User Model User-Adap Inter (2007) 17:439–474
DOI 10.1007/s11257-007-9034-9

ORIGINAL PAPER

Inferences, suppositions and explanatory extensions in argument interpretation

Sarah George · Ingrid Zukerman · Michael Niemann

Received: 17 December 2006 / Accepted in revised form: 21 February 2007 /
Published online: 17 July 2007

Abstract We describe a probabilistic approach for the interpretation of user arguments that integrates three aspects of an interpretation: inferences, suppositions and explanatory extensions. Inferences fill in information that connects the propositions in a user's argument, suppositions postulate new information that is likely believed by the user and is necessary to make sense of his or her argument, and explanatory extensions postulate information the user may have implicitly considered when constructing his or her argument. 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. Our evaluations show that suppositions and explanatory extensions are necessary components of interpretations, and that users consider appropriate the suppositions and explanatory extensions postulated by our system.

Keywords Discourse interpretation · Suppositions · Explanatory extensions · Probabilistic approach · Bayesian networks

---

This article integrates and extends research described in George et al., 2004; Zukerman et al., 2004; Zukerman and George, 2005; George et al., 2005. The research described in this article was conducted while Sarah George was employed at Monash University and was supported in part by the ARC Centre for Perceptive and Intelligent Machines in Complex Environments .

S. George · I. Zukerman (✉) · M. Niemann
Faculty of Information Technology, Monash University,
Clayton VIC, 3800, Australia
e-mail: ingrid@csse.monash.edu.au

S. George
CVS Dude, 80 Jephson St., Toowong, QLD 4066, Australia
e-mail: sarahg@developingprogrammers.com

M. Niemann
e-mail: niemann@csse.monash.edu.au

1 Introduction

When generating discourse, people often rely on information without stating it explicitly. They may skip steps in a chain of reasoning, take into account information without mentioning it, or suppose facts that are not necessarily in evidence. Failure to identify these unstated information items may lead to mis-communication. For instance, the implication "Only the blue car is in the driveway, therefore Mary is probably out" would be rather unintelligible, unless the addressee knew that the blue car wasn't Mary's. Likewise, the implication "Jack is tall, so Jill must be tall" supposes that Jack and Jill are related. This supposition must be taken into account in order to respond cooperatively, e.g., saying "Actually, Jack and Jill are not related" rather than "I don't think so, Jill may or may not be tall".

In previous work, we introduced a probabilistic approach to argument interpretation that was implemented in a system called BIAS (*Bayesian Interactive Argumentation System*). BIAS receives as input an argument for a goal proposition, and generates one or more *interpretations* that consist of propositions associated with degrees of belief, and relations between propositions. For example, if a user in Melbourne said "If I walk to the main road, then I'll probably be in Sydney tomorrow", one possible interpretation would be "*WalkMainRoad* → ***TakeBus*** → *ArriveSydney* [Probably]", and another would be "*WalkMainRoad* → ***HitchRide*** → *ArriveSydney* [Probably]".

Our approach starts out from the tenet that an interpretation is a representation of what an interlocutor said in terms of the mental model maintained by the addressee. When the addressee is a computer, this representation is constrained by the knowledge representation employed by the system—in our case a Bayesian network (BN) (Pearl 1988). Thus, the objective of our research is to obtain an interpretation of a user's argument in terms of BIAS' underlying domain BN (rather than constructing this BN). In our initial work, we defined an interpretation as a subnet of this BN, and applied a probabilistic approach to select a subnet that fits the user's argument well. The main contributions of that research were: (1) an anytime algorithm for proposing candidate subnets of the domain BN (George et al. 2004); and (2) a formalism for calculating the probability of these subnets—this probability encodes how probable are the structure and beliefs of these subnets in the context of the domain BN (Zukerman et al. 2003; Zukerman and George 2005). Our formalism is applicable to a variety of BNs (which is orthogonal to how well the BNs model their domain).

In this article, we extend our previous definition of an interpretation by incorporating *suppositions* and *explanatory extensions*. Suppositions postulate beliefs held by the user which are not consistent with the beliefs in the user model. Explanatory extensions present information that was omitted from the user's argument, but appears in the user model and was likely considered by the user when constructing his or her argument. To illustrate these concepts, imagine that our Melbournian user had said "If I walk to the main road, I'll have to hitch a ride to Sydney". Now, if the user had previously been bemoaning a current bus strike, the system would use this shared but unstated information to explain the user's conclusion, yielding an interpretation such as "You'll have to hitch a ride to Sydney, *because of the bus strike you mentioned*". However, if the strike (or other reasons for hitching a ride) had never been mentioned, in order to explain the consequent, the system would have to postulate that the user believes there is a reason for hitching a ride (e.g., buses are too slow or there is a bus strike), and *suppose* the most probable of these reasons according to the user model. This supposition would yield an interpretation such as "You say that you'll have to hitch a ride to Sydney. *Are you supposing that there is a bus strike?*". Explanatory extensions are added to interpretations to improve their coherence, while suppositions are included in the user model and added to interpretations to justify a user's stated beliefs. The incorporation of these components in an interpretation signifies a shift in paradigm from our earlier work, in that we now view an interpretation as an explanation, rather than a representation, of what the user said.

The main contributions of this article are: (1) an extension of the definition of an interpretation that includes suppositions and explanatory extensions, (2) an extension of the formalism for calculating the probability of an interpretation, and (3) procedures for postulating suppositions and for including explanatory extensions in an interpretation.

In the next section, we describe our domain of implementation, knowledge representation and user model. In Sect. 3, we define an interpretation in terms of our knowledge representation, and outline our interpretation-generation process. Section 4 describes our mechanism for positing suppositions, followed by its evaluation. Section 6 describes our procedure for including explanatory extensions in an interpretation, also followed by its evaluation. In Sect. 8, we demonstrate the generalizability of our approach by applying it to a BN downloaded from the Netica website (www.norsys.com). We then discuss related research, and present concluding remarks, focusing on the limitations and contributions of our approach.

2 Domain and user model

It is widely accepted that people are not Bayesian (Kahneman et al. 1982). Nonetheless, we have chosen BNs (Pearl 1988) to represent and reason about BIAS' domain knowledge, due to their ability to perform causal and evidential reasoning under uncertainty, which is essential for argumentation. An interpretation of a user's argument is a mapping of this argument into BIAS' domain knowledge. Our formalism for selecting an interpretation is also probabilistic (it selects the interpretation with the highest posterior probability), but it is not represented as a BN.

Our domain of implementation is a murder mystery, for which we developed a series of BNs (Zukerman et al. 2003; Zukerman and George 2005). The examples in this paper are drawn from two similar 32-node binary murder-mystery BNs.[1] Each node in a binary BN may be set to True or False, or remain unset (with a probability between 0 and 1 inferred by Bayesian propagation).

Our BNs are merely a test platform used to demonstrate the applicability of our interpretation formalism. They represent a generic "story" that we made up, from which many specific scenarios can be generated. The user discovers the details of a scenario as he or she explores our web interface (Zukerman and George 2005). Figure 1 shows one of the BNs used for our examples. The observable evidence nodes are boxed (a specific configuration of values for evidence nodes yields a particular scenario). In our example, five of these evidence nodes have been observed so far: NbourSawGreenAroundGardenAt11, FingerprintsFoundOnGun, ForensicsMatchGreen'sFingerprints, GunFoundInGarden and BulletsFoundInBody'sBody (boldfaced and shaded). With this evidence in hand, the user could construct several arguments. Examples are: (1) "Since the neighbour saw Mr Green around the garden at 11 and the gun was found in the garden, it is probable that Mr Green had the means to kill Mr Body", (2) "The gun being found in the garden but forensics not matching Mr Green's fingerprints with those on the gun indicates that Mr Green possibly did not murder Mr Body", and (3) "The fact that bullets were found in Mr Body's body and a gun was found in the garden implies that Mr Body was murdered". The white boldfaced nodes in the BN (GreenInGardenAtTime

[1] The size of these BNs is comparable to that of many BNs obtainable from the Netica web site. Specifically, the average size of the "real-life" BNs is 17 nodes, and that of "large and complex" BNs is 62 nodes.

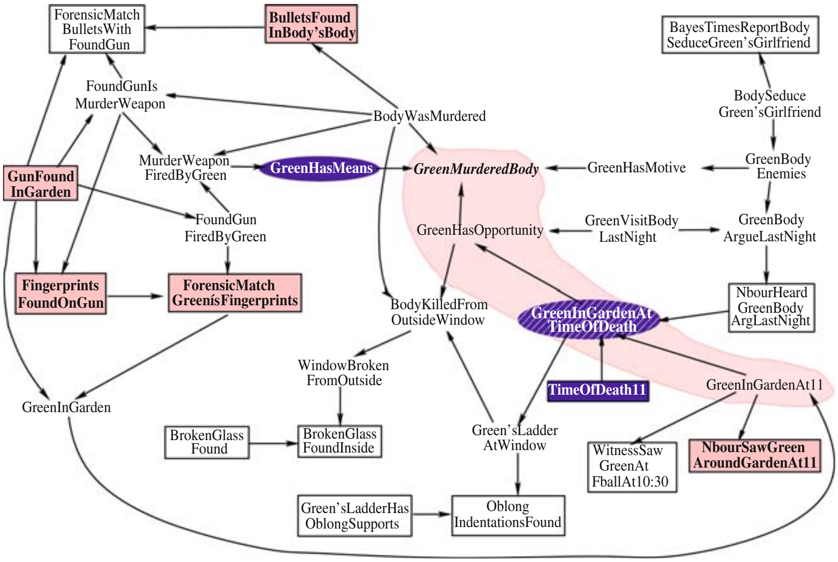


Fig. 1 Domain BN and interpretation of a user's argument

OfDeath, GreenHasMeans and TimeOfDeath11) pertain to the interpretation of the argument in Fig. 2 (Sect. 3). TimeOfDeath11 is an evidence node that has not been observed yet, while GreenInGardenAtTimeOfDeath and GreenHasMeans contain intermediate hypotheses about the murder case at hand.

