

Key words. Bayesian networks, discourse interpretation, probabilistic approach

1. Introduction

Discourse interpretation is a cornerstone of human–computer communication, and is an essential component of any dialogue system. In order to interpret a user's Natural Language (NL) utterances, the concepts referenced by the user's words must be identified, the propositions built using these concepts must be understood, and the relations between these propositions must be determined. Each of these tasks is fraught with uncertainty.

Dialogue systems developed to date typically deal with this uncertainty by restricting the dialogue contributions users are allowed to make. This works well for systems with a specific and restricted functionality, e.g., look-up systems. However, systems with a more open-ended functionality, e.g., tutoring systems, should be able to handle more complex responses presented by users.

The argument interpretation mechanism presented in this paper constitutes a significant step towards achieving this objective. Our mechanism interprets structured arguments presented by users in the context of a web-based argumentation system called BIAS (*Bayesian Interactive Argumentation System*). BIAS is designed

to be a comprehensive argumentation system that will eventually engage in an unrestricted interaction with users. However, currently we are focusing on the argument interpretation process, which is the subject of this paper.

Our system uses Bayesian networks (BNs) (Pearl, 1988) as its knowledge representation and reasoning formalism. The system "translates" user arguments into interpretations, which take the form of Bayesian subnets. These subnets are then used for reasoning about the arguments. Our probabilistic interpretation-selection mechanism evaluates these interpretations in terms of two parameters which are independent of the underlying representation: (1) the similarity between the interpretations and the user's argument, and (2) the simplicity of the interpretations. The main idea behind our mechanism is that the best interpretation of a user's argument is that with the highest posterior probability. That is, given a user's argument *UArg* and a set of n candidate interpretations $\{SysInt_1, \ldots, SysInt_n\}$, we will choose the interpretation with the highest posterior probability.

$$SysIntBest = \text{argmax}_{i=1,\ldots,n} \Pr(SysInt_i | UArg)$$

According to Bayes Rule, we can equivalently perform the following maximization.

$$SysIntBest = \text{argmax}_{i=1,\ldots,n} \{\Pr(SysInt_i) \times \Pr(UArg | SysInt_i)\}$$

where the posterior probability of an interpretation given an argument depends on

- $\Pr(SysInt)$ – the prior probability of the interpretation,
- $\Pr(UArg|SysInt)$ – the probability of the argument given the interpretation (this is the probability that a person who intended *SysInt* would have uttered *UArg*).

This general approach is referred to as the source-channel approach, which is widely used for low-level NL tasks, such as machine translation and speech recognition (Epstein, 1996). In this paper, we apply this approach to a high-level NL task, viz discourse interpretation, and offer a framework for the derivation of the above probabilities. This framework constitutes a significant departure from the normal usage of the source-channel approach. We also investigate the incorporation of two types of user models into this formalism: (1) a simple user model that only records the evidence the user has encountered and (2) a more complex user model that takes into account other features of the evidence, specifically the manner in which the evidence was accessed, how frequently and recently it was accessed, and whether this evidence could remind the user of other things.

We conducted two types of evaluations of our probabilistic formalism. First, we performed a synthetic evaluation of the formalism with the simple user model. This was done by first automatically generating arguments from our domain BN, and automatically generating distortions of these arguments. The system then attempted to re-create the original arguments from the distorted arguments. Our second evaluation was user based. In it we asked users to rate the interpretations

obtained by our system when using the simple user model and when using two versions of the complex model.

In the following section we describe our experimental set up. Next, we outline our knowledge representation formalism, and discuss the process for generating candidate interpretations for an argument. In Section 5 we describe our probabilistic formulation of the discourse interpretation process using the simple user model, followed by the results of the synthetic evaluation of our formalism. In Section 7 we consider the incorporation of the complex user model into this formalism, followed by the results of our evaluation with people. We then discuss related research, and present concluding remarks. The Appendix contains the details of the probabilistic formulation.

2. Experimental Set Up

Our experimental set up follows the set up first described in (Zukerman, 2001), which takes the form of a game where the user and the system are partners in solving a murder mystery. However, there is an important difference between the current system and the previous one. The previous BIAS (BIAS-I) was system-driven, while the current BIAS (BIAS-II) is user-driven. That is, in BIAS-I the system was mainly the 'speaker' with the user operating in a 'listener/critiquer" capacity, while the opposite is true for BIAS-II. Specifically, BIAS-I generated arguments, to which the user was allowed to present only single-sentence rejoinders, which in turn were interpreted by BIAS, and rebutted if necessary. In contrast, BIAS-II interprets complete arguments presented by a user.

This difference between BIAS-I and BIAS-II also leads to a difference in the set up of the game. In BIAS-I both the system and the user could conduct their own investigations, so that the system could generate its own arguments, and the user could present rejoinders. In contrast, in BIAS-II the user is a junior detective who interacts with a desk-bound boss. Thus, in BIAS-II there are three entities: the system, which represents the domain and what happened in it; the boss, who knows only what the user tells him; and the user, who finds out information through investigations of the domain. This is done by navigating through a virtual crime scene, making observations and interviewing witnesses. The user then reports periodically to the boss by presenting successively evolving arguments for the main suspect's guilt or innocence. Further, the user has limited resources, i.e., time and money, which are depleted as the investigation progresses. To win the game, the user must build a cogent argument regarding the guilt or innocence of the main suspect prior to exhausting his/her resources.

The current implementation focuses on the interpretation capabilities of the system, rather than on its dialogue model. We therefore restrict users' interaction with the system to a single round. That is, a user reads an initial 'police report' generated by the system, optionally explores the virtual scenario, and then presents an argument to his/her boss. The system interprets the argument, and presents its

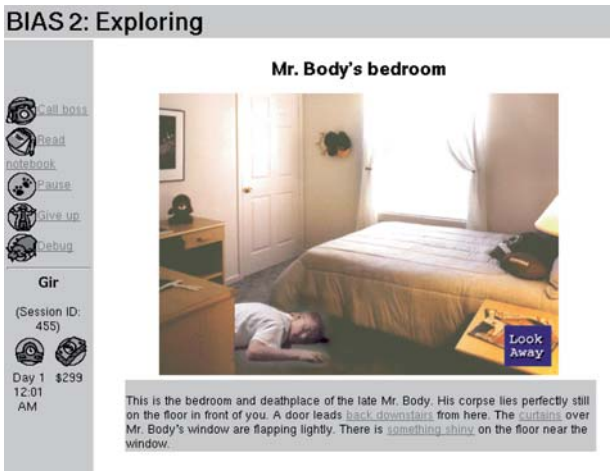


Figure 1. Sample screen of the WWW interface.

interpretation back to the user for validation (this process has been simulated in the pencil-and-paper user-based evaluation described in Section 8). In the future, the boss will present counter-arguments, point out flaws in the user's argument or make suggestions regarding further investigations.

2.1. PLAYING THE GAME – INITIAL INTERACTION

The game starts with the presentation of a police report that describes the preliminaries of the case for a particular scenario. The following police report is presented for the scenario used in this paper.

> *Yesterday, Mr Body was found dead in his bedroom, which is in the second story of his house. Fatal bullet wounds were found in Mr Body's body. A gun was found in the garden, and fingerprints were found on the gun. Forensics established that the time of death was 11 pm.*

After reading the police report, the user may navigate through a virtual scenario to gather additional information (Figure 1 shows a screen shot of the victim's bedroom). The user may record information s/he considers interesting in his/her *Notebook* (Figure 2), which is consulted by the user during the argument construction process. Upon completion of his/her investigation, the user builds an argument composed of a sequence of implications leading from evidence to the argument goal. Each implication is composed of one or more antecedents and a consequent. In the current implementation, the antecedents and consequents are obtained by copying propositions from a menu into slots in the argument-construction interface (Figure 3).[1] Figure 4 shows a screen-shot of an argument built

[1]An alternative version of our system accepts free-form NL input for antecedents and consequents. However, in our current version this capability has been replaced with a menu-based capability.

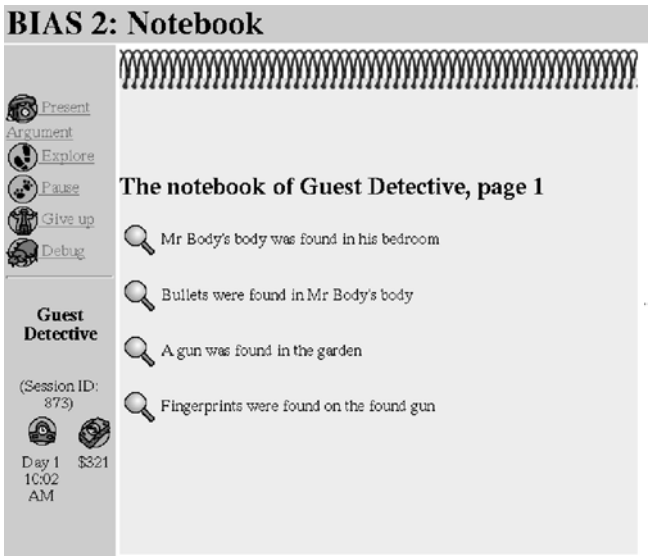


Figure 2. Detective's notebook.

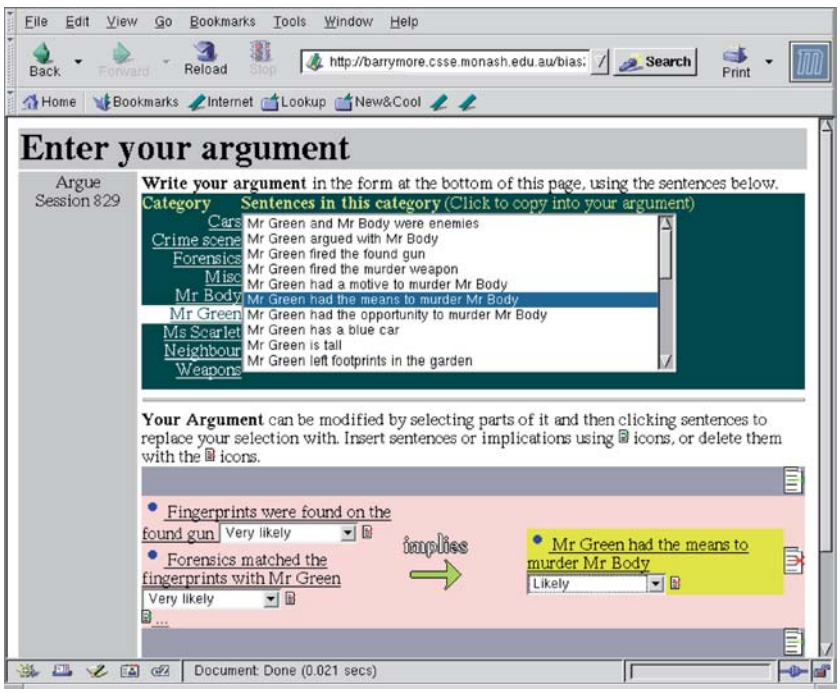


Figure 3. Menu for argument construction.

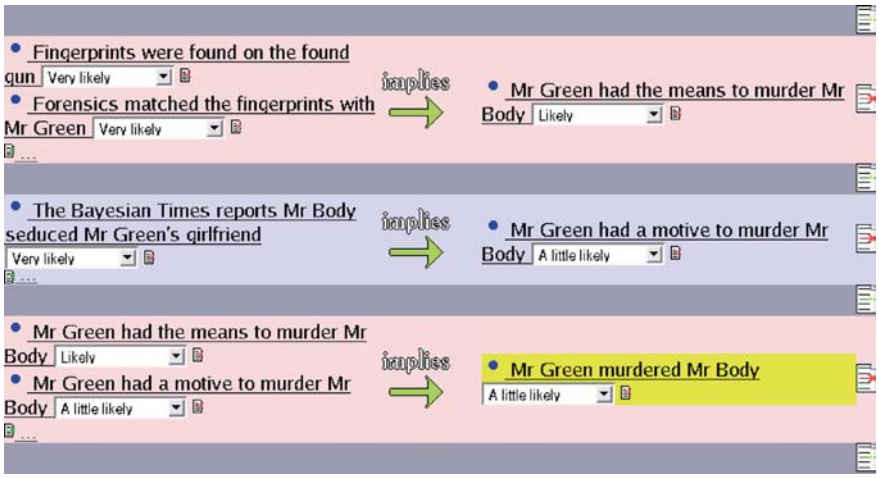


Figure 4. Argument-construction screen and user's argument.

by a particular user after she has read the police report, seen the newspaper and spoken to the forensic experts. This argument may be glossed as follows.

> *Fingerprints being found on the gun, and forensics matching the fingerprints with Mr Green implies that it is likely that Mr Green had the means to murder Mr Body.*
> *The Bayesian Times reporting that Mr Body seduced Mr Green's girlfriend implies that it is a little likely that Mr Green had a motive to murder Mr Body.*
> *Since it is likely that Mr Green had the means to murder Mr Body, and it is a little likely that Mr Green had a motive to murder Mr Body, then it is a little likely that Mr Green murdered Mr Body.*

To illustrate the argument construction process, let us consider the first implication in Figure 4, which is built as follows. The user clicks the "Forensics" sub-menu and finds the statement about fingerprints being found on the gun, which she copies to the implication, assigning it a belief of VeryLikely (Figure 3).[2] The user also finds in the "Forensics" sub-menu the statement regarding the match between the fingerprints on the gun and Mr Green's, to which she also assigns a belief of VeryLikely. The consequent about Mr Green having the means to murder Mr Body can be found in the "Mr Green" sub-menu, and it receives a belief of Likely.[3]

Figure 5 shows the interpretation generated by BIAS for the argument in Figure 4 (we have boxed the propositions in the original argument, and drawn arrows to indicate the implications mentioned in the original argument). In it the system points out its beliefs and the user's, and fills in propositions and relations where

[2]The user can select a belief from the following categories: VeryLikely, Likely, ALittleLikely, EvenChance, ALittleUnlikely, Unlikely and VeryUnlikely.

[3]In the current interface, users are restricted to presenting simple implications, such as those shown in Figure 4. This interface supports the construction of arguments, in the sense that it allows users to express how the antecedents of an implication influence their consequent. However, at present our interface does not support the expression of more sophisticated relations, such as exceptions and contradictions.

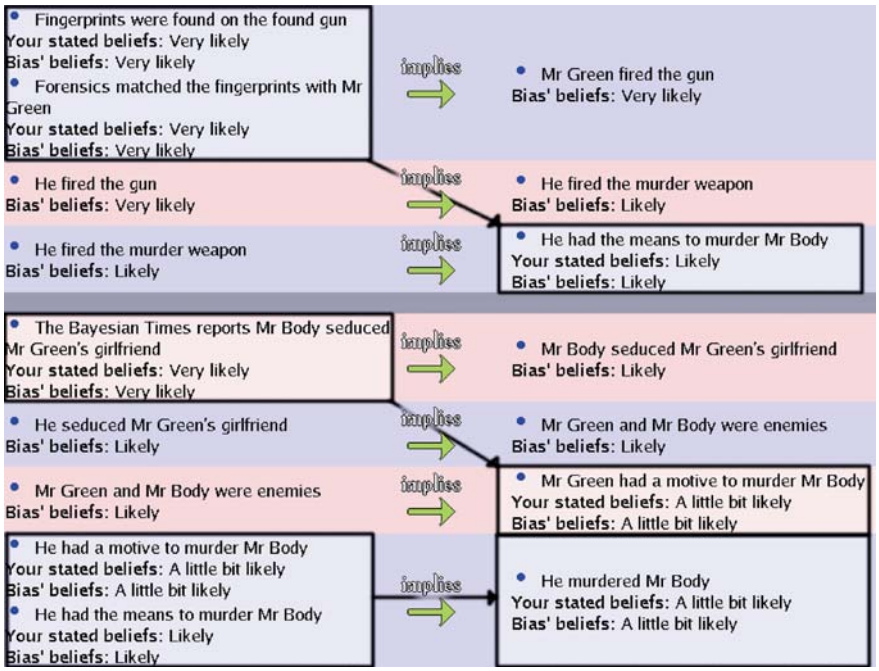


Figure 5. BIAS' interpretation of the user's argument.

the user has made inferential leaps. These propositions and relations correspond to nodes and arcs which are required to connect the user's propositions in BIAS' domain BN. These nodes and arcs are inferred as described in Sections 4 and 5.