Our BNs are configurable in the sense that different nodes may be set or unset as evidence nodes. As stated above, this enables us to generate a large number of scenarios, which provide the evidence that the user encounters as he or she investigates the murder mystery. For example, the BN in Fig. 1 has a node about the time of death being 11, which may be set to True or False. Similarly, the neighbour may or may not have seen Mr Green in the garden, Mr Green's fingerprints may or may not have been found on the gun, etc. Further, as shown in Sect. 8, BIAS may be applied to other binary BNs.

BIAS was originally designed as a computer game, where users are expected to enter arguments about the guilt or innocence of a suspect (Zukerman and George 2005). Our current research focuses on the interpretation capabilities of the system. Hence, a user's interaction with the system consists of (1) reading a system-generated "police report", which provides background information for the murder mystery (the report may contain preliminary eyewitness reports, forensic information and findings obtained from the scene of the crime); (2) optionally exploring a virtual murder scenario, where the user discovers evidence for our murder mystery; and (3) presenting an argument for the guilt or innocence of Mr Green. This argument, which is presented through a web interface, is then interpreted by the system.[2]

The first two steps provide information to populate our user model, which is consulted when interpreting the user's argument. In Zukerman and George (2005), we compared the performance obtained using several user models of increasing complexity: the *Simple* model

[2] In a previous implementation, we accepted free-form Natural Language input for the antecedents and consequents of an argument, which then had to be reconciled with the nodes in the domain BN. In the current implementation, this capability is replaced by a web-based interface, so that we can focus on suppositions and explanatory extensions.

records only the evidence accessed by the user, the *Access* model also takes into account the manner in which the evidence was accessed (e.g., seen or explicitly accepted) and how frequently and recently it was accessed, and the *Access&Similarity* model also considers whether this evidence could remind the user of other things. This information was used to calculate the probability of an interpretation. Our evaluation showed a trend whereby the interpretations produced by the *Access&Similarity* model were more acceptable to people than the interpretations produced by the simpler models. Hence, we use the *Access &Similarity* model in this paper.

3 What is an interpretation?

In the context of a BN, an interpretation of a user's argument is the tuple $\{SC, IG, EE\}$, where SC is a *supposition configuration*, IG is an *interpretation graph*, and EEs are *explanatory extensions*.

- A *Supposition Configuration* is a set of suppositions attributed to a user to account for the beliefs in his or her argument.
- An *Interpretation Graph* is a subnet of the domain BN which connects the nodes mentioned in the user's argument. The nodes and arcs that are included in an interpretation graph but were not mentioned by the user fill in additional detail in the context of the BN, bridging inferential gaps in the user's argument.
- *Explanatory Extensions* consist of subnets of the domain BN which are added to an interpretation graph to improve the coherence of the inferences in it.

Figure 2 illustrates these elements of an interpretation.

- The top segment shows the five pieces of evidence in the user model (boldfaced and boxed in light gray in Fig. 1). This evidence may be obtained from the police report or from the user's investigation of BIAS' virtual scenario. As indicated in Sect. 2, our user model assigns different strengths to propositions based on the manner in which the information in question was accessed by the user, e.g., evidence read in a police report is considered weaker than a fact stated by the user in his or her own argument (Zukerman and George 2005).
- The second segment contains a sample argument, which was generated using a menu in an argument-construction interface (Zukerman and George 2005). As seen in this segment, we continue using the seven linguistic categories of belief adopted in (Zukerman and George 2005). However, we have changed the wording associated with these categories, as our surveys indicated that people found wordings using "probable" and "possible" clearer than wordings using "likely" and "a little likely". Thus, our belief categories are: VeryProbably, Probably, Possibly, EvenChance, PossiblyNot, ProbablyNot and VeryProbablyNot.
- The third segment shows the interpretation produced by adding a node from the BN and its links to bridge a gap between two propositions in the user's argument (the proposition corresponding to the added node is boldfaced, and the node is white boldfaced and circled in striped dark gray in Fig. 1; the resultant interpretation graph is surrounded by a light gray bubble in Fig. 1). However, the beliefs in this interpretation do not match those of the argument consequents. This is because without knowing the time of death, there is no reason to believe that being in the garden at 11 implies opportunity.
- The fourth segment contains the supposition BIAS attributes to the user in order to account for the beliefs stated in the argument. This supposition, which is that the time of death is 11, brings the beliefs in the interpretation in line with those in the argument (the supposition

Evidence in the user model
Neighbour saw Mr Green around Mr Body's garden at 11.
Bullets were found in Mr Body's body.
A gun was found in the garden.
Fingerprints were found on the gun.
Forensics did not matcht he fingerprints on the gun with Mr Green.

Argument
Mr Green *probably* being in the garden at 11 implies that
he *possibly* had the opportunity to kill Mr Body, but
he *possibly* did *not* murder Mr Body.

Interpretation Graph: Bridging a gap in the argument
Mr Green *probably* being in the garden at 11
IMPLIES
Mr Green *probably* was *not* in the garden at the time of death
IMPLIES
Mr Green *probably* did *not* have the opportunity to kill Mr Body
IMPLIES
Mr Green *probably* did *not* murder Mr Body

Supposition Configuration: Making suppositions
Supposing that: **Time of death was 11**
Mr Green *probably* being in the garden at 11
IMPLIES
Mr Green *probably* was in the garden at the time of death
IMPLIES
Mr Green *possibly* had the opportunity to kill Mr Body
IMPLIES
Mr Green *possibly* did *not* murder Mr Body

Explanatory Extension: Incorporating unstated information
Supposing that: Time of death was 11
Mr Green *probably* being in the garden at 11
IMPLIES
Mr Green *probably* that was in the garden at the time of death
IMPLIES
Mr Green *possibly* had the opportunity to kill Mr Body
BUT Mr Green *probably* did *not* have the means to murder Mr Body
IMPLIES
Mr Green *possibly* did *not* murder Mr Body

Fig. 2 Bridging gaps in an argument, making suppositions and incorporating explanatory extensions

is boldfaced, and the corresponding node is white boldfaced and boxed in dark gray in Fig. 1). However, the interpretation is still somewhat problematic, as the belief in the final consequent is significantly lower than that in its antecedent (in our trials, people objected to such drops in belief, Sect. 6).

– The final segment illustrates how BIAS addresses this problem by including in the interpretation an explanatory extension about Mr Green not having the means to murder Mr Body, which explains the drop in belief (the explanatory extension is boldfaced, and the corresponding node is white boldfaced and circled in dark gray in Fig. 1; the belief in this node is inferred through Bayesian propagation from the forensic evidence).

3.1 Proposing interpretations

In this section, we discuss the generation of candidate interpretations for a user's argument, and the calculation of the probabilities of these interpretations. The probability of an interpretation depends on (1) how well it matches the underlying domain knowledge (its plausibility

Algorithm *GenerateInterpretations(UserArg)*
while {there is time}
{
1. Propose a supposition configuration *SC* that accounts for the beliefs stated in the argument. **(Section 4)**
   $SC \leftarrow$ *GetSuppositionConfig(UserArg)*
2. Propose an interpretation graph *IG* that connects the nodes in *UserArg* under supposition configuration *SC*. **(Zukerman and George, 2005)**
   $IG \leftarrow$ *GetInterpretationGraph(UserArg,SC)*
3. Propose explanatory extensions *EE* for interpretation graph *IG* under supposition configuration *SC*. **(Section 6)**
   $EE \leftarrow$ *GetExplanatoryExtensions(SC,IG)*
4. Calculate the probability of interpretation $\{SC, IG, EE\}$. **(Sections 3.2, 4.2 and 6.2)**
5. Retain the top *N* (=4) most probable interpretations.
}

Fig. 3 Anytime algorithm for generating interpretations

in the context of this knowledge); and (2) how well the user's argument matches the interpretation. That is, the particulars of the domain or the argument are not important. What is important is the similarity between an interpretation and the domain, and between the argument and the interpretation. The interpretations with the highest posterior probability are then selected for further consideration.[3]

The problem of finding the best interpretation is exponential. In Zukerman and George (2005), we presented an *anytime* algorithm (Dean and Boddy 1998; Horvitz et al. 1989) that generated only interpretation graphs, and a probabilistic formalism for selecting the best interpretation. Our algorithm was generalized in George et al. (2004), where we generated interpretations composed of node configurations (which contain BN nodes that match Natural Language sentences in an argument), supposition configurations, and interpretation graphs. In this article, we apply our anytime algorithm to generate interpretations comprising supposition configurations, interpretation graphs and explanatory extensions (as stated above, in our current implementation we obtain a user's input from a menu, and hence do not posit node configurations).

Algorithm ***GenerateInterpretations*** (Fig. 3) generates interpretations of the form $\{SC, IG, EE\}$ until it runs out of time. It first proposes a supposition configuration that accounts for the beliefs in a user's argument. Next, it generates an interpretation graph that connects the nodes in the argument. Each interpretation graph is complemented with explanatory extensions that improve the coherence of the inferences in the graph. The probability of the resultant interpretation is then calculated, and the top N (=4) interpretations are retained. This process is repeated until the algorithm runs out of time.

Figure 4a depicts a portion of the search tree generated by our algorithm. Each level of the tree corresponds to a different component. Supposition configurations are generated in the first level, and interpretation graphs in the second level. This order is motivated by the effect of supposition configurations on interpretation graphs, i.e., a supposition may block a path in a BN (precluding the propagation of evidence through this path), or unblock a previously blocked path (for a discussion of blocked paths, see Pearl, 1988). These interactions, which

[3] In a complete dialogue system, if there was a clear winner, the system would act on that interpretation, and if there were several good candidates, a clarification dialogue would ensue. However, at present, we just generate promising interpretations.