3. Domain Representation

The domain propositions and the relationships between them are represented by means of a BN (Pearl, 1988). Each BN in the system can support a variety of scenarios, depending on the instantiation of the evidence nodes. The murder mystery used for this paper is represented by means of a 32-node BN, which is a less detailed version of the 85-node BN used in (Zukerman, 2001; Zukerman et al., 2003a).[4]

Figure 6 shows our 32-node BN: the observable evidence nodes are boxed, and the goal node [GreenMurderedBody] is circled. The four evidence nodes mentioned in the police report are boldfaced and shaded ([GunFoundInGarden], [FingerprintsFoundOnGun], [BulletsFoundInBody'sBody] and [TimeOfDeath11]), and the three evidence nodes obtained by the user in her investigation are white boldfaced and dark shaded ([ForensicMatchGreen'sFingerprints], [BayesTimesReportBodySeduceGreen'sGirlfriend] and [NbourHeardGreenBodyArgLastNight]).

[4]Both the 32-node BN and the 85-node BN were hand-built. Their plausibility was assessed by performing Bayesian propagation under a variety of evidence conditions, and also during trials with users. The additional detail in the larger BN adds interest to the game. However, it obscures the issues being investigated here, which are better highlighted by the smaller BN.

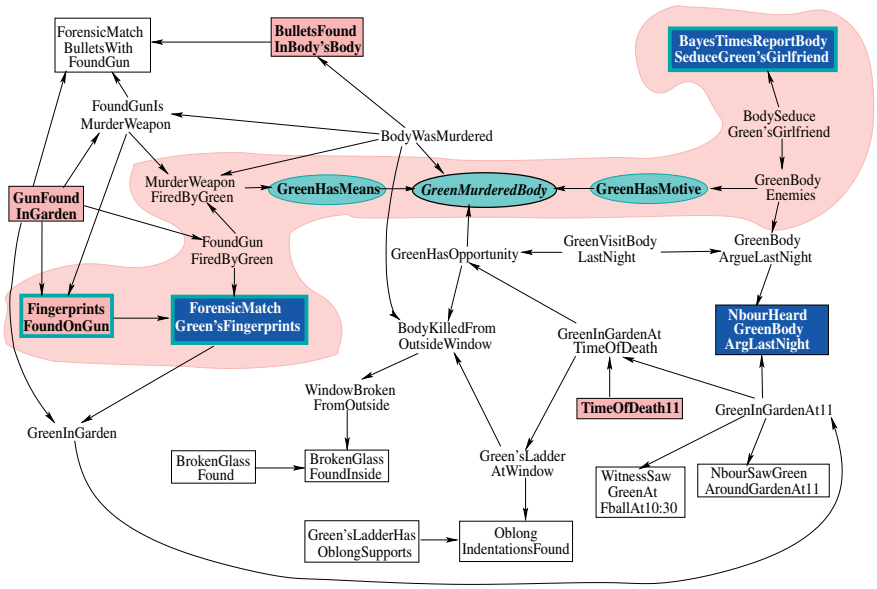


Figure 6. Domain BN and interpretation of the user's argument.

In general, when building an argument, the user does not necessarily employ all the evidence nodes s/he has encountered. For instance, the argument in Figure 4 uses one evidence node from the police report ([FingerprintsFoundOnGun]), and two evidence nodes from the three encountered by the user in her investigation ([BayesTimesReportBodySeduceGreen'sGirlfriend] and [ForensicMatchGreen'sFingerprints]). These nodes have a thick frame in Figure 6. The nodes corresponding to the consequents in the user's argument are boldfaced and shaded ([GreenHasMeans], [GreenHasMotive] and [GreenMurderedBody]), and the gray bubble indicates the interpretation preferred by BIAS.

4. Proposing Interpretations

Our system generates candidate interpretations for a user's argument by finding different ways to connect the propositions in the argument – each variant being a candidate interpretation. The posterior probability of each interpretation is then calculated using the formalism described in Section 5, and the interpretation with the highest probability is selected.

In order to interact with users in real time, we use an *anytime* algorithm (Dean and Boddy, 1988) that generates increasingly complex interpretations as time progresses. This algorithm, called *GenerateInterpretations*, produces a stream of interpretations until it is interrupted, i.e., when time runs out (for our current trials, the time limit is set to 5 seconds).

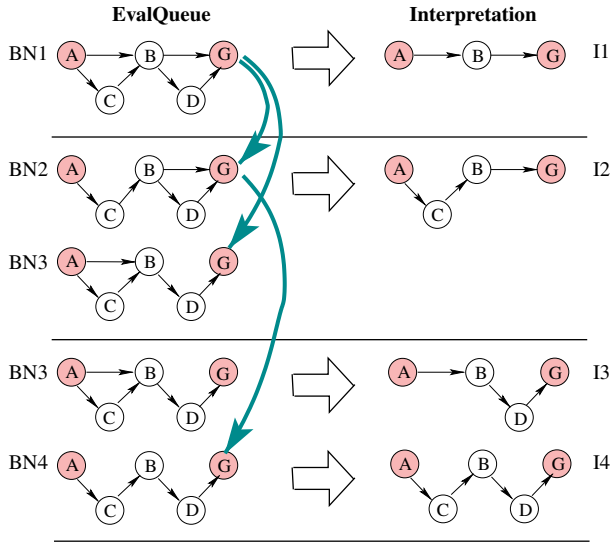


Figure 7. Generating interpretations for an argument.

Algorithm *GenerateInterpretations*

1. *EvalQueue* ← domain BN; interpretation counter $i \leftarrow 1$.
2. While there is time
   (a) Remove the first BN from *EvalQueue*, call it *curBN*.
   (b) Find $Interpretation_i$, the minimum spanning tree in *curBN* that connects the nodes in the user's argument.[5]
   (c) Make copies of *curBN* such that a different arc from $Interpretation_i$ is removed from each copy (if $Interpretation_i$ has N arcs, N copies are generated – one for each removed arc).
   (d) Append the copies of *curBN* to *EvalQueue*.
   (e) Increment i.

Figure 7 illustrates the application of this algorithm to generate interpretations for a simple argument comprising two nodes $A \Rightarrow G$ in the context of a domain BN comprising 5 nodes (the nodes mentioned in the argument are shaded). First BIAS generates the most concise interpretation I1, which consists of $A \rightarrow B \rightarrow G$, from the complete domain BN (BN1 in Figure 7). The removal of the arc between A and B yields the copy of this BN labelled BN2, and the removal of the arc between B and G yields BN3. After producing interpretation I2 from BN2, three additional copies are generated: one without the arc between A and C, one without the arc between C and B, and one without the arc between B and G. The first two copies are not viable, as the argument propositions are not fully connected. Thus, only

[5]A minimum spanning tree in this context is a tree that connects the nodes in *curBN* and has the smallest number of arcs. In the future BIAS will also generate graphs.

the last copy (BN4) is retained and appended to *EvalQueue*. When BN3 moves up in the queue, another interpretation (I3) is generated. However, no new BN copies are produced (as for BN2, the first two copies are not viable; the third copy has been generated from BN2).

The interpretations generated by this process are structurally subnets of the domain BN, but not probabilistically. That is, the *Conditional Probability Tables (CPTs)* of the nodes in these subnets may include influences from parent nodes that do not appear in the subnets. For instance, nodes *B* and *G* in interpretation I1 in Figure 7 have only one parent node, while in the original BN (BN1) they have two parent nodes. In order to turn interpretation I1 into a legal Bayesian subnet, the CPT for node *B* must be adjusted to include only node *A* as a parent, and the CPT for node *G* must be adjusted to include only node *B*. This adjustment is performed by a process called *marginalization*, which retains the influence of the removed nodes (Pearl, 1988).

In general, to generate a subnet from a BN, we must (1) propagate belief through the BN, (2) marginalize the parent nodes that are not in the subnet, (3) remove the child nodes that are not in the subnet, and (4) re-propagate belief in the subnet. Marginalization is a process that calculates a weighted average of the influence of all the parent nodes of a node, where the weights are the propagated probabilities of the parent nodes to be removed, thereby retaining their influence. In contrast, according to BN theory, child nodes to be removed are simply ignored (without considering their influence after the initial belief propagation).[6] Finally, belief is re-propagated through the resulting Bayesian subnet, so that the influence of its nodes can be determined.

5. Selecting an Interpretation

In this section we present our formulation for calculating the posterior probability of argument interpretations, and apply this formulation to select the best (intended) interpretation for an argument from the interpretations obtained as described in Section 4.

5.1. PROBABILISTIC FORMULATION OF THE INTERPRETATION PROCESS

The probability that a user who presented an argument *UArg* intended an interpretation *SysInt* is $\Pr(SysInt|UArg)$, the posterior probability of *SysInt* given *UArg*. As stated in Section 1, *SysIntBest*, the interpretation with the highest posterior probability, is that which satisfies the following equation:

$$SysIntBest = \text{argmax}_{i=1,\ldots,n} \Pr(SysInt_i|UArg) \quad (1)$$

where n is the number of interpretations.

[6]The removal of child nodes followed by belief re-propagation sometimes yields nodes with unintuitive beliefs (Figure 13).

The application of Bayes Theorem to Equation 1 yields

$$SysIntBest = \text{argmax}_{i=1,\ldots,n} \frac{\Pr(SysInt_i) \times \Pr(UArg|SysInt_i)}{\Pr(UArg)}$$

Since the denominator is constant (as all the candidate interpretations are generated for the same argument *UArg*) we obtain[7]

$$SysIntBest = \text{argmax}_{i=1,\ldots,n}\{\Pr(SysInt_i) \times \Pr(UArg|SysInt_i)\} \quad (2)$$

According to this formulation, the posterior probability of an interpretation given an argument depends on Pr(*SysInt*) – the prior probability of the interpretation, and Pr(*UArg*|*SysInt*) – the probability of the argument given the interpretation (this is the probability that a person who intended *SysInt* would have uttered *UArg*).

Now, it is reasonable to say that the probability of an argument given an interpretation depends on how easily the argument can be derived from the interpretation (easier derivations being more probable).[8] However, the factors that affect the prior probability of an interpretation may be more open to debate. For our basic formalism, we propose to use simplicity as the factor that determines the prior probability of an interpretation (simpler interpretations being more probable).[9] In Section 7, we consider the impact of different user models on this probability.

In light of these considerations, Equation 2 gives a probabilistic formulation of Occam's Razor, which may be stated as follows: "If you have two theories both of which explain the observed facts, then you should use the simplest until more evidence comes along". If we view a user's argument as a set of "observations" (these observations are the statements in the user's argument, which may include data, warrants and inferences), and the candidate interpretations of this argument as competing theories that explain these observations, then according to Occam's Razor, the preferred interpretation (i.e., the most probable) is the simplest that matches the argument well. Hopefully, this is also the intended interpretation. It is important to note that according to Occam's Razor (and our probabilistic formulation), the simplest interpretation of an argument is not necessarily the best, as it may have a poor fit with the user's argument; the best and hopefully intended interpretation is one that balances simplicity with argument (data) fit.

Now, the questions in front of us are: (1) how to calculate the probability of an interpretation based on its simplicity, and (2) how to calculate the conditional probability of an argument given an interpretation based on the similarity between the argument and the interpretation. These calculations appear in Appendices A

[7]This technique compares interpretations in order to select the most probable one. Hence, in principle, it does not need to calculate the precise probabilities of the interpretations. Our software calculates these probabilities, since sometimes the best interpretation is still not plausible. However, in the paper we have omitted these detailed calculations for clarity of presentation.

[8]Note that the calculation of Pr(*UArg*|*SysInt*) is not symmetric to the direct calculation of Pr(*SysInt*|*UArg*), because the propositions in *UArg* are a subset of those in *SysInt*.

[9]The consideration of other factors does not affect the basic tenets of our approach.

and B respectively. The sections below summarize the results of these calculations and illustrate them with examples.

5.2. ESTIMATING THE PROBABILITY OF AN INTERPRETATION

The technique for calculating the probability of an interpretation *SysInt* is based on the idea that we need to select the nodes and arcs in the interpretation from those in the domain BN.

We identify an interpretation by specifying the number of nodes in it, the number of arcs, and the actual nodes and arcs in it. Thus, the probability of an interpretation *SysInt*, Pr(*SysInt*), in the context of the domain BN is defined as

$$\Pr(SysInt) = \Pr(Arcs_{SI}, Nodes_{SI}, a_{SI}, n_{SI}|dBN) \tag{3}$$

where dBN is the domain BN (composed of A arcs and N nodes), $Arcs_{SI}$ is the set of arcs in the interpretation *SysInt*, $Nodes_{SI}$ the set of nodes in *SysInt*, a_{SI} the number of arcs in *SysInt*, i.e., $|Arcs_{SI}|$, and n_{SI} the number of nodes in *SysInt*, i.e., $|Nodes_{SI}|$.

Applying the chain rule of probability theory yields

$$\begin{aligned}\Pr(SysInt) = {} & \Pr(Arcs_{SI}|Nodes_{SI}, a_{SI}, n_{SI}, dBN) \times \Pr(a_{SI}|Nodes_{SI}, n_{SI}, dBN) \\ & \times \Pr(Nodes_{SI}|n_{SI}, dBN) \times \Pr(n_{SI}|dBN)\end{aligned} \tag{4}$$

Here we summarize the calculation of these probabilities. Their precise calculation appears in Appendix A.

- $\Pr(n_{SI}|dBN)$ is the probability of having n_{SI} nodes in an interpretation. We model this probability by means of a truncated Poisson distribution, $Poisson(\beta)$, where β is the average number of nodes in an interpretation (Equation A.3).
- $\Pr(Nodes_{SI}|n_{SI}, dBN)$ is the probability of selecting the particular n_{SI} nodes in $Nodes_{SI}$ from the N nodes in dBN. Assuming that all nodes have an equal probability of being selected, there are $\binom{N}{n_{SI}}$ ways to select these nodes (Equation A.4).
- $\Pr(a_{SI}|Nodes_{SI}, n_{SI}, dBN)$ is the probability of having a_{SI} arcs in an interpretation. The number of arcs in an interpretation is between the minimum number of arcs needed to connect n_{SI} nodes $(n_{SI} - 1)$, and the actual number of arcs in dBN that connect the nodes in $Nodes_{SI}$, denoted va_{SI}. Hence, we model the probability of a_{SI} by means of a uniform distribution between $n_{SI} - 1$ and va_{SI} (Equation A.5).
- $\Pr(Arcs_{SI}|Nodes_{SI}, a_{SI}, n_{SI}, dBN)$ is the probability of selecting the particular a_{SI} arcs in $Arcs_{SI}$ from the va_{SI} arcs in dBN that connect the nodes in *SysInt*. There are $\binom{va_{SI}}{a_{SI}}$ ways to select these arcs (Equation A.6).

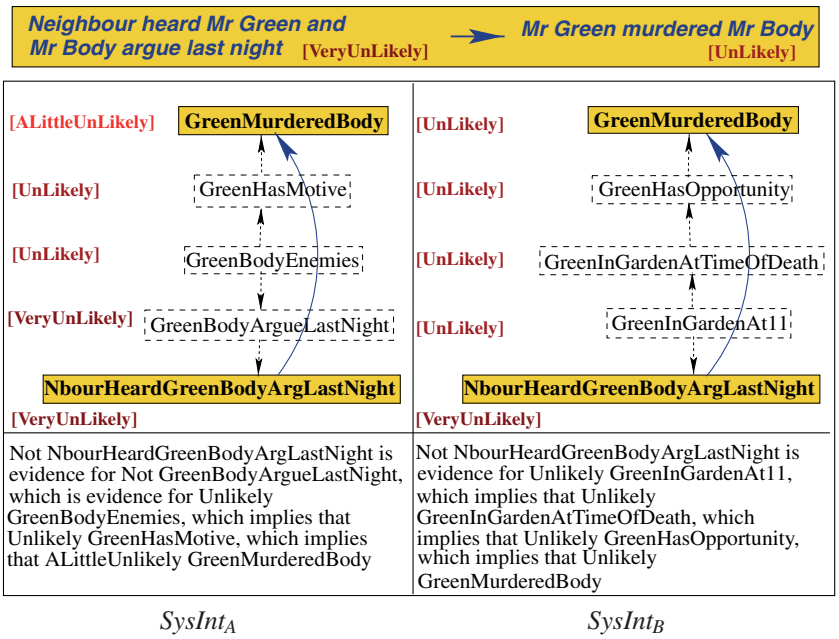


Figure 8. Simple argument and interpretations.

Example

To illustrate the results obtained from Equation 4 consider our domain BN of 32 nodes (Figure 6), and the simple argument and interpretations in Figure 8 (these interpretations may be traced starting from node [NbourHeardGreenBodyArgLastNight] on the right-hand-side of Figure 6). The nodes mentioned by the user appear in bold and are dark shaded, and the beliefs in the user's argument and those obtained by BIAS in the interpretations appear in brackets. The arcs indicate the flow of evidence in the domain BN. The gloss for each interpretation appears under it.

Both interpretations have the same prior probability due to the following reasons:

- Both interpretations have 5 nodes ($n_A = n_B = 5$). Hence, $\Pr(n_A|\text{BN})=\Pr(n_B|\text{BN})$. Further, since the nodes in the BN are equiprobable, $\Pr(Nodes_A|n_A, \text{BN}) = \Pr(Nodes_B|n_B, \text{BN})$.
- The number of arcs in the BN in Figure 6 that connect between the nodes in each interpretation (va_A and va_B) is the minimum number of arcs ($va_A = a_A = n_A - 1 = 4$ and $va_B = a_B = n_B - 1 = 4$). Hence, there is only one possible number of arcs for each interpretation (4), and there is only one way to choose a_A arcs from va_A arcs, and a_B arcs from va_B arcs. That is, $\Pr(a_A|Nodes_A, n_A, \text{BN}) = \Pr(a_B|Nodes_B, n_B, \text{BN}) = 1$, and $\Pr(Arcs_A|Nodes_A, a_A, n_A, \text{BN}) = \Pr(Arcs_B|Nodes_B, a_B, n_B, \text{BN}) = 1$.

Hence, at this stage we have no grounds for preferring one interpretation over the other (this example is continued in Section 5.3, where data fit considerations are applied).

5.3. ESTIMATING THE PROBABILITY OF THE ARGUMENT GIVEN AN INTERPRETATION

A user's argument is a sequence of implications of the form:

$$Antecedent_1\ Antecedent_2 \ldots Antecedent_n \Rightarrow Consequent$$

where $\Rightarrow$ indicates that the antecedents imply the consequent, without distinguishing between causal and evidential implications.

A user's argument, denoted *UArg*, may be represented as a graph where the antecedents and consequents correspond to nodes in our BN. Pr(*UArg*|*SysInt*) is the probability that a user uttered *UArg* when s/he intended *SysInt*. This probability depends on the similarity between *UArg* and *SysInt*, i.e., the more similar *UArg* is to an interpretation, the more probable it is that the user presented this argument when s/he intended this interpretation. This similarity depends on the structural similarity between *SysInt* and *UArg* (where structure is represented in terms of nodes and arcs), and on the closeness between the beliefs in the nodes in *UArg* and the beliefs in the corresponding nodes in *SysInt* (these beliefs are obtained by performing Bayesian propagation through *SysInt*; hence, different *SysInt*s may yield different beliefs in the consequents of an argument). Thus,

$$\Pr(UArg|SysInt) = \Pr(Struct(UArg), Bel(UArg)|Struct(SysInt), Bel(SysInt)) \quad (5)$$

where, *Struct(UArg)* denotes the structure of the user's argument *UArg*, *Struct(SysInt)* denotes the structure of interpretation *SysInt*, *Bel(UArg)* denotes the beliefs expressed in *UArg*, and *Bel(SysInt)* denotes the beliefs obtained in *SysInt*.

By applying the chain rule of probability theory, and making some simplifying assumptions (Appendix B), we obtain the following formula, which considers separately structure and belief.

$$\begin{aligned}\Pr(UArg|SysInt) = {} & \Pr(Struct(UArg)|Struct(SysInt)) \\ & \times \Pr(Bel(UArg)|Bel(SysInt))\end{aligned} \quad (6)$$

Calculating Pr(*Bel(UArg)*|*Bel(SysInt)*)

The calculation of Pr(*Bel*(*UArg*)|*Bel*(*SysInt*)) is based on the similarity between the beliefs stated by the user in *UArg* and those obtained in *SysInt* as a result of Bayesian propagation. This propagation relies on the evidence in the simple user model, which contains the information encountered by the user in the police report and in his/her exploration of the virtual scenario.

Bel(UArg) comprises the beliefs stated by the user with respect to the nodes in *UArg*. That is, $Bel(UArg) = \{Bel(Nd_1, UArg), \ldots, Bel(Nd_{n_{UA}}, UArg)\}$, where n_{UA} is the number of nodes in *UArg* ($n_{UA} \leq n_{SI}$, which is the number of nodes in

SysInt). Similarly, *Bel(SysInt)* comprises the beliefs in nodes $Nd_1, \ldots, Nd_{n_{UA}}$ in *SysInt*, which are obtained by means of Bayesian propagation (nodes that appear only in *SysInt* are handled by the component which describes structural differences, presented below). Thus,

$$\begin{aligned}&\Pr(Bel(UArg)|Bel(SysInt))\\&\quad = \Pr(Bel(Nd_1, UArg), \ldots, Bel(Nd_{n_{UA}}, UArg)|Bel(Nd_1, SysInt), \ldots,\\&\qquad Bel(Nd_{n_{UA}}, SysInt))\end{aligned}$$

By applying the chain rule of probability theory, and making some simplifying assumptions (Appendix B.1), we obtain the following formula, which considers each node in the user's argument separately:

$$\Pr(Bel(UArg)|Bel(SysInt)) = \prod_{i=1}^{n_{UA}} \Pr(Bel(Nd_i, UArg)|Bel(Nd_i, SysInt)) \tag{7}$$

In order to calculate $\Pr(Bel(Nd_i, UArg)|Bel(Nd_i, SysInt))$, we need to perform the following tasks:

- Consider belief categories (from VeryUnlikely to VeryLikely as per our interface, Section 2.1), rather than point probabilities (Equation B.4).
- Devise a probabilistic measure that rewards similarities between the beliefs in a user's argument and the corresponding beliefs in an interpretation, and penalizes discrepancies between these beliefs. We have selected the Zipf distribution for this task (Equation B.5).

Calculating Pr(*Struct(UArg)|Struct(SysInt)*)

The calculation of $\Pr(Struct(UArg)|Struct(SysInt))$ is based on the idea that we need to select the nodes and arcs in *UArg* from those in *SysInt*. This idea is similar to that used to calculate $\Pr(SysInt)$ (where we selected the nodes and arcs in *SysInt* from those in *dBN*). However, in this case there is a complicating factor, since the user could mention implications (arcs) which do not exist in *SysInt*. Hence, the calculation of $\Pr(Struct(UArg)|Struct(SysInt))$ resembles the calculation of $\Pr(SysInt)$ in Equation 3, but distinguishes between arcs in *UArg* that are selected from *SysInt* and arcs that are newly inserted. These arcs are designated as follows: $Arcs_{\text{sel}}$ is the set of arcs in *UArg* selected from *SysInt*, $Arcs_{\text{ins}}$ is the set of newly inserted arcs in *UArg* (i.e., arcs that cannot be obtained from *SysInt*), a_{sel} is the number of selected arcs, i.e., $|Arcs_{\text{sel}}|$, and a_{ins} is the number of inserted arcs, i.e., $|Arcs_{\text{ins}}|$.

This results in the following definition for $\Pr(Struct(UArg)|Struct(SysInt))$:

$$\begin{aligned}&\Pr(Struct(UArg)|Struct(SysInt))\\&\quad = \Pr(Arcs_{\text{sel}}, Arcs_{\text{ins}}, a_{\text{sel}}, a_{\text{ins}}, Nodes_{UA}, n_{UA}|Struct(SysInt))\end{aligned}$$

where $Nodes_{UA}$ designates the nodes in *UArg*, and n_{UA} is the number of nodes in *UArg*.

By applying the chain rule of probability theory, and making some simplifying assumptions (Appendix B.2), we obtain the following formula:

$$\begin{aligned}
&\Pr(Struct(UArg)|Struct(SysInt)) \\
&\quad = \Pr(Arcs_{\text{sel}}|a_{\text{sel}}, Nodes_{UA}, n_{UA}, SysInt) \times \Pr(a_{\text{sel}}|Nodes_{UA}, n_{UA}, SysInt) \\
&\qquad \times \Pr(Arcs_{\text{ins}}|a_{\text{ins}}, Nodes_{UA}, n_{UA}, SysInt) \times \Pr(a_{\text{ins}}|Nodes_{UA}, n_{UA}, SysInt) \\
&\qquad \times \Pr(Nodes_{UA}|n_{UA}, SysInt) \times \Pr(n_{UA}|SysInt) \qquad (8)
\end{aligned}$$

Here we summarize the calculation of these probabilities. Their precise calculation appears in Appendix B.2.

- $\Pr(n_{UA}|SysInt)$ is the probability of having n_{UA} nodes in an argument. We model this probability by means of a truncated Poisson distribution, $Poisson(\rho)$, where ρ is the average number of nodes in a user's argument (Equation B.9).
- $\Pr(Nodes_{UA}|n_{UA}, SysInt)$ is the probability of selecting the particular n_{UA} nodes in $Nodes_{UA}$ from the n_{SI} nodes in *SysInt*. We model this probability as we did for *SysInt* (Section 5.2). However, we take into account the fact that owing to the way in which interpretations are derived by procedure *GenerateInterpretations* (Section 4), all the leaf nodes in *SysInt* must appear in *UArg*, and only non-leaf nodes in *SysInt* are uncertain of appearing in *UArg*. This observation reduces the number of ways in which the n_{UA} nodes in *UArg* may be selected from the n_{SI} nodes in *SysInt*, and thereby yields more accurate (and higher) probabilities for the node configurations in *UArg* (Equation B.10).
- $\Pr(a_{\text{sel}}|Nodes_{UA}, n_{UA}, SysInt)$ is the probability of selecting from *SysInt* a_{sel} arcs that connect the nodes in $Nodes_{UA}$. As for *SysInt*, we model this probability using a uniform distribution. However, we take into account the fact that *UArg* may contain inferences that skip some non-leaf nodes in *SysInt*. For instance, a user may have said $A \rightarrow C$, while the corresponding structure in the domain BN is $A \rightarrow B \rightarrow C$. Such cases are considered algorithmically prior to calculating the probability of a_{sel} (Equation B.11). This is done by redirecting arcs in *SysInt* so that they connect between the children and the parents of *SysInt* nodes that were omitted from the user's argument (e.g., in our example, the arcs around B are redirected, so that A is connected directly to C). As above, this observation increases the accuracy of the calculated probabilities.
- $\Pr(a_{\text{ins}}|Nodes_{UA}, n_{UA}, SysInt)$ is the probability of inserting a_{ins} arcs in *SysInt*. We model this probability using a truncated Poisson distribution, $\text{Poisson}(\mu)$, where μ is the mean number of inserted arcs (Equation B.12).
- $\Pr(Arcs_{\text{sel}}|a_{\text{sel}}, Nodes_{UA}, n_{UA}, SysInt)$ is the probability of selecting the particular a_{sel} arcs in $Arcs_{\text{sel}}$ from the arcs in *SysInt* that connect the nodes in $Nodes_{UA}$. This probability is calculated as for *SysInt* (Section 5.2), but $Arcs_{sel}$ is chosen from the arcs redirected as explained above, rather than from the original arcs in *SysInt* (Equation B.13).

Table I. Observations made before the simple argument in Figure 8.

Proposition	Value and source
Bullet wounds were found in Mr Body's body	T (police rep.)
A gun was found in the garden	T (police rep.)
Fingerprints were found on the gun	T (police rep.)
Forensics established that the time of death was 11 pm	T (police rep.)
Neighbour heard Mr Green and Mr Body argue last night	**F** (exploration)
A witness saw Mr Green at the football at 10:30	F (exploration)
Bayesian Times reported that Mr Body seduced Mr Green's girlfriend	F (exploration)
Broken glass was found inside Mr Body's window	T (exploration)

– $\Pr(Arcs_{\text{ins}}|a_{\text{ins}}, Nodes_{UA}, n_{UA}, SysInt)$ is the probability of selecting a_{ins} arcs from the maximum possible number of arc insertions in *SysInt* (Equation B.14).

Example

In this example we illustrate the influence of beliefs on the results obtained from Equation 6 (in Section 7 we consider a more complex example that balances the probabilities obtained from structure and belief).

Let us reconsider the simple argument and interpretations in Figure 8. Table I shows the evidence in the simple user model (the evidence in the antecedent of the user's argument is boldfaced). Note that even though most of the evidence in Table I is omitted from the interpretations, it may still affect the beliefs in the propositions in the interpretations due to the marginalization process (Section 4).

$SysInt_A$ and $SysInt_B$ are structurally equivalent due to the following reasons:

– The user's argument has two nodes ($n_{UA} = 2$), and both interpretations have five nodes ($n_{SI} = 5$), yielding $\Pr(n_{UA}|SysInt_A) = \Pr(n_{UA}|SysInt_B)$, and $\Pr(Nodes_{UA}|n_{UA}, SysInt_A) = \Pr(Nodes_{UA}|n_{UA}, SysInt_B)$.
– The argument has only one arc, which is selected from the redirected arcs in $SysInt_A$ and $SysInt_B$ ($a_{\text{sel}} = 1$). Since the arc-redirection process yields one arc (that in *UArg*) for both interpretations, we obtain $\Pr(a_{\text{sel}}|Nodes_{UA}, n_{UA}, SysInt) = 1$, and $\Pr(Arcs_{\text{sel}}|a_{\text{sel}}, Nodes_{UA}, n_{UA}, SysInt) = 1$ for both $SysInt_A$ and $SysInt_B$.
– No extraneous arc insertions are required ($a_{\text{ins}} = 0$) for both interpretations.[10] Hence, $\Pr(a_{\text{ins}}|Nodes_{UA}, n_{UA}, SysInt_A) = \Pr(a_{\text{ins}}|Nodes_{UA}, n_{UA}, SysInt_B)$, and $\Pr(\text{Arcs}_{\text{ins}}|a_{\text{ins}}, Nodes_{UA}, n_{UA}, SysInt_A) = \Pr(\text{Arcs}_{\text{ins}}|a_{\text{ins}}, Nodes_{UA}, n_{UA}, SysInt_B)$.

[10] For clarity of exposition, the examples presented in this paper have a linear argumentation structure. As a result, the arc redirection and arc insertion components are simplified out of the formulas. While this occurs often, it cannot be assumed to be generally the case. For instance, the user may have said $A \rightarrow B \rightarrow C$, when in the actual BN A and B are children of C, and there is no arc between A and B. Such cases are accounted for by the arc insertions in our formulation.

Let us now consider the beliefs in the user's argument. For both interpretations, the probability of the beliefs is calculated as follows:

$$\begin{aligned}
&\Pr(Bel(UArg)|Bel(SysInt)) \\
&= \Pr(Bel(\text{NbourHeardGreenBodyArgLastNight}, UArg)| \\
&\qquad Bel(\text{NbourHeardGreenBodyArgLastNight}, SysInt)) \\
&\quad \times \Pr(Bel(\text{GreenMurderedBody}, UArg)|Bel(\text{GreenMurderedBody}, SysInt))
\end{aligned}$$

Since $SysInt_B$ matches both beliefs in $UArg$, $\Pr(Bel(UArg)|Bel(SysInt_B))=1$. In contrast, $SysInt_A$ differs from $UArg$ in the belief in node [GreenMurderedBody] (the user postulates a belief of Unlikely, while the system infers ALittleUnlikely). This discrepancy is penalized by assigning a lower probability (=0.25) to the factor corresponding to this belief for $SysInt_A$ (this probability is obtained using Equation B.5).

Thus, $\Pr(UArg|SysInt_B)$ is 4 times more probable than $\Pr(UArg|SysInt_A)$. Now, recall that in Section 5.2 we obtained the same prior probabilities for $SysInt_A$ and $SysInt_B$. In order to combine these two results, we return to Equations 1 and 2, and express the posterior probability for $SysInt_A$ in terms of the probabilities for $SysInt_B$. This yields

$$\begin{aligned}
\Pr(SysInt_A|UArg) &= \Pr(SysInt_A) \times \Pr(UArg|SysInt_A) \\
&= \Pr(SysInt_B) \times 0.25 \Pr(UArg|SysInt_B), \\
\Pr(SysInt_A|UArg) &= 0.25 \Pr(SysInt_B|UArg) \qquad (9)
\end{aligned}$$

That is, given $UArg$, $SysInt_B$ is 4 times more probable than $SysInt_A$.

Now, one may argue that such a small discrepancy in belief should not tip the balance in favour of a particular interpretation to such a large extent. However, we impose such a heavy penalty for discrepancies in belief as a result of our preliminary trials with users, where our trial subjects objected strongly to small differences in belief between their arguments and BIAS' interpretations (Appendix B.1). More importantly, our formalism represents explicitly the effect of the components of an interpretation on the overall probability of the interpretation, where the effect of each component is determined by the parameters of the distribution used to calculate its probability. Some of these parameters may be adjusted based on the importance ascribed to the influence of a component, as was done for belief.