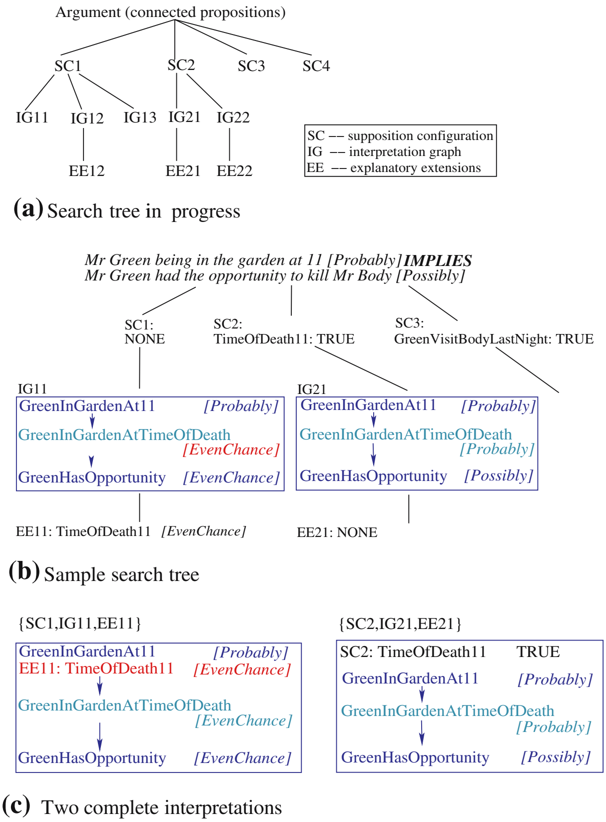


Fig. 4 Process for generating interpretation: (**a**) Search tree in progress, (**b**) Sample search tree, and (**c**) Two complete interpretations

are difficult to predict until an interpretation graph is complete, also motivate the large number of alternatives considered in the first two levels of the search tree. In contrast, explanatory extensions do not seem to have complex interactions with interpretation graphs or supposition configurations. Hence, they are deterministically generated in the third level of the search tree, i.e., only one set of explanatory extensions is proposed for each interpretation, rather than multiple options. Figure 4b depicts a portion of the search tree that was instantiated for the short argument at the root node of this tree: "*Mr Green probably being in the garden at 11 implies that Mr Green possibly had the opportunity to kill Mr Body*". Figure 4c shows two complete interpretations of this argument.

In this example, the user's belief in the consequent of this argument differs from the belief obtained by BIAS by means of Bayesian propagation (in the domain BN) from the evidence nodes observed by the user. As indicated above, BIAS attempts to address this problem by making suppositions (Sect. 4). However, if appropriate suppositions can not be posited or the resultant interpretation still has discrepancies between the user's beliefs and BIAS', BIAS just acknowledges these discrepancies. This is done by prefacing the interpretation with a sentence such as "I know this is not quite right, but it is the best I could do given what I believe about this situation". An alternative approach would consist of updating the system's inference patterns to match the user's. In the context of BNs, this involves modifying the

Conditional Probability Tables (CPTs) for the offending implications. However, this is a laborious process with far-reaching effects with respect to the system's reasoning. Hence, this type of solution is left for future investigation.

The first level of the sample search tree in Fig. 4b contains three supposition configurations $SC1$, $SC2$ and $SC3$ (only the beliefs that depart from those in the user model are shown): $SC1$ posits no beliefs that differ from those in the user model, thereby retaining the mismatch between the user's belief in the consequent and BIAS' belief; $SC2$ posits that the user supposes that the time of death is 11; and $SC3$ posits that the user supposes that Mr Green visited Mr Body last night.

$SC1$ yields interpretation graph $IG11$, where the belief in the consequent differs from that stated by the user (due to the absence of suppositions), prompting the generation of a preface that acknowledges this fact. In addition, the interpretation graph has a large jump in belief (from Probably to EvenChance), which causes BIAS to add the proposition TimeOfDeath11[EvenChance] as an explanatory extension. The resultant interpretation, which appears in Fig. 4c, may be glossed as follows (the explanatory extension appears in boldface italics).

I know this is not quite what you said, but it is the best I could do
given what I believe about this situation.
Since it is probable that Mr Green was in the garden at 11, and
***it is even chance that the time of death was 11**, it is even chance*
that Mr Green was in the garden at the time of death, which implies
that it is even chance that he had the opportunity to kill Mr Body.

$SC2$ yields interpretation graph $IG21$, which does not require explanatory extensions. This interpretation, which also appears in Fig. 4c, may be glossed as follows.

Your argument seems to suppose that the time of death was 11.
Hence, Mr Green probably being in the garden at 11 implies that
he probably was in the garden at the time of death, which implies
that he possibly had the opportunity to kill Mr Body.

Note that both interpretations mention TimeOfDeath11. However, in the first interpretation this proposition is used as an explanatory extension (with a belief of EvenChance obtained by Bayesian propagation), while in the second interpretation it is used as a supposition (with a belief of True).

Upon completion of this process, BIAS retains the four best interpretations (the evaluation of the goodness of an interpretation is described in Sect. 3.2). In this example, the winning interpretation is $\{SC2, IG21, EE21\}$.

3.2 Evaluating interpretations

An interpretation is evaluated by calculating its posterior probability. This is a "second order probability" in the sense that it encodes how probable are the structure and beliefs of an interpretation in the context of the domain BN. The best interpretation is that with the highest posterior probability.

$$SysIntBest = \text{argmax}_{i=1,\ldots,n} \Pr(SC_i, IG_i, EE_i | UserArg)$$

where n is the number of interpretations.

After applying Bayes rule, we obtain

$$SysIntBest = \text{argmax}_{i=1,\ldots,n}\{\Pr(UserArg|SC_i, IG_i, EE_i) \times \Pr(SC_i, IG_i, EE_i)\} \quad (1)$$

We now make the simplifying assumption that given SC_i and IG_i, *UserArg* is conditionally independent of EE_i, i.e.,

$$\Pr(UserArg|SC_i, IG_i, EE_i) = \Pr(UserArg|SC_i, IG_i) \quad (2)$$

One could argue that this assumption is not strictly correct, because the more explanatory extensions are postulated in an interpretation, the less likely it is that a user who intended this interpretation uttered *UserArg*. However, the presence of explanatory extensions is taken into account in Eq. 5 when calculating the probability of an interpretation. Hence, incorporating them in the calculation of the probability of *UserArg* would double-count them.

To calculate the prior probability of an interpretation, $\Pr(SC_i, IG_i, EE_i)$, we consider separately the structure and the beliefs in interpretation graph IG_i. This yields

$$\Pr(SC_i, IG_i, EE_i) = \Pr(SC_i, \text{beliefs in } IG_i, \text{struct of } IG_i, EE_i) \quad (3)$$

After applying the chain rule of probability we obtain

$$\begin{aligned} &\Pr(SC_i, \text{beliefs in } IG_i, \text{struct of } IG_i, EE_i) \quad (4)\\ &= \Pr(\text{beliefs in } IG_i|SC_i, \text{struct of } IG_i, EE_i) \times \Pr(EE_i|SC_i, \text{struct of } IG_i)\\ &\quad \times \Pr(\text{struct of } IG_i|SC_i) \times \Pr(SC_i) \end{aligned}$$

Substituting Eqs. 2, 3 and 4 in Eq. 1, we obtain

$$\begin{aligned} SysIntBest = \text{argmax}_{i=1,\ldots,n}&\{\Pr(UserArg|SC_i, IG_i)\\ &\times \Pr(\text{beliefs in } IG_i|SC_i, \text{struct of } IG_i, EE_i)\\ &\times \Pr(EE_i|SC_i, \text{struct of } IG_i)\\ &\times \Pr(\text{struct of } IG_i|SC_i) \times \Pr(SC_i)\} \quad (5) \end{aligned}$$

We now consider these factors from last to first (for ease of exposition).

- $\Pr(SC_i)$ depends on how close the suppositions in a configuration are to the current beliefs in the user model. The closer they are, the higher the probability of the supposition configuration. The calculation of $\Pr(SC_i)$ is described in Sect. 4.2.1.
- $\Pr(\text{struct of } IG_i|SC_i)$ is the probability of selecting the nodes and arcs in IG_i from the domain BN. The calculation of this probability is described in (Zukerman and George 2005). Here we present a brief summary. The simplest calculation implements ideas from combinatorics: given a BN with N nodes and an interpretation graph IG_i with n nodes, there are $\binom{N}{n}$ ways to choose the n nodes in IG_i. This probability is refined using the *Access&Similarity* user model, which stores whether the user previously encountered the nodes in IG_i or similar ones, and the type, frequency and recency of these encounters (Sect. 2). The probability of the arcs in IG_i is also calculated using combinatorics. It involves choosing the arcs in IG_i from the arcs in the BN that are incident on the nodes in IG_i.
  As mentioned above, the structure of IG_i depends on SC_i because a path may be blocked or unblocked by making suppositions. The probability of an interpretation graph that contains blocked paths is significantly reduced, as the resultant beliefs do not reflect the reasoning employed in the user's argument. This precludes the selection of such an interpretation graph, unless all available graphs have blocked paths.