6. Synthetic Evaluation

Our synthetic evaluation consisted of an automated experiment where the system interprets noisy (distorted) versions of its own arguments. These arguments were generated from different subnets of its domain BN, and they were distorted by changing the beliefs in the nodes, and inserting and deleting arcs and nodes. The distortions were performed as follows. Beliefs were distorted by assigning to each node a belief that is within 10%, 20%, 30% or 40% of the belief in this node in the original argument. Node deletions

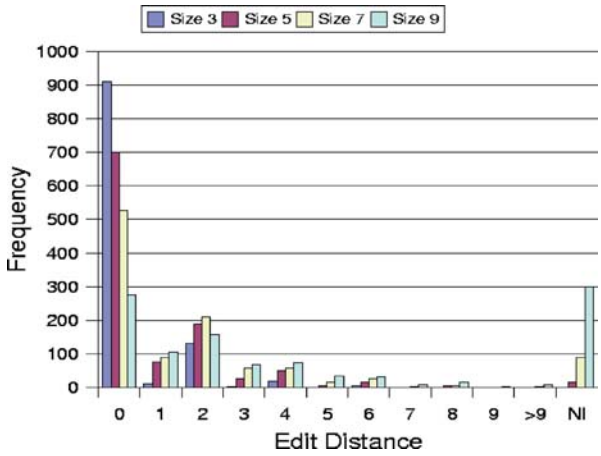


Figure 9. Frequency of edit-distances for all noise conditions (4280 trials).

were performed by randomly selecting from each argument 10–40% of the nodes to be deleted. Similarly, node insertions were performed by creating additional nodes to be inserted in the original argument (these nodes comprise 10–40% of the nodes in the argument).[11] Thus, the distorted (noisy) versions are "user arguments" whose interpretations should match the original arguments. These distortions were performed for Bayesian subnets (original arguments) of different sizes (3, 5, 7 and 9 arcs) yielding 4280 trials (1070 trials for each subnet size). Our measure of performance is the edit-distance between the original argument subnet and the interpretation subnet preferred by BIAS. That is, we counted the number of operations that need to be performed to match the source Bayesian subnet and the interpretation subnet. For instance, two subnets that match perfectly have an edit-distance of 0, and if the subnets differ by the position of one arc then the edit-distance is 2 (one addition and one deletion).

Overall, our results were as follows. Our system produced an interpretation in 91% of the 4280 trials. In 82% of the 4280 trials (91% of the trials where an interpretation was produced), the generated interpretations had an edit-distance of 3 or less from the original Bayesian subnet, and in 56% of the trials, the interpretations matched perfectly the original subnet. Figure 9 depicts the frequency of edit-distances for the different subnet sizes under all noise conditions. We plotted edit-distances of 0, ..., 9 and > 9, plus the category NI, which stands for 'No Interpretation'. As seen in Figure 9, the 0 edit-distance has the highest frequency for subnets of 7 arcs or less, and performance deteriorates as the size of the subnet increases. Further, for subnets of 7 arcs or less, interpretations were generated in 97% of the 3210 trials, and in 91% of the trials the interpretations had an edit-

[11] In our current implementation, the propositions in an argument are selected by the user from a menu. Hence, the user cannot present propositions that are unknown to BIAS, i.e., it is not possible to insert new nodes in an argument. However, node insertion was performed in our synthetic trials, as the NL version of our system allows users to mention propositions which the system cannot reconcile with nodes in the BN.

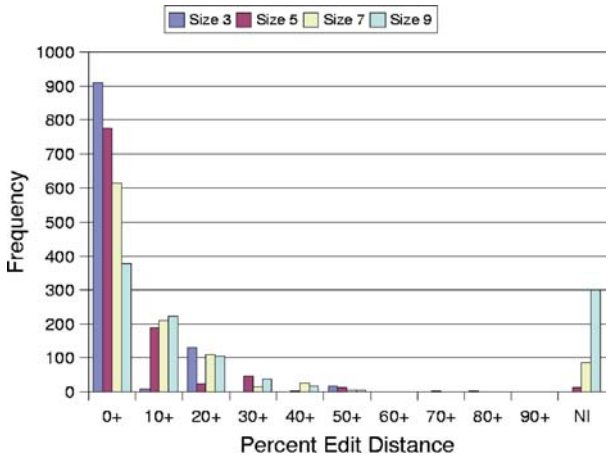


Figure 10. Frequency of edit-distances as percent of maximum edits for all noise conditions (4280 trials).

distance of 3 or less from the original argument subnet. Only for Bayesian subnets of 9 arcs the number of NIs exceeded the number of perfect matches.

The edit-distance measure alone may not inform us about the quality of the interpretations obtained by our system. For example, an interpretation of a small argument which differs from the original by 3 arcs or nodes may be inadequate. Hence, we offer a different view of our results. Figure 10 displays edit-distance as a percentage of the possible changes for an argument subnet of a particular size (the x-axis is divided into buckets of 10%). For example, if a selected interpretation differs from its source Bayesian subnet by the insertion of one arc, the percent-edit-distance will be $100 \times \frac{1}{(2N+1)}$, where N is the number of arcs in the source Bayesian subnet.[12] The results shown in Figure 10 are consistent with the previous results, with the vast majority of the edits being in the [0,10)% bucket. That is, most of the interpretations are within 10% of their source Bayesian subnets.

We also tested each kind of noise (distortion) separately, maintaining the other kinds of noise at 0%. All the distortions were between 0 and 40%. We performed 1560 trials for arc noise and node insertions, and 2040 trials for belief noise, which warranted additional observations. Figures 11 and 12 show the recognition accuracy of our system (in terms of average edit-distance) as a function of arc and belief noise percentages respectively (our system's performance for node insertions is similar to that obtained for belief noise). The performance for the different Bayesian subnet sizes (in arcs) is also shown. Our results indicate that the main factor that affects recognition performance is the size of the Bayesian subnet, while the average edit-distance remains stable for the different percentages of belief and arc noise, as well

[12] A graph of N arcs has a maximum of $N+1$ nodes, yielding a maximum of $2N+1$ edits to create the graph.

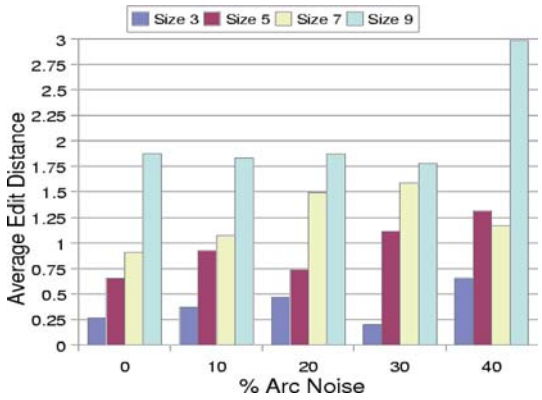


Figure 11. Effect of arc noise on performance (1560 trials).

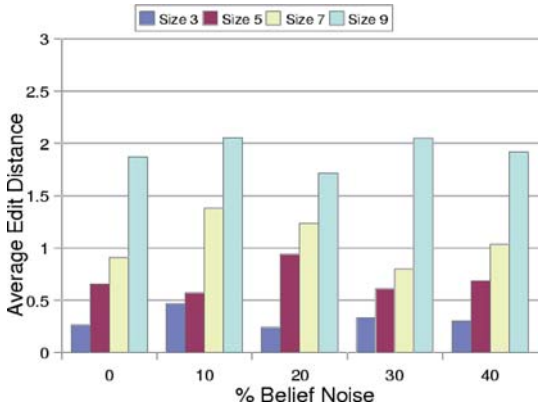


Figure 12. Effect of belief noise on performance (2040 trials).

as for node insertions. The only exception occurs for 40% arc noise and subnets of size 9, but even in this case the average edit-distance is only 3.

7. Incorporating the Complex User Model

Our basic probabilistic formalism assumes that every proposition is equally likely to be included in an interpretation. However, this may not be the case in reality. We postulate that interpretations including propositions familiar to the user (e.g., recently made observations, or propositions from his/her Notebook) are more probable than interpretations that include other domain propositions (although the user may still intend to include unseen propositions of which s/he has thought independently). More specifically, we posit that the probability of including a domain proposition in an interpretation depends on the salience (or level of activation) of the proposition in question in the user's focus of attention.

We model salience by means of three directly observable factors: (1) the type of access to a proposition, (2) the frequency and recency with which the proposition was accessed, and (3) its similarity to other accessed propositions. To derive from these factors the probability of including a node in an interpretation, we need to provide numerical values for these factors.

Type of access to a proposition. Users are more likely to recall propositions they have considered deliberately than propositions they have seen only in passing. We model this phenomenon by distinguishing between four types of access: observations may be `seen` or `accepted`, and statements may be `mentioned` or `confirmed` (these access types are similar to those introduced in Zukerman, 2001).

- `seen` observations are those that the user has encountered but has not acknowledged.
- `accepted` observations have been entered by the user in his/her Notebook.
- `mentioned` propositions have been explicitly included in the user's previous arguments.
- `confirmed` propositions were not mentioned by the user, but are incorporated by BIAS into an interpretation in order to connect the `mentioned` nodes. These propositions are `confirmed` when the user agrees with the interpretation presented by BIAS (Section 2).

We assign the following numerical strengths to our access categories. These strengths reflect the influence of the type of access on the salience of a proposition:

$$Str = \begin{cases} A & \text{if } \texttt{accepted} \\ A & \text{if } \texttt{mentioned} \\ \max\left\{\frac{A}{F_S}, \frac{A}{\#\_\text{of}\_\text{props}\_\text{seen}+1}\right\} & \text{if } \texttt{seen} \\ \frac{A}{F_C} & \text{if } \texttt{confirmed} \end{cases} \tag{10}$$

where A, F_S and F_C are constants determined during system development ($F_S > 1$ and $F_C > 1$). According to this formula, the strength of `seen` propositions is always less than the strength of `accepted` propositions, and is inversely proportional to the number of propositions viewed concurrently (e.g., read in the same web page) up to a certain number of propositions (we assume that this effect is monotonic only up to a point). The strength of `confirmed` propositions is slightly lower than that of `mentioned` propositions, since the user has confirmed these propositions in an interpretation, but has not presented them him/herself.[13]

Frequency and recency of access. Propositions that have been accessed frequently and recently are more likely to stand out in a user's memory than propositions accessed a

[13]Note that only `accepted` and `seen` propositions are relevant to the current operation of our system, as the user enters only one argument, which is interpreted and then validated.

while back. We model this influence by means of the following function.

$$\sum_{i=1}^{n}[CurTime - TimeStmp_i + 1]^{-b} \tag{11}$$

where n is the number of times a proposition was accessed, b (=1) is an exponent whose value was determined during system development, *CurTime* is the current time, and $TimeStmp_i$ is the time of the ith access. According to this formula, the level of activation of a node decays as a function of the time elapsed since its access. In addition, when a node is accessed, activation is added to the current accumulated (and decayed) activation. That is, there is a spike in the level of activation of the node, which starts decaying from that point again.

We use the following formula to express the salience of a node in terms of its access type and its frequency and recency (Zukerman et al., 2003a). This formula yields the access score of a node, assigning a high score to nodes that were recently `accepted` or `mentioned` by a user.

$$AccessScore(Nd) = \sum_{i=1}^{n} Str_i(Nd) \times [CurTime - TimeStmp_i + 1]^{-b} \tag{12}$$

where $Str_i(Nd)$ is the strength of the ith access to node Nd.

Node similarity. Propositions encountered by users are likely to remind them of similar propositions. For instance, [The neighbour saw Mr Green around the garden at 11] is similar to [Mr Green was in the garden at 11] and somewhat less similar to [Mr Green was in the garden]. Thus, encountering the first proposition during the domain investigation may prompt the user to think of the second proposition, and to a lesser extent of the third proposition.

We employ the procedure described in (Zukerman et al., 2003b) to calculate in advance the similarity between each pair of nodes in our domain BN. This procedure uses WordNet (Miller et al., 1990) and word-similarity information obtained from an automatically generated thesaurus to calculate a similarity score denoted $SimScore(Nd_i,Nd_j)$, where Nd_i and Nd_j are nodes in the domain BN (*SimScore* ranges between [0,1], with $SimScore(Nd_i,Nd_i)$=1).

The combined effect of access type, frequency and recency, and similarity on the activation of node Nd_k is modeled by passing to Nd_k the activation of accessed nodes (obtained from Equation 12) moderated by the degree of similarity between Nd_k and the accessed nodes. This is done by the following formula (we raise the factors to a power of 2 to polarize the effect of access and similarity).[14]

$$Score(Nd_k) = \sum_{j=1}^{N}[SimScore(Nd_k, Nd_j) \times AccessScore(Nd_j)]^2 \tag{13}$$

[14]This formula constitutes a cruder (and computationally more tractable) approximation of the salience of a node than that used in the NAG system (McConachy et al., 1998), where the salience of a node was dynamically calculated as nodes were mentioned.

where N is the number of nodes in the domain BN.

Equation 13 yields a score that reflects the salience of a node in the user's attentional focus. This score is used to derive $\Pr(m_k)$, the probability of including node Nd_k in an interpretation.

$$\Pr(m_k) = \frac{Score(Nd_k) + GC}{\sum_{i=1}^{N}[Score(Nd_i) + GC]} \tag{14}$$

where $GC = \frac{1}{2N}$ (GC is a small number that corresponds to Good's flattening constant (Good, 1965), Appendix C).

This probability in turn is used to calculate $\Pr(Nodes_{SI}|n_{SI}, dBN)$, instead of using the equiprobable distribution assumed in Section 5.2. The calculation of $\Pr(Nodes_{SI}|n_{SI}, dBN)$ is detailed in Appendix C. This calculation is based on a multinomial distribution of the random variables $m_1, \ldots, m_N$, adjusted to take into account the fact that these variables are dependent (a multinomial distribution assumes independent variables). It yields the following formula.

$$\Pr(Nodes_{SI}|n_{SI}, dBN) = \Pr'(m_1, \ldots, m_N) = n_{SI}! \prod_{\substack{k=1 \\ \forall m_k=1}}^{N} \frac{\Pr(m_k)}{\left\{1 - \sum_{\substack{j=1 \\ \forall m_j=1}}^{k-1} \Pr(m_j)\right\}} \tag{15}$$

where $m_k = 1$ if node $Nd_k \in dBN$ appears in $Nodes_{SI}$.

This equation is used instead of Equation A.4 from Appendix A to incorporate information from the complex user model into Equation 2 for the calculation of *SysIntBest*.

The complex user model is used only to calculate $\Pr(SysInt)$. We do not use it to calculate $\Pr(UArg|SysInt)$ because the effect of node probabilities on whether they are mentioned or implied is unclear: do users mention more probable nodes and leave less probable ones implicit, or do they mention the less probable nodes and assume that the more probable nodes will be understood? Therefore, when calculating $\Pr(UArg|SysInt)$ we use our basic formalism, which assumes a uniform distribution of configurations of non-leaf nodes from *SysInt* (Appendix B.2).

7.1. EXAMPLE

In this section we illustrate the contribution of the complex user model to the estimation of the probability of candidate interpretations of a simple argument. We also compare the effect of the user model that considers only access type, frequency and recency of a node, with the effect of the user model that also considers similarity. The former model uses $SimScore(Nd_k, Nd_j) = 0$ for $j \neq k$ and $SimScore(Nd_k, Nd_k) = 1$ when computing $Score(Nd_k)$ in Equation 13, while the latter uses the pre-calculated value of *SimScore* in Equation 13.

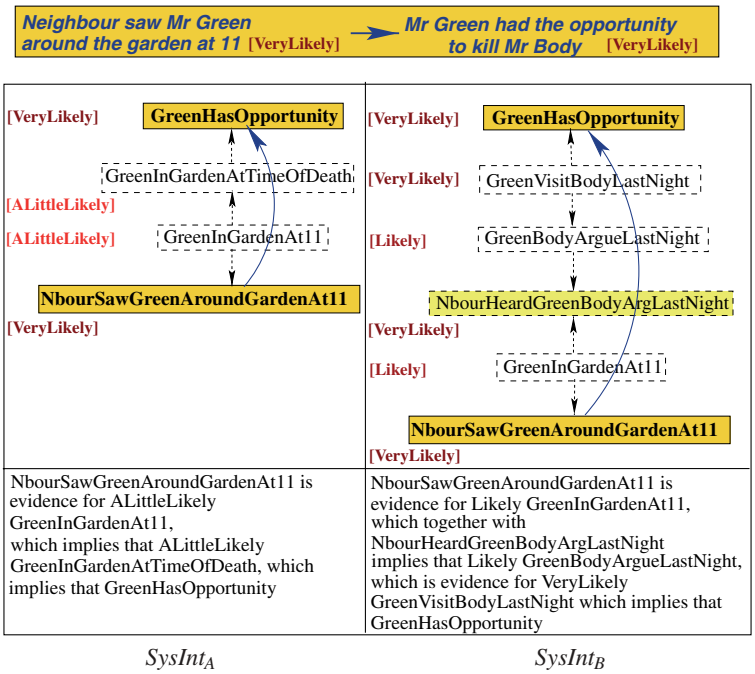


Figure 13. Simple argument and interpretations.