– The probability $\Pr(EE_i|SC_i, \text{struct of } IG_i)$ has a structural component only. There is no belief component, as nodes in explanatory extensions do not provide additional evidence, and hence they do not affect the beliefs in a BN. They only affect the acceptability of an interpretation from the point of view of users (Sect. 6). $\Pr(EE_i|SC_i, \text{struct of } IG_i)$ depends on the number of nodes included in an explanatory extension and on their ability to explain the consequent. The less nodes are included and the higher their explanatory power, the higher the probability of an explanatory extension. The calculation of $\Pr(EE_i|SC_i, \text{struct of } IG_i)$ is described in Sect. 6.2.2.
– $\Pr(\text{beliefs in } IG_i|SC_i, \text{struct of } IG_i, EE_i)$, the probability of the belief component of an interpretation graph, reflects feedback received in preliminary trials, where users objected to inferences that had increases in certainty or large changes in belief from their antecedents to their consequent (Zukerman et al. 2004; Zukerman and George 2005). Accordingly, interpretations that contain such objectionable inferences have a lower probability than interpretations where the beliefs in the consequents of the inferences fall within an "acceptable range" of the beliefs in their antecedents. The calculation of this probability is described in Sect. 6.2.1.
– The calculation of $\Pr(\mathit{UserArg}|SC_i, IG_i)$ is based on the calculation of $\Pr(\mathit{UserArg}|IG_i)$ described in Zukerman and George (2005). As for IG_i, this calculation also separates the structural component from the belief component of *UserArg*. This yields

$$\Pr(\mathit{UserArg}|SC_i, IG_i) = \Pr(\text{struct of } \mathit{UserArg}|\text{struct of } IG_i) \times \Pr(\text{beliefs in } \mathit{UserArg}|SC_i, \text{beliefs in } IG_i) \quad (6)$$

  • The structural component reflects how easy it is to obtain *UserArg* from IG_i by performing structural modifications to IG_i, such as deletions of nodes and additions and deletions of arcs (nodes are not added, as the user is not allowed to introduce new propositions). Given the structure of IG_i, this component is independent of SC_i.
  • The belief-based component represents how well the beliefs in the nodes in *UserArg* match the beliefs in the corresponding nodes in IG_i. This component depends on SC_i, as SC_i influences the beliefs in IG_i. The "right" suppositions can bring the beliefs in IG_i closer to those in *UserArg*, while the "wrong" suppositions can pull them apart.

The more structural modifications are required and the higher the discrepancy between the beliefs in IG_i and those in *UserArg*, the lower the probability that a user who intended IG_i and supposed SC_i said *UserArg*.

4 Positing suppositions

As stated in Sect. 2, the nodes in our BNs are binary. Hence, the possible supposition states that BIAS can posit are: SET TRUE—suppose that a node is True; SET FALSE—suppose that a node is False; and UNSET—suppose that a node has not been observed (i.e., ignore any evidence supplied by this node). Making a supposition may strengthen the influence of the antecedents of an inference on their consequent (as shown in the fourth segment in Fig. 2) or weaken it.

A supposition configuration describes the state of every node in the BN, hence there are 3^N such configurations (where N is the number of nodes in the BN). Since the number of nodes in the BNs implemented in BIAS ranges between 32 and 85, we cannot consider all possible supposition configurations, and we certainly cannot combine them with large numbers of interpretation graphs in the next step of algorithm *GenerateInterpretations*. We

Algorithm *GetSuppositionConfig(UserArg)*

1. If *SuppositionConfigList* is empty
   a) Call *MakeNewConfig(Supposition) K* (=300) times, where each time *MakeNewConfig* returns the best supposition configuration.
   b) Assign the top *k* (=20) supposition configurations to *SuppositionConfigList*.
2. Select an element from *SuppositionConfigList* at random.
3. Return the chosen configuration.

Algorithm *MakeNewConfig(ConfigType)*

1. If the priority queue is empty, **propose an initial configuration, calculate its impact on the probability of an interpretation**, and add the configuration and its probability impact to the priority queue.
2. Remove the first configuration from the queue.
3. **Generate the children of this configuration, calculate their probability impact**, and insert them in the queue so that the queue remains sorted in descending order of the probability impacts of the configurations.
4. Return the chosen (removed) configuration.

Fig. 5 Algorithms for generating suppositions

therefore find promising supposition configurations by generating only a limited number of supposition configurations that are close to the beliefs in the user model, and selecting from these the best *k* configurations as the basis for the generation of interpretation graphs.

Next we present our procedure for generating supposition configurations (George et al. 2005), followed by the calculation of the contribution of a supposition configuration to the probability of an interpretation.

4.1 Proposing supposition configurations

Algorithm ***GetSuppositionConfig*** (Fig. 5), which is called in Step 1 of algorithm *Generate Interpretations* (Fig. 3), receives as input an argument *UserArg* and returns a supposition configuration randomly selected from a short-list of *k* (=20) configurations. This short-list, which is denoted *SuppositionConfigList*, is generated by calling *MakeNewConfig(Supposition) K* (=300) times, and selecting the best *k* configurations.

Algorithm *MakeNewConfig*, which is called in Step 1(a) of *GetSuppositionConfig*, maintains a priority queue of configurations and their probabilities. Each time it is called, it removes the configuration at the top of the queue (which has the highest probability), generates its "child configurations" (derived from the removed one), inserts them in the queue according to their probability, and returns the removed configuration.[4] The boldfaced segments of the algorithm are explained later in this section.

We have adopted this process for the generation of supposition configurations, because observations of our system's behaviour indicate that there are only a few promising supposition configurations among the many possible options, but these configurations generally do not follow a monotonic pattern. Hence, a procedure that just descends a priority queue will not yield good results reliably. Further, trials performed during system development show that 20 promising configurations are sufficient to generate an appropriate interpretation in a

[4] This algorithm is also used to generate interpretation graphs and node configurations that match NL sentences, but here we focus on its use for generating supposition configurations.

Table 1 Sample supposition score table

node$_1$	*node*$_2$	...	*node*$_{32}$
UNSET: 0.7	SET TRUE: 0.8	...	UNSET: 0.7
SET TRUE: 0.21	UNSET: 0.15	...	SET TRUE: 0.15
SET FALSE: 0.09	SET FALSE: 0.05	...	SET FALSE: 0.15

relatively short time, and that the top 300 supposition configurations (obtained by repeatedly accessing the priority queue) provide a suitable basis for selecting these 20 configurations.[5]

The generation of supposition configurations and their children employs a structure called *Supposition Score Table*, which maps nodes to suppositions (Table 1 shows a sample Supposition Score Table for a 32-node BN). Each column in the Supposition Score Table corresponds to a node in the BN. Each node is associated with a list of three <supposition: probability> tuples—one for supposing SET TRUE, one for SET FALSE and one for UNSET—sorted in descending order of probability. The probabilities for these tuples are obtained by applying the following heuristics.

– *No change is best:* There is a strong bias towards not making suppositions that differ from the beliefs in the user model.
– *Users are unlikely to change their mind about observed evidence:* If a user has observed a node (i.e., its value is True or False), he or she is unlikely to change his or her belief in this node.
– *Small changes in belief are better than large changes:* If a node that is left unset has a high propagated belief, then it is more likely that the user is assuming it True than if its propagated belief was lower.

4.1.1 Generating and using the supposition score table

The above heuristics are implemented by means of the probabilities in Table 2. The second and third columns in Table 2 specify the probabilities of making suppositions about nodes that have been observed by a user. For example, if the user knows that GreenInGardenAt11 = True, then the probability of setting this node to True (leaving it unchanged) is 0.8, the probability of unsetting this node is 0.15, and the probability of setting it to False is 0.05. The fourth column in Table 2 specifies the probabilities of making suppositions about nodes which have not been observed by a user (i.e., nodes that are unset). As per the above heuristics, the bulk of the probability mass (0.7) is allocated to leaving a node unset. The remainder of the probability mass (0.3) is allocated as follows: 2/3 of this mass ($\Pr_{floating} = 0.2$) is distributed between the values True and False in proportion to the propagated probability of the node, and 1/3 of this mass is distributed equally between True and False as a fixed component ($\Pr_{fixed} = 0.05$ for each value). This ensures that some probability mass is allocated to each value, i.e., the probability of setting a node to True or False can not go below 0.05. For instance, if the propagated belief of unobserved node GreenHasMeans is Pr(GreenHasMeans) $= 0.8$, then the probability of leaving it unset is 0.7, the probability of setting it to True is $0.8 \times 0.2 + 0.05 = 0.21$, and the probability of setting it to False is $0.2 \times 0.2 + 0.05 = 0.09$. The probability assignments in Table 2, which reflect a reluctance to posit suppositions that differ from the beliefs in the

[5] The number of supposition configurations that should be generated for a particular BN would normally depend on its size. However, an experimental study of the effect of number of configurations and BN size on search performance is outside the scope of this research.

Table 2 Probability of making suppositions

Probability	Node was observed by the user Node = FALSE	 Node = TRUE	Node was not observed by the user
Pr(UNSET)	0.15	0.15	Pr_{unset} (=0.7)
Pr(SET FALSE)	0.8	0.05	Pr(FALSE) $\times$ $Pr_{floating} + Pr_{fixed}$
Pr(SET TRUE)	0.05	0.8	Pr(TRUE) $\times$ $Pr_{floating} + Pr_{fixed}$

user model, were derived manually. They lead to a stable system behaviour, in the sense that the system will posit a supposition that differs from the belief in a node only if it yields a payoff, i.e., a substantially better match between the beliefs in an interpretation and those in a user's argument.

The Supposition Score Table is used by elements of algorithm *MakeNewConfig* (Fig. 5) to generate supposition configurations as follows.

Propose an initial configuration (Step 1 of MakeNewConfig). Get the first row from the Supposition Score Table, which contains the supposition configuration with the highest probability. For the Supposition Score Table in Table 1, this configuration is {$node_1$: UNSET, $node_2$: SET TRUE, ..., $node_{32}$: UNSET}.

Generate the children of a configuration (Step 3). The ith child is generated by moving down one place in column i in the Supposition Score Table, while staying in the same place in the other columns. This process yields the following children for the initial configuration obtained from the Supposition Score Table in Table 1: {<u>$node_1$: SET TRUE</u>, $node_2$: SET TRUE, ..., $node_{32}$: UNSET}, {$node_1$: UNSET, <u>$node_2$: UNSET</u>, ..., $node_{32}$: UNSET}, ..., where the underlined node-supposition pair replaces an element in the parent supposition configuration.