Let us reconsider our domain BN of 32 nodes (Figure 6), and consider the simple argument and interpretations in Figure 13 (these interpretations may be traced starting from node [NbourSawGreenAroundGardenAt11] at the bottom right of Figure 6). $SysInt_A$ has 4 nodes and 3 arcs, while $SysInt_B$ has 6 nodes and 5 arcs. The nodes mentioned by the user appear in bold and are dark shaded, while node [NbourHeardGreenBodyArgLastNight], which was encountered by the user during his domain exploration, is light shaded. Note that the relation between this node and its neighbours in $SysInt_B$ is a special *head-to-head* relation, which is different from simple causal or evidential relations (Jensen, 1996). Also, recall that the beliefs in argument interpretations are obtained by propagating the evidence nodes in the domain BN, cutting out the subnet corresponding to an interpretation, and then re-propagating beliefs (Section 4). This sometimes leads to unintuitive beliefs in the cut-out subnet, as illustrated in $SysInt_A$, where after propagation there is a substantial increase in belief from [GreenInGardenAtTimeOfDeath] ALittleLikely to [GreenHasOpportunity] VeryLikely. This point is further discussed in Section 8.

In order to consider the influence of access type, frequency and recency, and similarity without altering the belief in propositions, we introduce the notion of *gossip*. Gossip contains propositions the user may have heard of, but whose value s/he does not know. For instance, "you overheard the neighbours discussing whether Mr Green visited Mr Body last night". Table II shows the information in

Table II. Observations made before the simple argument in Figure 13.

Proposition	Value and source
Bullet wounds were found in Mr Body's body	T (police rep., `accepted`)
A gun was found in the garden	T (police rep., `accepted`)
Fingerprints were found on the gun	T (police rep., `accepted`)
Forensics established that the time of death was 11 pm	T (police rep., `accepted`)
Neighbour heard Mr Green and Mr Body argue last night	**T** (accepted)
Neighbour saw Mr Green around the garden at 11	**T** (**accepted**)
Mr Green and Mr Body argued last night	(*gossip*, `seen`)
Mr Green visited Mr Body last night	(*gossip*, `seen`)

the complex user model (the evidence in the antecedent of the user's argument is boldfaced, and gossip is italicized).

In this example, we first estimate the posterior probability of our candidate interpretations by calculating Pr(*SysInt*) using the simple user model, and then calculating Pr(*UArg*|*SysInt*). We then re-estimate Pr(*SysInt*) using both complex user models, and re-calculate the posterior probability of the interpretations.

Estimating Pr(*SysInt*)

$SysInt_A$ has 4 nodes ($n_A = 4$) and $SysInt_B$ has 6 nodes ($n_B = 6$). We now consider separately the effect of arcs and nodes on the prior probabilities of these interpretations.

- **Arcs –** Since both interpretations have a linear structure, the number of valid arcs in them is the minimum number of arcs: $va_{SI} = a_{SI} = n_{SI} - 1$ ($va_A = 4-1 = 3$ and $va_B = 6-1 = 5$). Hence, there is only one possible number of arcs for each interpretation, and there is only one way to choose a_A arcs from va_A arcs, and a_B arcs from va_B arcs. That is, $\Pr(a_A|Nodes_A, n_A, \mathrm{BN}) = \Pr(a_B|Nodes_B, n_B, \mathrm{BN}) = 1$, and $\Pr(Arcs_A|Nodes_A, a_A, n_A, \mathrm{BN}) = \Pr(Arcs_B|Nodes_B, a_B, n_B, \mathrm{BN}) = 1$.
- **Nodes –**
  - $\Pr(n_A|\mathrm{BN})$ and $\Pr(n_B|\mathrm{BN})$ are modeled by means of a truncated Poisson distribution with mean β. Using Equation A.3 from Appendix A, we obtain
    $$\Pr(n_A|\mathrm{BN}) = \delta\frac{e^{-\beta}\beta^4}{4!}$$
    $$\Pr(n_B|\mathrm{BN}) = \delta\frac{e^{-\beta}\beta^6}{6!}$$
    where δ is a normalizing constant.
  - $\Pr(Nodes_A|n_A, \mathrm{BN})$ and $\Pr(Nodes_B|n_B, \mathrm{BN})$ are modeled using the combinatorial formula for choosing n_{SI} nodes from the 32 nodes in the domain BN (Equation A.4), as follows:
    $$\Pr(Nodes_A|n_A, \mathrm{BN}) = \frac{1}{\binom{32}{4}}$$
    $$\Pr(Nodes_B|n_B, \mathrm{BN}) = \frac{1}{\binom{32}{6}}$$

Substituting these values in Equation 4, we obtain

$$\Pr(SysInt_A) = 1 \times 1 \times \frac{1}{\binom{32}{4}} \times \delta \frac{e^{-\beta}\beta^4}{4!}$$

$$\Pr(SysInt_B) = 1 \times 1 \times \frac{1}{\binom{32}{6}} \times \delta \frac{e^{-\beta}\beta^6}{6!}$$

For $\hat{\beta} = 11$ (where $\hat{\beta}$ is an estimate of β obtained experimentally, Appendix A), $SysInt_A$ is 6.25 times more probable than $SysInt_B$.

Estimating Pr(*UArg*|*SysInt*)

First, we consider the belief component of $\Pr(UArg|SysInt)$. Since both $SysInt_A$ and $SysInt_B$ match the beliefs stated by the user, $\Pr(Bel(UArg)|Bel(SysInt)) = 1$ for both interpretations.

We now consider the structure of $\Pr(UArg|SysInt)$, and as for *SysInt*, we examine separately the effect of arcs and nodes on the conditional probabilities $\Pr(Struct(UArg)|Struct(SysInt_A))$ and $\Pr(Struct(UArg)|Struct(SysInt_B))$.

– **Arcs**

- The user's argument has one arc only selected from the redirected arcs in $SysInt_A$ and $SysInt_B$ ($a_{\text{sel}} = 1$). Since the arc-redirection process yields one arc (that in *UArg*) for both interpretations, we obtain for both $SysInt_A$ and $SysInt_B$ $\Pr(a_{\text{sel}}|Nodes_{UA}, n_{UA}, SysInt) = 1$, and $\Pr(Arcs_{\text{sel}}|a_{\text{sel}}, Nodes_{UA}, n_{UA}, SysInt) = 1$.
- No extraneous arc insertions are required ($a_{\text{ins}} = 0$) for both interpretations. Hence, $\Pr(a_{\text{ins}}|Nodes_{UA}, n_{UA}, SysInt_A) = \Pr(a_{\text{ins}}|Nodes_{UA}, n_{UA}, SysInt_B)$, and $\Pr(\text{Arcs}_{\text{ins}}|a_{\text{ins}}, Nodes_{UA}, n_{UA}, SysInt_A) = \Pr(\text{Arcs}_{\text{ins}}|a_{\text{ins}}, Nodes_{UA}, n_{UA}, SysInt_B)$.

– **Nodes**

- The argument has two nodes ($n_{UA} = 2$). Thus, the truncated Poisson distribution from Equation B.9 yields

$$\Pr(n_{UA}|SysInt) = \phi \frac{e^{-\rho}\rho^2}{2!}$$

for both interpretations, where ϕ is a normalizing constant, and ρ is the average number of nodes in an argument. However, as indicated in Appendix B, our previous experimental results show that users' arguments typically contain half the nodes in their interpretations, i.e., $\rho = n_{SI}/2$. Hence, the

probabilities obtained for n_{UA} given $SysInt_A$ (which has four nodes) and n_{UA} given $SysInt_B$ (which has six nodes) differ as follows:

$$\Pr(n_{UA}|SysInt_A) = \phi \frac{e^{-n_A/2}(\frac{n_A}{2})^2}{2!} = \phi \frac{e^{-4/2}(\frac{4}{2})^2}{2!} = \phi \frac{e^{-2}2^2}{2!}$$

$$\Pr(n_{UA}|SysInt_B) = \phi \frac{e^{-n_B/2}(\frac{n_B}{2})^2}{2!} = \phi \frac{e^{-6/2}(\frac{6}{2})^2}{2!} = \phi \frac{e^{-3}3^2}{2!}$$

- $\Pr(Nodes_{UA}|n_{UA}, SysInt_A)$ and $\Pr(Nodes_{UA}|n_{UA}, SysInt_B)$ are modeled using the combinatorial formula for choosing the non-leaf nodes in *UArg* from the non-leaf nodes in *SysInt* (Equation B.10). However, both nodes in *UArg* are leaf nodes. Hence, for both interpretations, there is only one way to choose the 0 non-leaf nodes in *UArg* from those in *SysInt*.

Substituting these values in Equation 8, we obtain

$$\begin{aligned}&\Pr(Struct(UArg)|Struct(SysInt_A))\\ &\quad = 1 \times 1 \times \Pr(Arcs_{\text{ins}}|a_{\text{ins}}, Nodes_{UA}, n_{UA}, SysInt_A)\\ &\qquad \times \Pr(a_{\text{ins}}|Nodes_{UA}, n_{UA}, SysInt_A) \times 1 \times \phi \frac{e^{-2}2^2}{2!}\end{aligned}$$

$$\begin{aligned}&\Pr(Struct(UArg)|Struct(SysInt_B))\\ &\quad = 1 \times 1 \times \Pr(Arcs_{\text{ins}}|a_{\text{ins}}, Nodes_{UA}, n_{UA}, SysInt_B)\\ &\qquad \times \Pr(a_{\text{ins}}|Nodes_{UA}, n_{UA}, SysInt_B) \times 1 \times \phi \frac{e^{-3}3^2}{2!}\end{aligned}$$

Thus, $\Pr(UArg|SysInt_A) \cong 1.21 \Pr(UArg|SysInt_B)$.

Recall that the prior probability of $SysInt_A$ is 6.25 times that of $SysInt_B$. In order to put together these results, we return to Equations 1 and 2, and express the posterior probability for $SysInt_A$ in terms of the probabilities for $SysInt_B$. This yields

$$\begin{aligned}\Pr(SysInt_A|UArg) &= \frac{\Pr(SysInt_A) \times \Pr(UArg|SysInt_A)}{\Pr(UArg)}\\ &= \frac{6.25 \Pr(SysInt_B) \times 1.21 \Pr(UArg|SysInt_B)}{\Pr(UArg)}\\ &= \frac{7.89 \Pr(SysInt_B) \times \Pr(UArg|SysInt_B)}{\Pr(UArg)}\\ \Pr(SysInt_A|UArg) &= 7.89 \Pr(SysInt_B|UArg) \qquad (16)\end{aligned}$$

Table III. Probability of including a node in an interpretation.

Node	*AccessScore*	*AccessScore&SimScore*
[GreenBodyArgueLastNight]	0.0525	0.0442
[GreenInGardenAt11]	0.0076	0.0374
[GreenInGardenAtTimeOfDeath]	0.0076	0.0163
[GreenHasOpportunity]	0.1857	0.1151
[GreenVisitBodyLastNight]	0.0525	0.0429
[NbourHeardGreenBodyArgLastNight]	0.0531	0.0870
[NbourSawGreenAroundGardenAt11]	0.3296	0.2287

That is, given *UArg*, $SysInt_A$ is 7.89 times more probable than $SysInt_B$. This is because $SysInt_A$ is significantly shorter than $SysInt_B$, and $SysInt_B$ has nothing to offer to overcome its length disadvantage.

Consulting the complex user model to reestimate Pr(*SysInt*)

We now consider the effect of the two complex user models on the probability of our two interpretations. These models are *AccessScore*, which considers access type, frequency and recency; and *AccessScore&SimScore*, which also considers the similarity between propositions.

In both models, the gossip together with the accepted evidence influence the probabilities of including the nodes in the domain BN in *SysInt*. Table III shows the probabilities obtained from Equation 14 for the nodes in $SysInt_A$ and $SysInt_B$ for the *AccessScore* model and the *AccessScore&SimScore* model (these probabilities add up to less than 1, as all the nodes in the BN have some probability to be included in *SysInt*). For comparison, if we assume an equiprobable node distribution, the probability of including a node in *SysInt* is $\frac{1}{32} = 0.0313$.

Substituting the probabilities in Table III in Equation 15 gives the following results for the *AccessScore* model (as mentioned in Appendix C, the probabilities are retrieved in alphanumeric order of the node names stored in the system, which is the order in Table III).

$$\begin{aligned}&\Pr(Nodes_A|n_A, \mathrm{BN})\\ &= 4! \times \frac{0.0076}{1} \times \frac{0.0076}{1-0.0076} \times \frac{0.1857}{1-0.0076-0.0076} \times \frac{0.3296}{1-0.0076\times 2-0.1857}\\ &= 4! \times 0.0076 \times 0.0077 \times 0.1886 \times 0.4125 = 0.00011\end{aligned}$$

and

$$\begin{aligned}&\Pr(Nodes_B|n_B, \mathrm{BN})\\ &= 6! \times \frac{0.0525}{1} \times \frac{0.0076}{0.9475} \times \frac{0.1857}{0.9399} \times \frac{0.0525}{0.7542} \times \frac{0.0531}{0.7017} \times \frac{0.3296}{0.6486}\\ &= 6! \times 0.0525 \times 0.0080 \times 0.1976 \times 0.0696 \times 0.0757 \times 0.5082 = 0.00016\end{aligned}$$

Substituting these results into Equation 4, together with the other components calculated for the simple user model at the beginning of this section, we obtain the

following results (the new component appears in boldface).

$$\Pr(SysInt_A) = 1 \times 1 \times \mathbf{1.1} \times \mathbf{10^{-4}} \times \delta \frac{e^{-\beta}\beta^4}{4!}$$

$$\Pr(SysInt_B) = 1 \times 1 \times \mathbf{1.6} \times \mathbf{10^{-4}} \times \delta \frac{e^{-\beta}\beta^6}{6!}$$

For $\hat{\beta} = 11$, this yields $\Pr(SysInt_A) = 0.17 \Pr(SysInt_B)$, in contrast to the result obtained for the simple user model, where $SysInt_A$ was 6.25 times more probable than $SysInt_B$. Substituting these results in Equations 1 and 2, and expressing the posterior probability for $SysInt_A$ in terms of the probabilities for $SysInt_B$ we obtain

$$\begin{aligned}\Pr(SysInt_A|UArg) &= \frac{\Pr(SysInt_A) \times \Pr(UArg|SysInt_A)}{\Pr(UArg)} \\ &= \frac{0.17 \Pr(SysInt_B) \times 1.21 \Pr(UArg|SysInt_B)}{\Pr(UArg)} \\ &= \frac{0.21 \Pr(SysInt_B) \times \Pr(UArg|SysInt_B)}{\Pr(UArg)} \\ \Pr(SysInt_A|UArg) &= 0.21 \Pr(SysInt_B|UArg) \end{aligned} \tag{17}$$

Thus, when consulting the *AccessScore* user model, $SysInt_B$ given *UArg* is nearly 5 times more probable than $SysInt_A$ given *UArg*.

Repeating the same calculation for the *AccessScore&SimScore* model yields the following results:

$$\Pr(SysInt_A) = 1 \times 1 \times \mathbf{2.1} \times \mathbf{10^{-5}} \times \delta \frac{e^{-\beta}\beta^4}{4!}$$

and

$$\Pr(SysInt_B) = 1 \times 1 \times \mathbf{4.5} \times \mathbf{10^{-7}} \times \delta \frac{e^{-\beta}\beta^6}{6!}$$

For $\hat{\beta} = 11$, this yields $\Pr(SysInt_A) = 11.57 \Pr(SysInt_B)$. Thus, in this example, the similarity scores reverse the effect of access, frequency and recency. When we substitute these results in Equations 1 and 2, $SysInt_A$ becomes the winning interpretation by a higher margin than for the simple user model.

$$\begin{aligned}\Pr(SysInt_A|UArg) &= \frac{11.57 \Pr(SysInt_B) \times 1.21 \Pr(UArg|SysInt_B)}{\Pr(UArg)} \\ \Pr(SysInt_A|UArg) &= 14 \Pr(SysInt_B|UArg) \end{aligned} \tag{18}$$

In general, the behaviour of a user model with similarity scores is more unpredictable than the behaviour of a user model without these scores, as sometimes these scores increase the probability of a particular interpretation, while other times they blur the differences between the probabilities of interpretations. Two of the trial sets in our user-based evaluation were designed to find out which user model yields interpretations preferred by users (Section 8).

8. Evaluation with Users

Our evaluation with users was designed to address the following objectives:

EvalObj 1. Determine whether our probabilistic approach to argument interpretation yields interpretations that are acceptable to users.

EvalObj 2. Gain insights into which user model yields the best interpretations (*Simple, AccessScore* or *AccessScore&SimScore*).

Note that these evaluation objectives address the main tenet of our approach only indirectly. Instead of determining whether users intend the interpretation with the highest posterior probability, the first objective determines whether people reading someone else's argument find the highest-probability interpretation(s) acceptable (and better than other options). The second objective determines which user model yields the highest-probability interpretation that best matches people's preferences.

We prepared four pencil-and-paper evaluation sets, which were designated with names of colours (PURPLE, BLUE, RED and WHITE). Our subjects were staff and students at Monash University and friends and family of the authors (the subjects exhibited different levels of computer literacy). All four sets were shown to our subjects, but not all the subjects completed all the sets. Each set consisted of the following items.

- Police report – same as that shown in Section 2.
- Additional facts – instantiated propositions which simulate the information gained by the user when exploring the virtual scenario (Section 2).
- Gossip – uninstantiated propositions "overheard" by the user, which were added to highlight the reminding effect of propositions (Section 7).
- Argument and interpretation(s) – our subjects were told that the argument was given by a hypothetical user and that the interpretations were generated by a computer system. They were then asked to give each interpretation a score between 1 (Very UNreasonable) and 5 (Very Reasonable) in light of the police report, additional facts and gossip, and to comment on aspects of the interpretations that they liked or disliked.

We used a pencil-and-paper evaluation rather than a full system evaluation, since we wanted users to have a uniform experience with the interpretation capabilities of the system, and we wanted to focus on small arguments that distinguish between the interpretations produced by the different user models.[15] Our previous experience shows that when users interact freely with the system, their arguments may not test behaviours of interest, and their assessment of the interpretations may be influenced by their experience with the web interface (Zukerman et al., 2003a).

[15]Note that, as illustrated in Figure 4, our system can handle complex arguments. Further, as illustrated in our examples, small arguments do not necessarily entail small interpretations. Hence, the argument interpretation task is challenging even for small arguments.

Table IV. Results of the user-based evaluation.

Eval set	# People	Mean (standard deviation)		
PURPLE	24	4.00 (1.02)		
BLUE	17	$SysInt_A$ 2.88 (0.99)	$SysInt_B$ **3.38 (1.45)**	$SysInt_C$ 2.94 (1.25)
RED	25	$SysInt_A$ (*Simple* and *AccSim*) **3.68 (1.11)**		$SysInt_B$ (*Acc*) 3.36 (1.29)
WHITE	20	$SysInt_A$ (*Simple* and *Acc*) 2.80 (1.06)		$SysInt_B$ (*AccSim*) **3.35 (1.39)**

Table V. Significance of interpretation preferences.

EvalSet	Interpretation pair	Significance
BLUE	$SysInt_B$–$SysInt_A$	90% ($p = 0.1$)
	$SysInt_B$–$SysInt_C$	94% ($p = 0.06$)
RED	$SysInt_A$–$SysInt_B$	85% ($p = 0.15$)
WHITE	$SysInt_B$–$SysInt_A$	93% ($p = 0.07$)

The evaluation sets were designed to address our evaluation objectives as follows.

EvalObj 1. The PURPLE and BLUE sets were designed for this objective. The PURPLE set contains a reasonably complex argument (Figure 4) and only one interpretation – that preferred by BIAS (Figure 5). The BLUE set contains a simple argument (Figure 8), and the top three interpretations generated by BIAS ($SysInt_B$ from Figure 8 is ranked equal first with another interpretation (denoted $SysInt_C$), and $SysInt_A$ is ranked second).

EvalObj 2. The RED and WHITE sets were designed for this objective. The RED set contains the argument and interpretations in Figure 13. In this set, $SysInt_A$ is preferred by the *Simple* user model and the *AccessScore&SimScore* model, while $SysInt_B$ is preferred by the *AccessScore* model. In the WHITE set, $SysInt_A$ is preferred by the *Simple* user model and the *AccessScore* model, while $SysInt_B$ is preferred by the *AccessScore&SimScore* model.

Table IV summarizes the results of our user-based evaluation. The first column contains the evaluation set, the second column shows the number of people who participated in each set, and subsequent columns contain the mean and standard deviation of the scores given by the users to the candidate interpretations. The scores for the preferred interpretations have been boldfaced.

We used a paired Z-test to assess the significance of our results. This was done by calculating for each pair of interpretations the difference in the scores assigned to them by each trial subject. Table V shows the results of these calculations, which indicate how much users preferred one interpretation over another.

Although the statistical significance of our results is lower than we had hoped, we can still make the following observations regarding our evaluation objectives.

EvalObj 1. People generally found our interpretations acceptable, with an average score that is better than neutral for most interpretations, and a high average score (=4) for the interpretation in the PURPLE evaluation set. Also, the ranked order derived from the average scores of the three interpretations in the BLUE evaluation set ($SysInt_B > SysInt_C > SysInt_A$) is consistent with the ranked order obtained by BIAS.

EvalObj 2. People seemed to prefer the interpretations generated using the *AccessScore&SimScore* user model. However, these preferences should be treated with caution because they were obtained from two experiments only (each with several users).

Thus, we feel that the question posed by **EvalObj 1** has been positively answered, and the question posed by **EvalObj 2** yielded interesting insights but requires further investigation. Even more encouraging is the fact that most of the problems our subjects pointed out regarding BIAS' interpretations are extraneous to the interpretation-selection process. These problems concern domain-related inferences that our subjects disagreed with, unexpected jumps in belief, or levels of belief in the consequents of implications.

- **Domain-related inferences –** We selected a 'commonsense' domain both for ease of design and to be able to conduct trials with non-experts. The nodes and arcs in the domain BN and the values in the CPTs were designed by the authors. A consequence of working in a commonsense domain is that the system's domain knowledge is limited and sometimes idiosyncratic. Thus, users may consider different factors than those considered by the system, and disagree with the system's inferences. For instance, according to BIAS, Mr Green and Mr Body being enemies implies that Mr Green very likely has a motive to kill Mr Body. However, several users disagreed with this inference. When an interpretation contained such inferences, users tended to dismiss the entire interpretation.
- **Unexpected jumps in belief –** Jumps in belief take place when the belief value for a consequent is not what the user had anticipated from the antecedent. For instance, one interpretation in the WHITE evaluation set says that '*It is likely that Mr Green had the means to murder Mr Body implies that it is* a little unlikely *that he murdered Mr Body*'. This unexpected jump in belief is caused by the unmentioned fact that Mr Green is unlikely to have the opportunity to murder Mr Body, thereby lowering the overall probability of Mr Green's guilt. Such jumps in belief caused the majority of the negative comments from our users.
- **Levels of belief –** Levels of belief in the consequents of an argument are especially important to users. Users reacted far more strongly than we had expected to slight discrepancies between beliefs stated in an argument and beliefs inferred in its interpretation by means of Bayesian propagation.

In order to address these problems, we propose to do the following.

- Consider domains that are more circumscribed than our current domain, e.g., technical domains, where it should be possible to model most factors (in contrast to the murder mystery domain, where users thought of several factors not modeled by the system). Also, the influence of these factors should be less debatable than the influence of the factors in the murder mystery.
- Address jumps in belief during the presentation of interpretations. This may be done by including in a presented interpretation factors that explain jumps in belief, e.g., propositions that have a significant effect on the consequent of an implication (Jitnah et al., 2000; Zukerman et al., 2004). Adaptive hypertext links could be used to provide this extra information (Bontcheva and Wilks, this issue).
- Address discrepancies between the beliefs in a user's argument and the beliefs in its interpretation. Discrepancies in belief may be reduced or completely removed during the generation of interpretations by postulating assumptions that explain the user's beliefs. These assumptions would then be presented to the user for validation.[16] Remaining discrepancies in belief should be acknowledged during the presentation of interpretations, rather than leaving them for the user to notice.
- Consider different functions for estimating $\Pr(Bel(UArg)|Bel(SysInt))$ from discrepancies in belief in order to increase the impact of such discrepancies.

It is worth noting that both the jumps in belief and the levels of belief in an interpretation are in part a result of the process used for generating interpretation subnets, which is carried out according to BN theory (Section 4). This process involves marginalizing parent nodes and deleting child nodes. The marginalization causes problems when presenting an interpretation, as the belief in a consequent node may take into account influences from parent nodes that have been marginalized, and hence it may not exactly follow from the antecedents of this node. Conversely, ignoring child nodes may lead to interpretations that do not take into account relevant evidence. In the future, we propose to address these problems by investigating a different approach to the generation of interpretations, where we will not cut out Bayesian subnets from the domain BN.

9. Related Research

This research builds on an earlier version of BIAS. This version used a domain and user model represented as a BN to generate arguments and rebuttals, and combined this representation with linguistic and attentional information to interpret single-proposition rejoinders entered by users after reading the system's arguments (Zukerman, 2001). The combination of these knowledge sources was based

[16] For research on user model inspection and validation see (Bull and Pain, 1995; Kay, 1999).

on heuristics. In later work, we investigated a principled approach for the selection of an interpretation of users' arguments based on the Minimum Message Length (MML) Principle (Wallace and Boulton, 1968; Wallace, 2005). We applied this principle to evaluate candidate interpretations of arguments of arbitrary complexity (Zukerman and George, 2002). In (Zukerman et al., 2003a) we incorporated a user model into this formalism.

In this paper, we integrate the last two contributions and provide a probabilistic representation for the interpretation-selection problem. We posit that the interpretation intended by a speaker is that to which the system ascribes the highest posterior probability. Thus, discourse interpretation is cast as the problem of finding the maximum-probability representation of the user's discourse according to the system's model of the world.

Graphical techniques for analyzing arguments include the well-known Toulmin warrant structure (Toulmin, 1958), Cohen's tree structures (Cohen, 1987), and Walton's argument schemes (Walton, 1996). The Toulmin warrant structure contains the following elements: *claim* – the argument goal; *data* – the evidence for the claim; *warrant* and *backing* – the reasoning used to link the data to the claim; *qualifier* – a phrase modifying the claim to indicate its strength; and *reservations* – circumstances or conditions that undermine the argument. Walton's structure for argument analysis is based on the identification of *schemes*, such as 'Appeal to Expert Opinion' and 'Argument from Position to Know'. These schemes, which are at a coarser level of granularity than Toulmin's warrant structure, have been implemented in the Araucaria system – a markup tool for argumentation (Reed and Walton, 2003). Cohen's method of argument analysis uses linguistic clues and the order of the statements in an argument to build a tree structure that represents the argument (Cohen, 1987). Each statement in the argument is represented by a node in the tree. The tree is built so that each node or statement offers support for its parent in the tree.

There are two important distinctions between the interpretation technique implemented in BIAS and these techniques. First, these techniques are analysis tools rather than interpretation systems. Second, BIAS integrates a user's argument into its world model, i.e., it uses its domain knowledge to infer information left implicit by the user and to distinguish between alternative interpretations. In contrast, the above techniques focus on analyzing the structure of a stand-alone argument, outside the context provided by domain knowledge. A minor difference between BIAS and these systems is that these systems distinguish between antecedents that support and antecedents that detract from their consequents, while the current version of BIAS does not offer this distinction in its argument construction interface.

Several researchers have viewed discourse interpretation as the process of integrating the contribution of a conversational partner (the speaker) into the addressee's mental model, e.g., (Kintsch, 1994; Kashihara et al., 1995). Kintsch demonstrated this view experimentally, while Kashihara et al. implemented it in a

discourse planning system. This system generated discourse which would present an 'optimal' cognitive load to the addressee when trying to integrate this discourse into his/her mental model.

Plan recognition systems also have this view of discourse interpretation. These systems generate one or more interpretations of a user's utterances, employing different resources to fill in the information omitted by the user, e.g., (Allen and Perrault, 1980; Litman and Allen, 1987; Raskutti and Zukerman, 1991; Quilici, 1992; Carberry and Lambert, 1999; Restificar et al., 1999). Allen and Perrault's seminal work on plan recognition describes a mechanism that employs domain-independent heuristics to select an interpretation for a statement presented by a user. Litman and Allen extended this mechanism by means of discourse plans. They used linguistic clues and coherence heuristics to infer discourse plans, which enabled them to handle multiple-sentence user inputs. Raskutti and Zukerman developed a probabilistic approach to the interpretation-selection problem. They used heuristics similar to those devised by Allen and Perrault and by Litman and Allen to estimate the probability of an interpretation, and applied information content considerations to determine which interpretations to retain for further processing. Quilici studied the generation and recognition of the justification for a proposal in a plan-based context. Both tasks were performed by applying a set of justification rules in backward chaining mode from the proposal to known premises. Carberry and Lambert's system recognized a user's intentions during expert-consultation dialogues, considering several knowledge sources, such as linguistic characteristics of the user's contribution, dialogue context, and stereotypical beliefs presumed shared by the user and the system. Finally, Restificar et al. applied argument schemata to recognize a user's intentions from his/her rejoinders to the system's arguments, and to generate short rebuttals to these rejoinders.

All of these systems dealt with dialogues where users' contributions were quite short, while BIAS interprets arguments of arbitrary length. More importantly, these systems relied on heuristics to select an interpretation, while BIAS offers a principled approach based on maximum posterior probability. Another difference between BIAS and these systems pertains to the knowledge representation formalism: these systems used plan libraries, while BIAS relies on BNs. This difference affects mainly the mechanism used for the generation of interpretations (Section 4). Our probabilistic approach for the evaluation and selection of an interpretation (Sections 5 and 7) is representation independent, but as seen in the Appendix, a network representation is assumed for the implementation of this approach. Finally, the systems described in (Litman and Allen, 1987; Carberry and Lambert, 1999) used linguistic features of the user's discourse during the interpretation-selection process. In the future, we expect to consider these features in the version of BIAS that accepts NL input.

BNs have been used in several plan recognition tasks, e.g., (Charniak and Goldman, 1993; Gertner et al., 1998; Horvitz and Paek, 1999). Charniak and Goldman's system handled complex narratives. It automatically built and incre-

mentally extended a BN from propositions read in a story, so that the BN represented hypotheses that became plausible as the story unfolded. During this process, Charniak and Goldman used marker passing to restrict the nodes included in the BN. In contrast, we use a domain BN to constrain our understanding of the propositions in a user's argument. In addition, our *AccessScore&SimScore* model uses a process similar to marker passing to moderate the probabilities of including nodes in an interpretation (rather than outright including or excluding nodes). Gertner et al. used BNs to represent solutions of physics problems. After observing an action performed by a student, their system (Andes) postulated candidate interpretations, each hypothesizing subsequent actions, and selected the interpretation with the highest probability (subject to tie-breaking heuristics). Since BIAS is presented with a complete argument, it not only takes into account the probability of an interpretation in the context of existing information, but also considers the fit between the argument and the interpretation. Horvitz and Paek used BNs at different levels of an abstraction hierarchy to infer a user's goal in information-seeking interactions with a Bayesian Receptionist. Their system considered linguistic distinctions obtained from an NL parser, and like the above systems, it handled short dialogue contributions. However, Horvitz and Paek used decision-theoretic strategies to guide the progress of the dialogue. We expect to employ such dialogue strategies when our system engages in a full dialogue with the user. We also envisage that these strategies could take into account predictions regarding the effectiveness of an interaction (Goodman et al., this issue).

10. Conclusion

We have offered a probabilistic mechanism that generates interpretations of extended arguments in the context of a BN. Our mechanism, which estimates the posterior probability of candidate interpretations of a user's argument, provides a theoretically sound framework for selecting a plausible interpretation among available options. This framework (1) enables us to take into account different information sources, such as domain knowledge, user model and attentional model, during the interpretation process; and (2) allows us to represent belief and structural discrepancies between the system's domain representation and the arguments produced by people (which typically contain inferential leaps).

As seen in the Appendix, our mechanism relies on careful and efficient modeling of the dependencies between variables. Otherwise, inaccurate probabilities may be assigned to different components of an interpretation, which in turn may cause the model to make inappropriate choices. For instance, a simple combinatorial model where the nodes in *UArg* are selected from the nodes in *SysInt* unnecessarily halves the probability of longer interpretations, thereby reducing their chances of winning.

Our synthetic evaluation yielded promising results, with interpretations that match perfectly or almost-perfectly the source Bayesian subnet being generated in

82% of the cases under all distortion conditions. Our user-based evaluation was designed to determine the general appropriateness of the interpretations generated by BIAS, and to gain insights regarding BIAS' performance when using our three user models: simple, complex with access information only, and complex with both access and similarity information. This evaluation pointed to problems concerning (1) the implementation domain, (2) our use of BNs to model human inference, and (3) the amount and kind of detail in the presentation of interpretations. These difficulties detracted from the scores assigned by our trial subjects to the interpretations. Despite this, our results are encouraging with respect to the general suitability of BIAS' interpretations, and indicated a slight preference for the complex user model with access and similarity information. In the future, we propose to modify our usage of BNs, and also improve the presentation of interpretations, so that our evaluation can focus on the interpretation process.

Acknowledgements

This research was supported in part by Australian Research Council Grants A49927212 and DP0344013, and by the ARC Centre for Perceptive and Intelligent Machines in Complex Environments. The authors thank Alfred Kobsa, Sandra Carberry and the three anonymous reviewers for their helpful suggestions.