4.2 Estimating the effect of supposition configurations on the probability of an interpretation

As seen in Eqs. 5 and 6, the probability of an interpretation depends partly on the match between the beliefs in IG_i (influenced by the suppositions in SC_i) and those in *UserArg*. Thus, if SC_i yields a better match between the beliefs in the interpretation and those in the user's argument, then Pr(beliefs in $UserArg|SC_i$, beliefs in IG_i) increases. As a result, the "cost" incurred by the suppositions in SC_i may be overcome by the "reward" resulting from the better match between the beliefs. This cost-reward balance, which reflects the impact of a supposition configuration on the probability of an interpretation, is represented by the product

$$\Pr(\text{beliefs in } UserArg|SC_i, \text{beliefs in } IG_i) \times \Pr(SC_i) \tag{7}$$

This product determines the position of configuration SC_i in the priority queue maintained by algorithm *MakeNewConfig* (Fig. 5). Thus, the supposition configurations that yield the best cost-reward balance among those considered so far are at the top of the queue (children that are more promising may be discovered next time *MakeNewConfig* is called). The first factor in this product appears in Eq. 6, and its calculation is summarized in Sect. 3.2. The second factor appears in Eq. 5, and its estimation is described below.

4.2.1 Estimating Pr(SC)

The prior probability of a supposition configuration, $\Pr(SC_i)$, is the product of the probabilities of the entries in the Supposition Score Table for the configuration in question. For instance, the initial configuration selected above has probability $0.7 \times 0.8 \times \cdots \times 0.7$, and configuration $\{\underline{node_1\text{: SET TRUE}}, node_2\text{: SET TRUE}, \cdots, node_{32}\text{: UNSET}\}$ has probability $\underline{0.21} \times 0.8 \times \cdots \times 0.7$. Thus, the more SC_i departs from the beliefs in the user model, the lower is $\Pr(SC_i)$, thereby reducing the overall probability of an interpretation which includes SC_i.

4.3 Determining the effect of suppositions on interpretations

Our process for generating supposition configurations proposes promising configurations in terms of improvements in the belief match between an argument and an interpretation. However, it does not take into account other types of interactions which may cause locally optimal supposition configurations and interpretation graphs to combine into interpretations that are sub-optimal or even invalid as a whole. For example, consider a situation where a user says $A \rightarrow G$*[Probably]*, and the most direct unblocked path between A and G in the BN is $A \rightarrow B \rightarrow C \rightarrow G$. Now, say that BIAS must suppose that proposition E is True in order to obtain the user's belief in G. This supposition in turn may unblock a shorter path between A and G, yielding a better interpretation. Such global effects are considered during the evaluation of an interpretation as a whole (Step 4 of algorithm *GenerateInterpretations*, Sect. 3.1).

5 Evaluation of suppositions

Our evaluation of the module for postulating suppositions was conducted as follows. We constructed four scenarios from one of our 32-node BNs: Crimson, Lemon, Sienna and Mauve (Fig. 6). These scenarios test various supposition alternatives as follows. The Crimson and Sienna scenarios required supposing that a node is True in order to strengthen the belief in the goal proposition of an argument; the Lemon scenario required a True supposition in order to unblock a path; and the Mauve scenario required unsetting or "forgetting" the value of a node to weaken the belief in the goal proposition of an argument. Each scenario contained background evidence (not shown in Fig. 6) and two versions of a short argument for a goal proposition in our BN. One version (denoted "We think that") stated the belief obtained for the goal proposition by performing Bayesian propagation from the evidence, and the other version (denoted "If someone says") gave a different belief for this proposition. The trial subjects were then asked to determine what this "someone" may be supposing in order to account for his or her belief in the goal proposition.

We have used this "indirect" evaluation method (instead of having subjects interact freely with the system) for several reasons. From a practical point of view, we wanted to focus on a particular behaviour of the system (the postulation of suppositions) that does not occur for every argument, and we wanted to remove extraneous factors (such as interface usability) from the evaluation. From a theoretical standpoint, ideally BIAS' performance should be comparable to that of people. That is, even if BIAS sometimes gets things wrong, its mistakes should be plausible. Hence, a fair assessment of BIAS' interpretations is whether they are considered reasonable by other "addressees".

Since the purpose of our evaluation is to determine whether BIAS generates sensible suppositions in the context of its domain knowledge, we needed to limit the suppositions available to our trial subjects to the propositions known to BIAS. However, at the same time, we did

CRIMSON SCENARIO	LEMON SCENARIO
We think that *If forensics matched the bullets in Mr Body's body with the found gun, then the suspect Mr Green **possibly** had the means to murder Mr Body.*	**We think that** *If broken glass was found, then Mr Body's window **probably wasn't** broken from outside.*
If someone says *Forensics matching the bullets with the found gun means Mr Green **very probably** had the means to murder Mr Body.*	**If someone says** *Broken glass being found means Mr Bodyís window **probably was** broken from outside.*
Then it would be reasonable to think that they are supposing	
S1) Mr Green fired the gun found in the garden.	S1) Broken glass was found inside the window.
S2) The gun found in the garden is the murder weapon.	S2) The suspect Mr Green argued with Mr Body last night.
S3) Mr Green fired the murder weapon.	S3) Mr Body was killed from outside the window.
S4) Mr Green murdered Mr Body.	S4) Mr Green was in the garden at the time of death.
S5) None of the above are suitable as suppositions. A more likely supposition (in light of what our system understands) is ... [LINK TO LIST OF PROPOSITIONS]	
S6) It is not appropriate to think they are supposing anything.	

SIENNA SCENARIO	MAUVE SCENARIO
We think that *If the suspect Mr Green was in the garden at 11 o'clock last night, then Mr Green **possibly didnút** have the opportunity to murder Mr Body.*	**We think that** *If the suspect Mr Green fired the murder weapon and the Bayesian Times newspaper reported that Mr Body seduced Mr Green's girlfriend, then Mr Green **probably** murdered Mr Body.*
If someone says *Mr Green being in the garden at 11 o'clock last night means Mr Green **probably did** have the opportunity to murder Mr Body.*	**If someone says** *Despite Mr Green firing the murder weapon and the Bayesian Times newspaper reporting that Mr Body seduced Mr Green's girlfriend, it is **only even chance** that Mr Green murdered Mr Body.*
Then it would be reasonable to think that	
they are supposing	**they have forgotten or ignored** ....
S1) The time of death was 11 o'clock.	S1) The neighbour heard Mr Green argue with Mr Body last night.
S2) Mr Green was in the garden at the time of death.	S2) Bullets were found in Mr Body's body.
S3) The time of death was NOT 11 o'clock.	S3) The time of death was 11 o'clock.
S4) Mr Green's ladder was found at the bedroom window.	S4) Broken glass was found inside the window.
S5) None of the above are suitable as suppositions. A more likely supposition (in light of what our system understands) is ... [LINK TO LIST OF PROPOSITIONS]	S5) They have made a supposition, not forgotten evidence. The most likely supposition (in light of what our system understands) is ... [LINK TO LIST OF PROPOSITIONS]
S6) It is not appropriate to think they are supposing anything.	S6) It is not appropriate to think they are supposing or forgetting anything.

Fig. 6 Scenarios for user trials for suppositions

not wish to burden our subjects with the need to look through BIAS' knowledge base to find out what BIAS knows. Additionally, we wanted to allow respondents some freedom to state their views, if they disagreed with BIAS' suppositions. These requirements were addressed by presenting our subjects with the following options (Fig. 6): (S1–S4) a list of four candidate suppositions (one was the top supposition recommended by BIAS, and most of the others were considered by BIAS to be reasonable options); (S5) an option to include an alternative supposition (the subjects were provided a link to a list containing the propositions in the BN, but could also write a supposition of their own); and (S6) an option to state that they didn't believe that any suppositions were required.

Table 3 Ranking of candidate suppositions for the four scenarios

	Total	R1	R2	Other
CRIMSON SCENARIO				
Supposition S1	20	10	8	2
Supposition S2	18	11	3	4
Total responses	75	34	21	20
LEMON SCENARIO				
Supposition S1	30	30	0	0
Supposition S3	11	0	11	0
Total responses	51	34	12	5
SIENNA SCENARIO				
Supposition S1 + S3	30	20	6	4
Supposition S4	16	7	6	3
Total responses	69	34	19	16
MAUVE SCENARIO				
Supposition S1	25	19	4	2
Supposition S5	14	8	5	1
Total responses	62	34	16	12

The order of presentation of the suppositions was randomized across the scenarios. However, for the discussion in this paper, BIAS' preferred supposition was re-labeled S1. The trial subjects had to award a rank of 1 to one option and could optionally rank additional alternatives (with inferior ranks). This allowed respondents to ignore suppositions that didn't make sense to them, while enabling them to include more than one option that seemed reasonable. At the same time, the results obtained by this method enable us to determine whether BIAS' suppositions are considered sensible, even if they are not our subjects' top-ranked preferences.

Our four scenarios were considered by 34 participants. Many of the respondents had not been exposed to BIAS previously and were from outside academia. The responses for the Lemon and Mauve scenarios were clearly positive, while the responses for the Crimson and Sienna scenarios were more ambiguous, but still positive. The results for these scenarios are shown in Table 3. The columns contain the total number of respondents that ranked a supposition (Total), and the number of respondents that ranked it first (R1), second (R2) or gave it a lower rank (Other). The top two rows for each scenario contain the suppositions that were preferred by the trial subjects, and the bottom row lists the total responses for each scenario and for the different ranks (recall that the only rank that had to be given was 1). Our results are summarized below.