Appendix

A. Calculating the Probability of an Interpretation

Let us consider a domain BN and an interpretation *SysInt* obtained from this BN. We adopt the following notation:

dBN – the domain BN (composed of arcs and nodes),
N – number of nodes in dBN,
A – number of arcs in dBN,
$Nodes_{SI}$ – the set of nodes in the interpretation *SysInt*,
$Arcs_{SI}$ – the set of arcs in *SysInt*,
n_{SI} – the number of nodes in *SysInt*, i.e., $|Nodes_{SI}|$,
a_{SI} – the number of arcs in *SysInt*, i.e., $|Arcs_{SI}|$,
$Nodes_{UA}$ – the set of nodes in the user's argument *UArg*,
$Arcs_{UA}$ – the set of arcs in *UArg*,
n_{UA} – the number of nodes in *UArg*, i.e., $|Nodes_{UA}|$,
a_{UA} – the number of arcs in *UArg*, i.e., $|Arcs_{UA}|$.

We identify an interpretation by specifying the number of nodes in it, the number of arcs, and the actual nodes and arcs in it. Thus, the probability of an interpretation *SysInt*, $\Pr(SysInt)$, in the context of the domain BN is defined as

$$\Pr(SysInt) = \Pr(Arcs_{SI}, Nodes_{SI}, a_{SI}, n_{SI}|dBN) \tag{A.1}$$

Applying the chain rule of probability theory yields

$$\begin{aligned}\Pr(SysInt) = {} & \Pr(Arcs_{SI}|Nodes_{SI}, a_{SI}, n_{SI}, dBN) \times \Pr(a_{SI}|Nodes_{SI}, n_{SI}, dBN) \\ & \times \Pr(Nodes_{SI}|n_{SI}, dBN) \times \Pr(n_{SI}|dBN) \end{aligned} \tag{A.2}$$

These probabilities are calculated as follows.

– $\Pr(n_{SI}|dBN)$ – Our preliminary experiments, where 10 users entered arguments into the system (Zukerman et al., 2003a), show that the number of nodes in an interpretation may be modeled using a truncated Poisson distribution with mean β. In these experiments, $\hat{\beta}$, the estimate of β from our sample data, was 11, with few interpretations being much shorter or much longer. Thus,

$$\Pr(n_{SI}|dBN) = \begin{cases} \delta \dfrac{e^{-\beta}\beta^{n_{SI}}}{n_{SI}!} & \text{if } n_{SI} \leq N \\ 0 & \text{otherwise} \end{cases} \tag{A.3}$$

where $\hat{\beta} = 11$ and δ is a normalizing constant.

– $\Pr(Nodes_{SI}|n_{SI}, dBN)$ – We want to select the n_{SI} nodes in $Nodes_{SI}$ from the N nodes in dBN. Assuming that all nodes have an equal probability of being selected, there are $\binom{N}{n_{SI}}$ ways to select these nodes. Thus, the probability of any particular configuration of n_{SI} nodes is

$$\Pr(Nodes_{SI}|n_{SI}, dBN) = \frac{1}{\binom{N}{n_{SI}}} \tag{A.4}$$

– $\Pr(a_{SI}|Nodes_{SI}, n_{SI}, dBN)$ – Only arcs joined at both ends to nodes in *SysInt* can be in *SysInt*. Hence, by knowing which nodes are in *SysInt* we can automatically define the *valid arcs* that connect the nodes in *SysInt*. Let us denote these arcs $ValidArcs_{SI}$, and let the number of valid arcs be va_{SI}. It is reasonable to assume that the number of arcs in an interpretation is uniformly distributed between $n_{SI}-1$ (the minimum number of arcs that connect n_{SI} nodes) and va_{SI}.[17] This yields

$$\Pr(a_{SI}|Nodes_{SI}, n_{SI}, dBN) = \frac{1}{va_{SI} - (n_{SI} - 1) + 1} \tag{A.5}$$

– $\Pr(Arcs_{SI}|Nodes_{SI}, a_{SI}, n_{SI}, dBN)$. We want to select the a_{SI} arcs in $Arcs_{SI}$ from the va_{SI} valid arcs in *SysInt*. There are $\binom{va_{SI}}{a_{SI}}$ ways to select these arcs.[17] Thus, the probability of any particular configuration of a_{SI} arcs in *SysInt* is

$$\Pr(Arcs_{SI}|a_{SI}, Nodes_{SI}) = \frac{1}{\binom{va_{SI}}{a_{SI}}} \tag{A.6}$$

[17] At present, our procedure for generating interpretations produces only trees. Hence, $va_{SI} = n_{SI} - 1$, and $a_{SI} = va_{SI}$. However, we retain this component, as in the future this procedure will generate graphs.

B. Calculating the Probability of the Argument given an Interpretation

Let us consider a user's argument *UArg* and an interpretation *SysInt*. The probability that a user presented argument *UArg* when s/he intended interpretation *SysInt* is Pr(*UArg*|*SysInt*). In order to define this probability, we distinguish between the structure of an argument and the beliefs stated about the nodes in the argument. The structure of the user's argument is represented by *Struct(UArg)*, and that of the system's interpretation by *Struct(SysInt)*. Similarly, the beliefs stated in the user's argument are represented by *Bel(UArg)* and those in the interpretation by *Bel(SysInt)*. Hence,

$$\Pr(UArg|SysInt) = \Pr(Struct(UArg),\ Bel(UArg)|Struct(SysInt),\ Bel(SysInt)) \tag{B.1}$$

Applying the chain rule of probability theory yields

$$\begin{aligned}\Pr(UArg|SysInt) = {} & \Pr(Struct(UArg)|Bel(UArg),\ Struct(SysInt),\ Bel(SysInt)) \\ & \times \Pr(Bel(UArg)|Struct(SysInt),\ Bel(SysInt))\end{aligned}$$

We now make the following simplifying assumptions.

- Given *Struct(SysInt)*, *Struct(UArg)* is conditionally independent of *Bel(UArg)* and *Bel(SysInt)* i.e., structure is not derived from beliefs.
- Given *Bel(SysInt)*, *Bel(UArg)* is conditionally independent of *Struct(SysInt)* i.e., the beliefs in *UArg* are independent of the structure of *SysInt*.

This yields

$$\begin{aligned}\Pr(UArg|SysInt) = {} & \Pr(Struct(UArg)|Struct(SysInt)) \\ & \times \Pr(Bel(UArg)|Bel(SysInt))\end{aligned} \tag{B.2}$$

where

- Pr(*Struct(UArg)*|*Struct(SysInt)*) is the probability that the user presented an argument of structure *Struct(UArg)* when s/he intended *Struct(SysInt)*, and
- Pr(*Bel(UArg)*|*Bel(SysInt)*) is the probability that the user stated the beliefs in *UArg* when s/he intended the beliefs in *SysInt*.

These probabilities are calculated as follows.

B.1. CALCULATING Pr(*Bel(UArg)*|*Bel(SysInt)*)

Bel(UArg) comprises the beliefs stated by the user with respect to the nodes in *UArg*. That is, $Bel(UArg) = \{Bel(Nd_1, UArg), \ldots, Bel(Nd_{n_{UA}}, UArg)\}$, where n_{UA} is the number of nodes in *UArg* ($n_{UA} \leq n_{SI}$, which is the number of nodes in

SysInt). Similarly, $Bel(SysInt)$ comprises the beliefs in nodes $Nd_1, \ldots, Nd_{n_{UA}}$ in *SysInt*, which are obtained by means of Bayesian propagation (nodes that appear only in *SysInt* are handled by the component which describes structural differences, Appendix B.2). Thus,

$$\Pr(Bel(UArg)|Bel(SysInt)) =$$
$$\Pr(Bel(Nd_1,UArg),\ldots,Bel(Nd_{n_{UA}},UArg)|Bel(Nd_1,SysInt),\ldots,Bel(Nd_{n_{UA}},SysInt))$$

We now make the simplifying assumption that given $Bel(Nd_i,SysInt)$, for $i, j = 1, \ldots, n_{UA}$ and $j \neq i$, $Bel(Nd_i,UArg)$ is conditionally independent of $Bel(Nd_j,UArg)$ and $Bel(Nd_j, SysInt)$. This assumption is not generally correct, as the belief in a node in an argument depends on the beliefs in other nodes in the argument. However, what we are assessing here is how likely the user is to state a particular belief in a node in his/her argument, when s/he intended the belief in the interpretation. The application of Bayes rule yields the following equation.

$$\Pr(Bel(UArg)|Bel(SysInt)) = \prod_{i=1}^{n_{UA}} \Pr(Bel(Nd_i,UArg)|Bel(Nd_i,SysInt)) \quad \text{(B.3)}$$

This equation represents the basic formalism for calculating the probability of the beliefs in an argument given the beliefs in an interpretation. However, since our system interacts with people, we discretize beliefs to fit seven linguistic categories of probability that people find acceptable (similar to those used in (Elsaesser, 1987)). As stated in Section 2.1, our categories are: {VeryUnlikely, Unlikely, ALittleUnlikely, EvenChance, ALittleLikely, Likely, VeryLikely}, numbered {1, 2, 3, 4, 5, 6, 7} respectively. This yields the following approximation of Equation B.3.

$$\Pr(Bel(UArg)|Bel(SysInt)) \cong \prod_{i=1}^{n_{UA}} \Pr(BelCat(Nd_i,UArg)|BelCat(Nd_i,SysInt)) \quad \text{(B.4)}$$

where $BelCat(Nd_i,UArg)$ and $BelCat(Nd_i,SysInt)$ are the categories for the belief in node Nd_i according to *UArg* and according to *SysInt* respectively.

We use the Zipf probability distribution to model the discrepancies between a user's beliefs and the system's beliefs, where the parameter of the distribution is the absolute value of the difference between the belief category of the user's belief in node Nd_i and that of the system's belief in this node.

$$difCat(Nd_i) = 1 + |BelCat(Nd_i, UArg) - BelCat(Nd_i, SysInt)|$$

This yields

$$\Pr(BelCat(Nd_i, UArg)|BelCat(Nd_i, SysInt)) = \frac{\theta}{difCat(Nd_i)^{\gamma}} \quad \text{(B.5)}$$

where γ is normally a low value (we have selected $\gamma = 2$), θ is a normalizing constant, and $1 \leq difCat \leq 7$ (the number of categories).

This distribution penalizes (assigns a small probability to) large differences between the beliefs in a user's argument and those inferred by BIAS. If a user's belief in a node matches the system's belief, this distribution yields the maximum probability θ. In contrast, if the user's belief is at one end of the spectrum and the system's belief is at the other end, the distribution yields $\frac{\theta}{7^2} = \frac{\theta}{49}$. We decided to impose such heavy penalties for discrepancies in belief as a result of our preliminary trials with users, who strongly objected to interpretations that contain beliefs which differ from the users' stated beliefs.

Substituting Equation B.5 in Equation B.4 we obtain

$$\Pr(Bel(UArg)|Bel(SysInt)) \cong \prod_{i=1}^{n_{UA}} \frac{\theta}{difCat(Nd_i)^{\gamma}} \tag{B.6}$$

B.2. CALCULATING $\Pr(Struct(UArg)|Struct(SysInt))$

Struct(UArg) represents the structure of a user's argument, i.e., its nodes and arcs, and *Struct(SysInt)* represents the structure of an interpretation. Since interpretations are generated to include all the nodes in a user's argument, the nodes in *UArg* are a subset of the nodes in *SysInt*, but the arcs in *UArg* may differ from those in *SysInt*.[18] That is, the user's argument may contain some arcs that are derivable from *SysInt* as well as arcs that are extraneous to *SysInt*. Hence, the calculation of $\Pr(Struct(UArg)|Struct(SysInt))$ resembles the calculation of $\Pr(SysInt)$ in Equation A.1, but distinguishes between arcs that are selected from *SysInt* and arcs that are newly inserted. These arcs are designated as follows.

- $Arcs_{\text{sel}}$ – the set of arcs in *UArg* selected from *SysInt*,
- $Arcs_{\text{ins}}$ – the set of newly inserted arcs in *UArg* (i.e., arcs that cannot be obtained from *SysInt*),
- a_{sel} – the number of selected arcs, i.e., $|Arcs_{sel}|$,
- a_{ins} – the number of inserted arcs, i.e., $|Arcs_{ins}|$.

This results in the following definition for $\Pr(Struct(UArg)|Struct(SysInt))$:

$$\begin{aligned}&\Pr(Struct(UArg)|Struct(SysInt))\\ &\quad = \Pr(Arcs_{\text{sel}}, Arcs_{\text{ins}}, a_{\text{sel}}, a_{\text{ins}}, Nodes_{UA}, n_{UA}|Struct(SysInt))\end{aligned}$$

where $Nodes_{UA}$ designates the nodes in *UArg*, and n_{UA} is the number of nodes in *UArg*.

[18] As indicated before, the NL version of our system allows users to state propositions unknown to the system. Clearly, nodes corresponding to these propositions cannot be extracted from *SysInt* and are taken into account by a different mechanism.

Applying the chain rule of probability theory yields

$$\begin{aligned}
&\Pr(Struct(UArg)|Struct(SysInt)) \\
&\quad = \Pr(Arcs_{\text{sel}}|Arcs_{\text{ins}}, a_{\text{sel}}, a_{\text{ins}}, Nodes_{UA}, n_{UA}, SysInt) \\
&\qquad \times \Pr(Arcs_{\text{ins}}|a_{\text{sel}}, a_{\text{ins}}, Nodes_{UA}, n_{UA}, SysInt) \\
&\qquad \times \Pr(a_{\text{sel}}|a_{\text{ins}}, Nodes_{UA}, n_{UA}, SysInt) \times \Pr(a_{\text{ins}}|Nodes_{UA}, n_{UA}, SysInt) \\
&\qquad \times \Pr(Nodes_{UA}|n_{UA}, SysInt) \times \Pr(n_{UA}|SysInt)
\end{aligned} \tag{B.7}$$

We now make the following simplifying assumptions based on the conditional independence between selected and inserted arcs.

- Given $Nodes_{UA}$, n_{UA} and $SysInt$, a_{sel} is conditionally independent of a_{ins}.
- Given $Nodes_{UA}$, n_{UA}, $SysInt$ and a_{ins}, $Arcs_{\text{ins}}$ is conditionally independent of a_{sel}.
- Given $Nodes_{UA}$, n_{UA}, $SysInt$ and a_{sel}, $Arcs_{\text{sel}}$ is conditionally independent of $Arcs_{\text{ins}}$ and a_{ins}.

These assumptions yield the following formula.

$$\begin{aligned}
&\Pr(Struct(UArg)|Struct(SysInt)) \\
&\quad = \Pr(Arcs_{\text{sel}}|a_{\text{sel}}, Nodes_{UA}, n_{UA}, SysInt) \times \Pr(a_{\text{sel}}|Nodes_{UA}, n_{UA}, SysInt) \\
&\qquad \times \Pr(Arcs_{\text{ins}}|a_{\text{ins}}, Nodes_{UA}, n_{UA}, SysInt) \times \Pr(a_{\text{ins}}|Nodes_{UA}, n_{UA}, SysInt) \\
&\qquad \times \Pr(Nodes_{UA}|n_{UA}, SysInt) \times \Pr(n_{UA}|SysInt)
\end{aligned} \tag{B.8}$$

These probabilities are calculated as follows.

- $\Pr(n_{UA}|SysInt)$ – Our preliminary experiments (Zukerman et al., 2003a) show that users' arguments typically contain about half the nodes in their interpretation ($\frac{n_{SI}}{2}$), while BIAS fills in the other half. Hence, as done in Equation A.3, we model the nodes in an argument using a truncated Poisson distribution as follows.

$$\Pr(n_{UA}|SysInt) = \begin{cases} \phi \dfrac{e^{-n_{SI}/2}(\frac{n_{SI}}{2})^{n_{UA}}}{n_{UA}!} & \text{if } n_{UA} \leq n_{SI} \\ 0 & \text{otherwise} \end{cases} \tag{B.9}$$

  where ϕ is a normalizing constant.