– Supposition S1 was clearly the most favoured choice for the *Lemon* scenario, with 30 of the 34 respondents ranking it first. Supposition S3 was clearly the next best choice, with 11 trial subjects giving it rank 2.
– Supposition S1 was the preferred choice for the *Mauve* scenario, with 19 of the 34 respondents giving it a rank of 1. The next best choice was an alternative supposition (S5), with only 8 subjects ranking it first. There were no clear preferences for rank 2, with all options receiving this rank at least once, but never more than five times.
– Suppositions S1 and S2 for the *Crimson* scenario were similarly ranked (each ranked first by about 1/3 of the subjects), with Supposition S1 being favoured slightly over S2 for the

first two ranks, but not significantly so. The other options were ranked first only by a few trial subjects.
- The responses for the *Sienna* scenario presented us with a special case. The comments provided by our trial subjects indicated that there was some confusion due to the wording of the instructions combined with the fact that, unlike the other scenarios, the Sienna scenario included a True and a False version of the same node (Supposition S1 was "The time of death was 11 pm last night" and S3 was the negation of S1). Further, Supposition S3 supports the "We think that" version, while S1 supports the "If someone says" version. As a result, most of the respondents were divided between giving a rank of 1 to Supposition S1 or Supposition S3. Nonetheless, the main outcome from this scenario is that regardless of how the respondents read it, they clearly felt that a supposition had to be made about the "time of death" node, which was ranked first by 20 of the 34 respondents.
- Overall, very few trial subjects felt that no suppositions were warranted (9 for all the scenarios combined). Further, BIAS' preferred supposition was consistently ranked first or second, with its average rank being the lowest (best) among all the options.

These results show the importance of making suppositions, and indicate that the suppositions posited by BIAS not only are considered reasonable by people, but also have significant support.

6 Proposing explanatory extensions

In early trials with our system (Zukerman and George 2005), users objected to inferences where

- the consequent had a greater degree of certainty than its antecedents, e.g., "Mr Green *probably* had the means to murder Mr Body. Therefore, Mr Green *very probably* murdered Mr Body"; or
- the belief in the consequent was substantially different from the belief in its antecedents (even if there was a reduction in the level of certainty), e.g., "Mr Green *probably* being in the garden at 11 implies that it is *even chance* that Mr Green was in the garden at the time of death".

Sometimes such *objectionable inferences* are caused by unintuitive inference patterns. These patterns, which are encoded in the Conditional Probability Tables (CPTs) of the arcs that connect the antecedents of an inference with the consequent, may be justified by explaining the CPTs in question (a task that is outside the scope of this research). However, many objectionable inferences may be explained by influences from nodes that are not part of BIAS' initial interpretation. We postulate that users may have implicitly considered these nodes when constructing their arguments, and hence including these nodes in the objectionable inferences in an interpretation would turn them into *acceptable inferences*. Such inclusions constitute explanatory extensions.

We then conducted another survey to determine the types of inferences preferred by people from the standpoint of the relationship between the consequent and the antecedents. Our survey was restricted to *monotonic inferences*, where a high/low probability for an antecedent yields a high/low probability for the consequent. The results from our preliminary survey prompted us to distinguish between three types of acceptable inferences for this new survey: *BothSides, SameSide* and *AlmostSame*.[6]

[6] Additional categories are required for "inverse" inferences, where a low/high probability antecedent yields a high/low probability consequent, but they are not discussed in this paper for clarity of exposition.

- *BothSides* inferences have antecedents with beliefs on both "sides" of the consequent (in favour and against), e.g.,
  A[VeryProbably] & B[ProbablyNot] implies C[EvenChance].
- All the antecedents in *SameSide* inferences have beliefs on "one side" of the consequent, but at least one antecedent has the same belief level as the consequent, e.g.,
  A[VeryProbably] & B[Possibly] implies C[Possibly].
- All the antecedents in *AlmostSame* inferences have beliefs on one side of the consequent, but the closest antecedent is one level "up" from the consequent, e.g.,
  A[VeryProbably] & B[Possibly] implies C[EvenChance].

Our survey considered inferences in three domains: burglary, going to the beach, and a court case. Our results showed that people prefer *BothSides* inferences to the other categories. They also prefer *SameSide* to *AlmostSame* for antecedents with beliefs in the negative range (VeryProbablyNot, ProbablyNot, PossiblyNot); and they do not distinguish between *SameSide* and *AlmostSame* for antecedents with beliefs in the positive range. Further, *BothSides* inferences with three antecedents were preferred to *SameSide* inferences with two antecedents, indicating that persuasiveness carries more weight than parsimony.

We now present our mechanism for identifying objectionable inferences, and generating explanatory extensions according to the above preferences. We then estimate the probabilities in Eq. 5 (Sect. 3.2) that are affected by these extensions.

6.1 Proposing explanatory extensions

Algorithm *GetExplanatoryExtensions* (Fig. 7), which is called in Step 3 of algorithm *GenerateInterpretations* (Fig. 3), receives as input a supposition configuration and an interpretation graph, and returns a (possibly empty) set of explanatory extensions to be added to inferences in the graph. To this effect, it performs the following actions for each inference. First, it activates algorithm *IsNonSequitur* to determine whether the inference is objectionable (according to user opinions expressed in our preliminary trials). If so, it calls algorithm *GetBestCategory* to propose a set of BN nodes that could defuse or mitigate the objection. *GetBestCategory* returns the best non-empty set of such nodes according to the preferences expressed by our survey participants. Unlike the process used to generate supposition configurations and interpretation graphs, the generation of explanatory extensions is deterministic, producing at most one explanatory extension (comprising one or more nodes) for each objectionable inference in the interpretation graph.

Algorithm *IsNonSequitur* receives as input an inference composed of antecedents and a consequent, and checks whether (a) the consequent has a greater degree of certainty than the most certain antecedent, or (b) the consequent has a lower level of certainty than the least certain antecedent, and the belief in this consequent is substantially different from the belief in this antecedent. As indicated in Sect. 3, beliefs are specified in terms of our seven linguistic belief categories, denoted *BelCat*. These are {VeryProbably, Probably, Possibly, EvenChance, PossiblyNot, ProbablyNot, VeryProbablyNot}.

Algorithm *GetBestCategory* receives as input an inference that was identified as a non-sequitur, an interpretation graph and a supposition configuration, and proposes BN nodes that explain the non-sequitur. Our algorithm restricts its attention to nodes that (a) are informative, i.e., their belief stems from independent evidence; (b) are not in the interpretation graph or in previously generated explanatory extensions; and (c) are closely related to the

Algorithm *GetExplanatoryExtensions(SC,IG)*
For each *Inference* $\in$ *IG*
If *IsNonSequitur(Inference)* Then

1. $EE \leftarrow$ *GetBestCategory(Inference,SC,IG)*
2. Include EE in *Inference*

Algorithm *IsNonSequitur(Inference)*
Let *Inference* = [$\{Antecedents\} \rightarrow Consequent$]
If the Consequent is more certain than the Antecedents OR
{ the Consequent is less certain than the least certain Antecedent AND
$|BelCat(Consequent) - BelCat(\text{least certain } Antecedent)| > 1$ }
Then *Inference* is a *NonSequitur*

Algorithm *GetBestCategory(Inference,SC,IG)*
Let *Inference* = [$\{Antecedents\} \rightarrow Consequent$]

1. For each *Sibling* of $\{Antecedents\}$, such that
Sibling $\notin IG$ AND *Sibling* $\notin$ a previous explanatory extension
If *IsInformative(Sibling)* Then add *Sibling* to *InformativeSiblings*
2. For each sibling in *InformativeSiblings*, determine the category of the resultant inference if that sibling was included in the inference (*BothSides, SameSide, AlmostSame* or *SmallNonSeq*, which is explained below). Siblings outside these categories are ignored.
3. Return all the siblings in the best non-empty category according to the following preference ordering: *BothSides* $\succ$ *SameSide* $\succ$ *AlmostSame* $\succ$ *SmallNonSeq*.

Algorithm *IsInformative(Sibling)*
If { *Sibling* is an evidence node OR
There is an unblocked path outside the interpretation graph
from an evidence node to *Sibling* }
Then *Sibling* is an *InformativeSibling*

Fig. 7 Algorithms for adding explanatory extensions to an interpretation

argument, specifically we focus on nodes with a link to the consequent of the offending inference (i.e., *siblings* of the antecedents).

Algorithm *IsInformative* receives a node as input, and determines whether there is independent evidence for its belief. For instance, as shown in Fig. 1 and in the first segment of Fig. 2, there is independent evidence supporting ¬GreenHasMeans, but there is no independent evidence for or against GreenHasMotive. Hence, as seen in the last segment of Fig. 2, only GreenHasMeans[ProbablyNot] was added to the interpretation to explain the drop in belief from GreenHasOpportunity[Possibly] to GreenMurderedBody[PossiblyNot].

To illustrate these algorithms, consider the following interpretation graph obtained from the BN in Fig. 1: GreenInGarden[Probably] $\rightarrow$ GreenInGardenAt11[VeryProbably] $\rightarrow$ Green InGardenAtTimeOfDeath[VeryProbably]. The first inference in this graph is a non-sequitur, as there is an increase in certainty. Hence, an explanatory extension is required. The siblings of the antecedent of this inference are GreenInGardenAtTimeOfDeath, WitnessSawGreen AtFootballAt10:30 and NbourSawGreenAroundGardenAt11. However, only the last sibling meets all the above conditions (the first sibling is already in the interpretation graph, and there is no evidence regarding the second sibling). Hence only NbourSawGreenAroundGardenAt11 is added to *InformativeSiblings*.

After the informative siblings have been identified, algorithm *GetBestCategory* determines how including each of these siblings in an objectionable inference would influence its acceptability. That is, whether a sibling would turn the inference into a *BothSides*, *SameSide*,

(a) Large drop in certainty, decrease in belief: creating a *BothSides* inference

Mr Green had the means to murder Mr Body [Probably]

BUT [EE] Mr Green had the opportunity to murder Mr Body [PossiblyNot]

IMPLIES

Mr Green murdered Mr Body [EvenChance]

(b) Increase in certainty, increase in belief: creating a *SameSide* inference

Mr Green visited Mr Body last night [PossiblyNot]

BUT [EE] Mr Green was in the garden at the time of death [VeryProbably]

IMPLIES

Mr Green had the opportunity to murder Mr Body [VeryProbably]

Fig. 8 Explanatory extensions added to sample interpretations

AlmostSame or *SmallNonSeq* inference (or none of these, in which case the sibling is discarded). The *SmallNonSeq* category, which was not defined before, comprises siblings that improve an inference, but not enough to make it acceptable. This category extrapolates from the results of our survey, in that the inclusion of siblings from this category presumes that a small departure from what is considered acceptable is less offensive than a large departure.[7] The algorithm then returns all the siblings in the most preferred non-empty category, where *BothSides* $\succ$ *SameSide* $\succ$ *AlmostSame* $\succ$ *SmallNonSeq*.

It is worth noting that our categorization of siblings considers only the belief in the siblings, assuming that their influence on the consequent of the inference in question is monotonic (i.e., the higher the belief in the sibling, the stronger its influence on the consequent). That is, we do not consider the impact of a sibling in the context of other nodes linked to the consequent. For instance, it is possible that due to the configuration of the CPTs for the inference, a *BothSides* sibling has less impact on the belief in the consequent than a *SameSide* sibling. A promising approach for addressing this problem consists of incorporating into our inference categories a probabilistic formulation of the impact of a node (Zukerman et al. 2000).

6.1.1 Examples

To illustrate the workings of algorithm *GetExplanatoryExtensions*, consider the inferences in Fig. 8. These inferences are non-sequiturs, which are defused by explanatory extensions.

The inference in Fig. 8a illustrates a large drop in certainty and a decrease in belief. It goes from "Mr Green *probably* having the means to murder Mr Body" to "Mr Green *may be (even chance)* murdering Mr Body". Our algorithm examines the siblings of GreenHasMeans, which are BodyWasMurdered, GreenHasMotive and GreenHasOpportunity (Fig. 1). There is no independent evidence for GreenHasMotive, so it is not an *InformativeSibling*. The other two siblings are informative. In this example, BodyWasMurdered has a belief of VeryProbably, which is obtained from the evidence node BulletsFoundInBody'sBody [VeryProbably] by Bayesian propagation. However, BodyWasMurdered does not fit the four categories of interest, as its belief is higher than that of the antecedent. Hence, it is dropped from consideration. GreenHasOpportunity has a belief of PossiblyNot derived from the evidence nodes NbourSawGreenAroundGardenAt11 and TimeOfDeath11, which in this example have a belief of VeryProbablyNot and VeryProbably respectively. Hence, GreenHasOpportunity is added to the interpretation as an explanatory extension that yields a *BothSides* inference.

7 Our trial focused on the acceptance of interpretation graphs. Therefore, users were not asked to compare between different types of non-sequiturs.

The inference in Fig. 8b illustrates an increase in certainty and an increase in belief. It goes from "Mr Green *possibly not* visiting Mr Body last night" to "Mr Green *very probably* having the opportunity to murder Mr Body". Our algorithm considers the siblings of GreenVisitBodyLastNight, which are BodyKilledFromOutsideWindow, GreenMurderedBody and GreenInGardenAtTimeOfDeath (Fig. 1). In this example, the first two siblings have a belief of EvenChance, which puts them in the *SmallNonSeq* category, while GreenInGardenAtTimeOfDeath has a belief of VeryProbably, which puts it in the *SameSide* category. Hence it is added as an explanatory extension.

6.2 Estimating the effect of explanatory extensions on the probability of an interpretation

According to Eq. 5 (Sect. 3.2), explanatory extensions affect the probability of an interpretation through the following product, which expresses a cost-reward balance similar to that described in Eq. 7 (Sect. 4.2) for supposition configurations.

$$\Pr(\text{beliefs in } IG_i | SC_i, \text{struct of } IG_i, EE_i) \times \Pr(EE_i | SC_i, \text{struct of } IG_i) \quad (8)$$

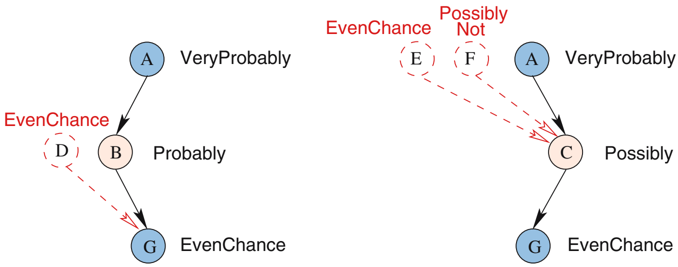
$\Pr(EE_i|SC_i, \text{struct of } IG_i)$ represents the cost incurred by presenting explanatory extensions, and $\Pr(\text{beliefs in } IG_i|SC_i, \text{struct of } IG_i, EE_i)$ represents the reward due to the elimination or reduction of non-sequiturs. The calculation of these factors appears in Appendix A. Below we summarize the results of these calculations and illustrate them with an example.

6.2.1 *Estimating Pr*(beliefs in $IG|SC$, struct of IG, EE)

The calculation of the probability of the beliefs in IG is described in Appendix A.1. This probability is not about the beliefs themselves in the domain BN. Rather, it is about the extent to which the inferences that yield these beliefs satisfy people's preferences about the relationship between the antecedents of these inferences and their consequent. Inferences which reflect people's preferences are more probable than inferences which do not reflect these preferences.

These preferences are represented in the three types of acceptable inferences gleaned from our surveys: *BothSides, SameSide* and *AlmostSame*. These inference categories specify a range of beliefs considered acceptable for the consequent of an inference given its antecedents. We call this range an *acceptable range*. For example, an inference with antecedents *A[Probably] & B[Possibly]* has the acceptable range {Probably, EvenChance} (an acceptable belief for its consequent is Probably, Possibly or EvenChance). The probability of an inference whose consequent falls within the acceptable range is higher than the probability of an inference whose consequent falls outside this range. In addition, we extrapolate from the results of our survey, and posit that the probability of an objectionable inference decreases as the distance of its consequent from the acceptable range increases. For example, the probability of *A[VeryProbably] implies B[Possibly]* is higher than the probability of *A[VeryProbably] implies B[PossiblyNot]*, as the acceptable range is {VeryProbably,Probably}, and Possibly is closer to this range than PossiblyNot.

Following Zukerman and George (2005), we use the Zipf distribution to model the probability of an inference, but extend that model to reflect the fact that the acceptable range may comprise more than one belief category. The probability of the beliefs in an interpretation graph is then obtained by multiplying the belief probabilities for the inferences in the graph.



(a) Interpretation 1 **(b)** Interpretation 2

Fig. 9 Two interpretations with explanatory extensions

6.2.2 Estimating $Pr(EE|SC, \text{struct of } IG)$

The calculation of the probability of an explanatory extension is described in Appendix A.2. This calculation is based on the following heuristics:

– Users prefer short explanations, but not at the expense of persuasiveness. This heuristic implements the Gricean Maxim of Brevity (Grice 1975) in combination with the feedback given by our survey respondents. Following Zukerman and George (2005), we model this heuristic using a truncated Poisson distribution, $Poisson(\mu)$, where μ is the average number of nodes in an explanatory extension. We use $\mu = 1.5$, which penalizes explanatory extensions with many nodes.
– Explanatory extensions that yield *BothSides* inferences are preferred over explanatory extensions that yield *SameSide* or *AlmostSame* inferences, which are much preferred over explanatory extensions that produce *SmallNonSeq* inferences. This heuristic reflects our subjects' feedback for antecedents with beliefs in the positive range (a similar heuristic exists for beliefs in the negative range). It is modelled by means of a manually-generated probability distribution that assigns a probability to an inference category according to people's preferences, i.e., higher preferences yield higher probabilities. However, it is worth noting that these preferences are qualitative, hence similar probability assignments may also work well.

Thus, the probability of a short explanatory extension that generates a *BothSides* inference is higher than that of a long explanatory extension that generates an *AlmostSame* inference. In contrast, the probability of a long *BothSides* explanatory extension could be similar to that of a short *AlmostSame* extension.

6.2.3 Example

Figure 9 illustrates the generation of explanatory extensions for the simple argument $A \rightarrow G$. The interpretation graphs differ in the BN node that was included to connect between the antecedent and the consequent. In addition, in Interpretation 1, the inference between node B and node G is objectionable, while in Interpretation 2, the inference between node A and node C is objectionable. Algorithm *GetExplanatoryExtensions* deterministically proposes the best explanatory extension for the objectionable inference in each interpretation. This turns the second inference of Interpretation 1 into a *SameSide* inference, and the first inference of Interpretation 2 into a *BothSides* inference. The probabilities of these explanatory extensions are then incorporated in the calculation of the overall probability of the interpretations, and the interpretation with the highest probability is selected. In this example this is Interpretation 2.

7 Evaluation of explanatory extensions

Our evaluation of the module for adding explanatory extensions was conducted as follows. We constructed two evaluation sets from two 32-node BNs, each consisting of a short argument and two alternative interpretations, one with explanatory extensions and one without (Fig. 10). The evaluation sets represented the following two conditions: in the first set, the interpretation without explanatory extensions had a small increase in belief, and in the second set, the interpretation without explanatory extensions had a large decrease in belief.

The two evaluation sets were shown to 20 subjects, who exhibited diverse levels of computer literacy. In our experiment, we first gave our subjects a definition and example of an interpretation, and told them that the aim of the experiment was to compare two methods for presenting BIAS' argument interpretations. We then showed them the arguments and interpretations in Fig. 10, but we inverted the order of presentation of the interpretations in Fig. 10b.

Our trial subjects clearly favoured the interpretations with explanatory extensions, which were preferred by 57.5% of the subjects, compared to 37.5% of the subjects who preferred the interpretations without such extensions, and 5% who were indifferent. Despite their clear preference for interpretations with explanatory extensions, 45% of the subjects felt that the extended interpretations were too verbose, while 17.5% thought that the extended interpretations still lacked information. Only 7.5% of the trial subjects thought that the short interpretations were already too verbose. These results indicate that the subjects preferred to know more about the system's reasoning, but had some problems with its presentation. These problems may be partially attributed to the presentation of the nodes as full canonical sentences (direct renditions of the propositional content of a node), which makes the interpretations appear repetitive in style, and hence may have an adverse influence on acceptance. The generation of stylistically diverse text is the subject of active research in Natural Language Generation, e.g., (Paiva 1999; Gardent and Kow 2005).