- $\Pr(Nodes_{UA}|n_{UA}, SysInt)$ – We want to select the n_{UA} nodes in $Nodes_{UA}$ from the n_{SI} nodes in $SysInt$. Now, a feature of the interpretations generated by procedure *GenerateInterpretations* (Section 4) is that all their leaf nodes (i.e., nodes with one arc only, which may also include the goal node) are in $UArg$. Hence, in order to express $Nodes_{UA}$ in terms of $SysInt$, we need to consider only the nodes in $UArg$ that are *not* leaf nodes in $SysInt$ (the leaf nodes can be obtained

algorithmically from *SysInt*). Assuming a uniform distribution of node configurations among the non-leaf nodes in *SysInt*, there are $\binom{n_{SI}-nleaf_{SI}}{n_{UA}-nleaf_{SI}}$ ways to select the non-leaf nodes in *UArg* from those in *SysInt*, where $nleaf_{SI}$ is the number of leaf nodes in *SysInt*. Thus, the probability of any particular configuration of n_{UA} nodes is equivalent to the probability of the $n_{UA} - nleaf_{SI}$ non-leaf nodes in this configuration. This probability is calculated as follows:

$$\Pr(Nodes_{UA}|n_{UA}, SysInt) = \frac{1}{\binom{n_{SI}-nleaf_{SI}}{n_{UA}-nleaf_{SI}}} \tag{B.10}$$

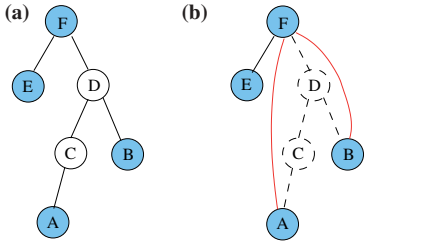
– $\Pr(a_{\text{sel}}|Nodes_{UA}, n_{UA}, SysInt)$ – *UArg* cannot contain arcs that are incident upon nodes that are *not* in $Nodes_{UA}$. In order not to leave dangling arcs between nodes in *UArg* and nodes that are in *SysInt* but not in *UArg*, we iteratively redirect each arc in *SysInt* that connects between a *UArg* node and a non-*UArg* node, so that it connects between the *UArg* node and the parent of the non-*UArg* node. This process is repeated until a *UArg* node is reached. Figure 14 illustrates this process with respect to the argument $A, B, E \Rightarrow F$. Figure 14(a) contains an interpretation *SysInt* (the shaded nodes are those mentioned by the user), and Figure 14(b) contains *SysInt* with redirected arcs (the unselected nodes and dangling arcs are dashed). Upon completion of this process, we have an intermediate graph whose nodes are those in *UArg* ($Nodes_{UA}$) and whose *updated arcs* include the redirected arcs in *SysInt* (plus arcs previously in *SysInt* that were not redirected). In fact, it is often the case that the resulting structure has exactly the arcs in *UArg*. Let us denote these updated arcs $UpdatedArcs_{SI}$, and let the number of updated arcs be ua_{SI}. Now, as for Equation A.5, we use a uniform distribution between $n_{UA}-1$ (the minimum number of arcs that connect n_{UA} nodes) and ua_{SI} to model the probability of selecting a_{sel} arcs. This yields

$$\Pr(a_{\text{sel}}|Nodes_{UA}, n_{UA}, SysInt) = \frac{1}{ua_{SI} - (n_{UA} - 1) + 1} \tag{B.11}$$

– $\Pr(a_{\text{ins}}|Nodes_{UA}, n_{UA}, SysInt)$ – The maximum number of arcs in a graph of n_{UA} nodes is $\frac{1}{2}n_{UA}(n_{UA} - 1)$. Since only arcs that don't exist in *SysInt* can be inserted, the maximum possible number of arc insertions is $\frac{1}{2}n_{UA}(n_{UA} - 1) - ua_{SI}$. We therefore use the following truncated Poisson distribution to model arc insertions:

$$\Pr(a_{\text{ins}}|Nodes_{UA}, n_{UA}, SysInt) = \begin{cases} \lambda \frac{e^{-\mu}\mu^{a_{\text{ins}}}}{a_{\text{ins}}!} & \text{if } a_{\text{ins}} \leq \frac{1}{2}n_{UA}(n_{UA} - 1) - ua_{SI} \\ 0 & \text{otherwise} \end{cases} \tag{B.12}$$

where μ is the mean of the distribution, and λ is a normalizing constant. Our preliminary investigations show that good interpretations have only a few arc insertions (Zukerman et al., 2003a). Hence, we use $\mu = 1$ to penalize interpretations with many arc insertions.



SysInt *SysInt* with redirected arcs

Figure 14. Selecting *UArg* from *SysInt*.

– $\Pr(Arcs_{\text{sel}}|a_{\text{sel}}, Nodes_{UA}, a_{UA}, n_{UA}, SysInt)$ – As for Equation A.6, we select a_{sel} arcs from ua_{SI} updated arcs. This yields

$$\Pr(Arcs_{\text{sel}}|a_{\text{sel}}, Nodes_{UA}, a_{UA}, n_{UA}, SysInt) = \frac{1}{\binom{ua_{SI}}{a_{\text{sel}}}} \tag{B.13}$$

– $\Pr(Arcs_{\text{ins}}|a_{\text{ins}}, Nodes_{UA}, a_{UA}, n_{UA}, SysInt)$ – Here we select a_{ins} arcs from the maximum possible number of arc insertions, which is $\frac{1}{2}n_{UA}(n_{UA}-1) - ua_{SI}$. This yields

$$\Pr(Arcs_{\text{ins}}|a_{\text{ins}}, Nodes_{UA}, a_{UA}, n_{UA}, SysInt) = \frac{1}{\binom{\frac{1}{2}n_{UA}(n_{UA}-1)-ua_{SI}}{a_{\text{ins}}}} \tag{B.14}$$

C. Calculating the Probability of an Interpretation in the Context of a User Model

Let m_k be a random variable whose value is 1 if node Nd_k is included in *SysInt*, and 0 otherwise. We posit that the probability of including a node in an interpretation depends on its salience in the user's focus of attention. That is, the user is more likely to intend salient nodes than nodes that are not salient. Thus, $\Pr(m_k)$, the probability that $m_k = 1$, is derived from $Score(Nd_k)$ as follows.

$$\Pr(m_k) = \frac{Score(Nd_k) + GC}{\sum_{i=1}^{N}[Score(Nd_i) + GC]} \tag{C.1}$$

where $Score(Nd_k)$ is obtained from Equation 13, and GC is a small number that corresponds to Good's *flattening constant* (Good, 1965).[19] This flattening constant was added to the scores so that nodes with a score of 0 still have some probability of being included in *SysInt*.

[19] We have chosen a constant of $\frac{1}{2N}$, which is half the prior probability of selecting a node in *dBN*, as this number is consistent with the constants obtained by the Minimum Message Length theory (Wallace and Boulton, 1968; Wallace, 2005). Also note that the denominator in Equation C.1 can be replaced by a normalizing constant.

Thus, interpretations that include nodes that were never accessed by the user (and are dissimilar to other nodes) will have a low probability, while interpretations that include nodes that have been repeatedly accessed (and are similar to other nodes) will have a higher probability. This is in contrast to the equiprobable node distribution assumed in our simple model when calculating Pr(*SysInt*) (Equation A.4).

In order to incorporate Equation (C.1) into our calculation of Pr(*SysInt*), we define a multinomial random variable $(m_1, \ldots, m_N)$, where each dimension corresponds to the inclusion of a node from *dBN* in *SysInt*, and $\sum_{k=1}^{N} m_k = n_{SI}$. That is, n_{SI} nodes of the N nodes in *dBN* are in *SysInt*. The probability distribution of this variable is defined as follows.

$$\Pr(Nodes_{SI} | n_{SI}, dBN) = \Pr(m_1, \ldots, m_N) = n_{SI}! \prod_{k=1}^{N} \frac{\Pr(m_k)^{m_k}}{m_k!}$$

Since $m_k \in \{0, 1\}$ for $k = 1, \ldots, N$, we can cancel the denominator. In addition, for $m_k = 0$, $\Pr(m_k)^{m_k} = 1$. Hence, we multiply only the probabilities for $m_k = 1$.

$$\Pr(Nodes_{SI} | n_{SI}, dBN) = \Pr(m_1, \ldots, m_N) = n_{SI}! \prod_{\substack{k=1 \\ \forall m_k = 1}}^{N} \Pr(m_k) \tag{C.2}$$

Notice, however, that multinomial distributions demand that the components of the random variable be independent, and this is not the case for the selection of the nodes in an interpretation (as the selection is done without replacement). In order to model this discrepancy, we reduce the state space after a node has been selected (to ensure consistent results, node selection is performed in a pre-determined order – the alphanumeric order of the node names in the system). For instance, consider a small interpretation containing two nodes Nd_i and Nd_j, with probabilities $\Pr_i$ and $\Pr_j$ respectively. After Nd_i has been selected (with probability $\Pr_i$), the probability mass left to be allocated is $1 - \Pr_i$, thus the probability of selecting Nd_j is $\frac{\Pr_j}{1-\Pr_i}$. In general, $\Pr'(m_k)$, the adjusted probability of including Nd_k in *SysInt*, is given by

$$\Pr'(m_k) = \frac{\Pr(m_k)}{\left\{1 - \sum_{\substack{j=1 \\ \forall m_j = 1}}^{k-1} \Pr(m_j)\right\}}$$

which when incorporated into Equation (C.2) yields

$$\Pr(Nodes_{SI} | n_{SI}, dBN) = \Pr'(m_1, \ldots, m_N) = n_{SI}! \prod_{\substack{k=1 \\ \forall m_k = 1}}^{N} \frac{\Pr(m_k)}{\left\{1 - \sum_{\substack{j=1 \\ \forall m_j = 1}}^{k-1} \Pr(m_j)\right\}} \tag{C.3}$$

Note that this formula yields Equation (A.4) for an equiprobable distribution of including a node in *SysInt*.

References

Allen, J. F. and Perrault, C. R.: 1980, Analyzing intention in utterances. *Artificial Intelligence* **15**(3), 143–178.

Bontcheva, K. and Wilks, Y.: this issue, Tailoring automatically generated hypertext. *User Modeling and User-Adapted Interaction.*

Bull, S. and Pain, H.: 1995, Did I say what I think I said, and do you agree with me?: inspecting and questioning the student model. In: *Proceedings of the World Conference on Artificial Intelligence in Education, Association for the Advancement of Computing in Education (AACE)*. Charlottesville, Virginia, pp. 501–508.

Carberry, S. and Lambert, L.: 1999, A process model for recognizing communicative acts and modeling negotiation subdialogues. *Computational Linguistics* **25**(1), 1–53.

Charniak, E. and Goldman, R. P.: 1993, A Bayesian model of plan recognition. *Artificial Intelligence* **64**(1), 53–79.

Cohen, R.: 1987, Analyzing the structure of argumentative discourse. *Computational Linguistics* **13**(1), 11–24.

Dean, T. and Boddy, M. S.: 1988, An analysis of time-dependent planning. In: *AAAI-88 – Proceedings of the Seventh National Conference on Artificial Intelligence*. St. Paul, Minnesota, pp. 49–54.

Elsaesser, C.: 1987, Explanation of probabilistic inference for decision support systems. In: *Proceedings of the AAAI-87 Workshop on Uncertainty in Artificial Intelligence*. Seattle, Washington, pp. 394–403.

Epstein, M. E.: 1996, Statistical Source Channel Models for Natural Language Understanding. Ph.D. thesis, Department of Computer Science, New York University, New York, New York.

Gertner, A., Conati, C. and VanLehn, K.: 1998, Procedural help in Andes: Generating hints using a Bayesian network student model. In: *AAAI98 – Proceedings of the Fifteenth National Conference on Artificial Intelligence*. Madison, Wisconsin, pp. 106–111.

Good, I. J.: 1965, *The Estimation of Probabilities: An Essay on Modern Bayesian Methods*, Research Monograph No. 30. Cambridge, Massachusetts: MIT Press.

Goodman, B. A., Linton, F. N., Gaimari, R. D., Hitzeman, J. M., Ross, H. J., and Zarrela, G.: this issue, Using dialogue features to predict trouble during collaborative learning. *User Modeling and User-Adapted Interaction.*

Horvitz, E. and Paek, T.: 1999, A computational architecture for conversation. In: *UM99 – Proceedings of the Seventh International Conference on User Modeling*. Banff, Canada, pp. 201–210.

Jensen, F. V.: 1996, *An Introduction to Bayesian Networks*. UCL Press, London, United Kingdom.

Jitnah, N., Zukerman, I., McConachy, R., and George, S.: 2000, Towards the generation of rebuttals in a Bayesian argumentation system. In: *Proceedings of the First International Natural Language Generation Conference*. Mitzpe Ramon, Israel, pp. 39–46.

Kashihara, A., Hirashima, T., and Toyoda, T.: 1995, A cognitive load application in tutoring. *User Modeling and User-Adapted Interaction* **4**(4), 279–303.

Kay, J.: 1999, A Scrutable user Modelling Shell for User-adapted Interaction. Ph.D. thesis, Basser Department of Computer Science, Sydney, Australia.

Kintsch, W.: 1994, Text comprehension, memory and learning. *American Psychologist* **49**(4), 294–303.

Litman, D. and Allen, J. F.: 1987, A plan recognition model for subdialogues in conversation. *Cognitive Science* **11**(2), 163–200.

McConachy, R., Korb, K. B., and Zukerman, I.: 1998, Deciding what not to say: An attentional-probabilistic approach to argument presentation. In: *Proceedings of the Twentieth Annual Conference of the Cognitive Science Society*. Madison, Wisconsin, pp. 669–674.

Miller, G., Beckwith, R., Fellbaum, C., Gross, D., and Miller, K.: 1990, Introduction to wordNet: An on-line lexical database. *Journal of Lexicography* **3**(4), 235–244.

Pearl, J.: 1988, *Probabilistic Reasoning in Intelligent Systems*. San Mateo, California: Morgan Kaufmann Publishers.

Quilici, A.: 1992, Arguing about planning alternatives. In: *COLING-92 – Proceedings of the Fourteenth International Conference on Computational Linguistics*. Nantes, France, pp. 906–910.

Raskutti, B. and Zukerman, I.: 1991, Generation and selection of likely interpretations during plan recognition. *User Modeling and User Adapted Interaction* **1**(4), 323–353.

Reed, C. and Walton, D.: 2003, Argumentation schemes in argument-as-process and argument-as-product. In: *Proceedings of the Conference Celebrating Informal Logic @25*. Windsor, Canada.

Restificar, A., Syed, A., and McRoy, S.: 1999, ARGUER: Using argument schemas for argument detection and rebuttal in dialogs. In: *UM99 – Proceedings of the Seventh International Conference on User Modeling*. Banff, Canada, pp. 315–317.

Toulmin, S.: 1958, *Uses of Argument*. Cambridge: Cambridge University Press.

Wallace, C.: 2005, *Statistical and Inductive Inference by Minimum Message Length*. Springer, Berlin, Germany.

Wallace, C. and Boulton, D.: 1968, An information measure for classification. *The Computer Journal* **11**(2), 185–194.

Walton, D. N.: 1996, *Argumentation Schemes for Presumptive Reasoning*. Lawrence Erlbaum Associates, Mahwah, New Jersey.

Zukerman, I.: 2001, An integrated approach for generating arguments and rebuttals and understanding rejoinders. In: *UM01 – Proceedings of the Eighth International Conference on User Modeling*. Sonthofen, Germany, pp. 84–94.

Zukerman, I. and George, S.: 2002, Towards a noise-tolerant, representation-independent mechanism for argument interpretation. In: *COLING 2002 – Proceedings of the 19th International Conference on Computational Linguistics*. Taipei, Taiwan, pp. 1170–1176.

Zukerman, I., George, S., and George, M.: 2003a, Incorporating a user model into an information theoretic framework for argument interpretation. In: *UM03 – Proceedings of the Ninth International Conference on User Modeling*. Johnstown, Pennsylvania, pp. 106–116.

Zukerman, I., George, S., and Wen, Y.: 2003b, Lexical paraphrasing for document retrieval and node identification. In: *IWP2003 – Proceedings of the Second International Workshop on Paraphrasing: Paraphrase Acquisition and Applications*. Sapporo, Japan, pp. 94–101.

Zukerman, I., Niemann, M., and George, S.: 2004, Improving the presentation of argument interpretations based on user trials. In: *AI'04 – Proceedings of the 17th Australian Joint Conference on Artificial Intelligence*. Cairns, Australia, pp. 587–598.

Authors' vitae

Ingrid Zukerman

School of Computer Science and Software Engineering, Monash University, Clayton, Victoria 3800, Australia

Ingrid Zukerman is an Associate Professor in Computer Science at Monash University. She received her B.Sc. degree in Industrial Engineering and Management and her M.Sc. degree in Operations Research from the Technion – Israel Institute of Technology. She received her Ph.D. degree in Computer Science from UCLA in 1986. Her areas of interest are discourse planning, plan recognition, question answering, and agent modeling.

Sarah George

School of Computer Science and Software Engineering, Monash University, Clayton, Victoria 3800, Australia

Sarah George completed a Computer Science degree with Honours at Monash University, where she has also been working throughout most of her studies. She currently works as a Research Fellow for Ingrid Zukerman. She has done programming work in a variety of contexts including financial, engineering, operating systems, and 3D modelling. She is a 'programming generalist' interested in systems, abstraction and language. Her curiosity about language as knowledge representation has led her to her current work in user modelling and natural language.