8 Generalizability: The Chest Clinic BN

As mentioned in Sect. 2, our implementation currently assumes that the nodes in the BN are binary. This assumption limits the BNs on which our formalism can be immediately applied, as most BNs are not entirely binary (the relaxation of this assumption is discussed in Sect. 10). The Netica website contains a few binary BNs, of which we found that the 8-node Chest Clinic BN (also known as Asia) was immediately usable (Fig. 11).[8]

Despite its small size, the Chest Clinic BN illustrates the behaviour of our procedures for proposing suppositions and generating explanatory extensions. Table 4 shows the four short arguments we devised to test this BN, together with BIAS' preferred interpretations. After performing Bayesian propagation from the instantiated nodes (the antecedents of the arguments), all the arguments yielded a belief of EvenChance in the consequent. Hence, all

[8] Other binary BNs are the 6-node Fire BN, the 8-node Neapolitan90 BN, the 23-node Boerlage BN, and the 76-node Win95pts BN. However, the first BN is too small, the nodes in the second are just letters of the alphabet, the third also represents a fictional story like ours, and the last requires domain-specific knowledge to follow the arguments.

(a) ARGUMENT
*The neighbour seeing Mr Green in the garden at 11 implies that it is **very probable** that Mr Green had the opportunity to murder Mr Body*

Interpretation Graph only	**Interpretation Graph + Explanatory Extension**
The neighbour seeing Mr Green in the garden at 11	*The neighbour seeing Mr Green in the garden at 11*
IMPLIES	IMPLIES
*It is **probable** that Mr Green was in the garden at 11*	*It is **probable** that Mr Green was in the garden at 11*
	TOGETHER WITH
	The neighbour hearing Mr Green and Mr Body argue late last night
	AND
	The time of death being 11
IMPLIES	IMPLIES
*It is **very probable** that Mr Green was in the garden at the time of death*	*It is **very probable** that Mr Green was in the garden at the time of death*
IMPLIES	IMPLIES
*It is **very probable** that Mr Green had the opportunity to murder Mr Body*	*It is **very probable** that Mr Green had the opportunity to murder Mr Body*

(b) ARGUMENT
*A gun was found in the garden and forensics matched Mr Green's fingerprints with those on the gun, but Mr Green **possibly did not** murder Mr Body*

Interpretation Graph only	**Interpretation Graph + Explanatory Extension**
A gun was found in the garden AND	*A gun was found in the garden AND*
Forensics matched the fingerprints on the gun with Mr Green	*Forensics matched the fingerprints on the gun with Mr Green*
IMPLIES	IMPLIES
*Mr Green **very probably** was in the garden*	*Mr Green **very probably** was in the garden*
	BUT
	The neighbour did not hear Mr Green argue with Mr Body late last night
THEREFORE	THEREFORE
*It is **possible** that Mr Green **was not** in the garden at 11*	*It is **possible** that Mr Green **was not** in the garden at 11*
IMPLIES	IMPLIES
*He **possibly was not** in the garden at the time of death*	*He **possibly was not** in the garden at the time of death*
IMPLIES	IMPLIES
*It is **possible** that Mr Green **did not** have the opportunity to murder Mr Body*	*It is **possible** that Mr Green **did not** have the opportunity to murder Mr Body*
IMPLIES	IMPLIES
Mr Green ***possibly did not*** murder Mr Body	Mr Green ***possibly did not*** murder Mr Body

Fig. 10 Arguments and interpretations in user trials for explanatory extensions

the interpretations required the postulation of suppositions to obtain the beliefs stated for the consequents.

The interpretations of the third and fourth arguments include the supposition as an explanatory extension (once a supposition has been added to the user model, it may be used as an explanatory extension). In the third argument, the supposition explains the large difference in belief between the antecedent and the consequent in the first inference. In the fourth argument, the supposition explains the increase in certainty that takes place between the antecedent and the consequent in the second inference. In addition, the supposition influences Dyspnea, which is included as an explanatory extension for the first inference, turning it into an *AlmostSame* inference.

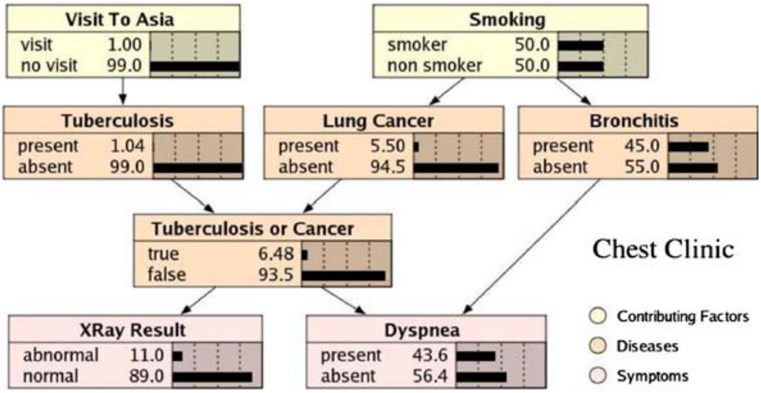


Fig. 11 ChestClinic BN

Table 4 Four arguments and interpretations for the Chest Clinic BN

	Argument/Interpretation
Argument 1	*Smoking =imply⇒ LungCancer [Possibly]*
Interpretation	**Supposing that XrayAbnormal** Smoking =imply⇒ LungCancer [Possibly]
Argument 2	*XrayAbnormal =imply⇒ LungCancer [Possibly]*
Interpretation	**Supposing that Smoker** XrayAbnormal =imply⇒ TBorCancer [Probably] TBorCancer [Probably] =imply⇒ LungCancer [Possibly]
Argument 3	*Despite XrayAbnormal, LungCancer [ProbNot]*
Interpretation	**Supposing that NOT Dyspnea** Although XrayAbnormal, **Not Dyspnea** =imply⇒ TBorCancer [ProbNot] TBorCancer [ProbNot] =imply⇒ LungCancer [ProbNot]
Argument 4	*Despite XrayAbnormal, LungCancer [VeryProbNot]*
Interpretation	**Supposing that NOT Smoker** Although XrayAbnormal, **Dyspnea [PossiblyNot]** =imply⇒ TBorCancer [ProbNot] TBorCancer [ProbNot] **+ Non Smoker** =imply⇒ LungCancer [VeryProbNot]

9 Related research

An important aspect of discourse understanding involves filling in information that was omitted by the interlocutor. In this paper, we have identified three types of such omitted information: inferential gaps, suppositions and unstated information, and we have presented a process that integrates the mechanisms for filling in this information in the context of an argumentation system. Our process is based on (1) an anytime algorithm that searches for candidate argument interpretations, (2) a probabilistic formulation for the evaluation of these interpretations, and (3) a BN for reasoning about the domain.

Our original version of BIAS featured the BN representation, and applied the probabilistic formulation to evaluate interpretations that fleshed-out intermediate reasoning steps omitted by a user (Zukerman et al. 2003; Zukerman and George 2005). The need to postulate suppositions and explanatory extensions was determined during the evaluation of that system. A preliminary algorithm for proposing explanatory extensions was presented in

Zukerman et al. (2004), but it was not integrated with our probabilistic process. Our anytime algorithm was described in George et al. (2004), and our procedure for positing suppositions in George et al. (2005).

In this section, we focus on research related to our work on positing suppositions and proposing explanatory extensions (research related to the other aspects of our system was discussed in Zukerman and George 2005).

Our approach offers an abductive account of a user's argument. Such accounts have been provided by several researchers for different discourse interpretation tasks, (e.g., Ng and Mooney 1990; Hobbs et al. 1993; McRoy and Hirst 1995). Our work resembles that of these researchers in its postulation of assumptions, and its use of expectations (which in our case are prior probabilities) to guide the selection of an interpretation. However, there are significant differences between our work and theirs. Hobbs et al. focused on problems of reference and disambiguation in single sentences, and McRoy and Hirst considered the identification of speech acts in a dialogue. Both of these tasks propose a model that explains a single datum. In contrast, like Ng and Mooney, our model explains relational information, i.e., discourse consisting of several propositions. However, Ng and Mooney apply a coherence heuristic to select an explanation for a user's discourse, while our selection process is based on a probabilistic framework that incorporates people's preferences. Additionally, the above researchers employ a logic-based formalism for the selection of an interpretation, while our approach is probabilistic.

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 incrementally extended a *plan recognition BN* from propositions read in a story, so that their BN represented a probability distribution over the set of possible explanations for these propositions. The most likely interpretation for a set of actions in the story was then selected. We also select the interpretation with the highest posterior probability. However, we use a domain BN to constrain our understanding of a user's argument. Further, our inclusion of suppositions in an interpretation is motivated by discrepancies between the user's stated beliefs and those obtained in the BN, and our inclusion of explanatory extensions is motivated by inferences that people find objectionable. 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). 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. Our mechanism for proposing suppositions could be applied to both Gertner et al.'s work and Horvitz and Paek's work to explain outcomes that differ from those predicted by their system.

Despite the pervasiveness of BNs, the explanation of the reasoning performed by a BN has been considered only by a few researchers (Druzdzel 1996; McConachy et al. 1998; Jitnah et al. 2000). Druzdzel and McConachy et al. studied different aspects of the presentation of BNs. Druzdzel focused on the reduction of the number of variables being considered, verbal expressions of uncertainty, and qualitative explanations, which were generated by tracing the influence of the nodes in a BN. McConachy et al. applied attentional models to the construction of probabilistic arguments, and studied probabilistic argumentation patterns and argumentation strategies. Jitnah et al. extended this work to the selection of strategies for rebutting users' rejoinders to the system's arguments, relying on a measure related to mutual information to determine the influence of a rejoinder node on the argument (Zukerman et al. 2000). The presentations produced by the last two systems hinged on discrepancies between the system's beliefs and the user's. Such discrepancies may be inferred by BIAS' procedure for positing suppositions, enabling the combination of these systems. Further, an interesting avenue for future research would involve incorporating models based on attention and mutual-information into our probabilistic formalism for deciding which nodes to include in an explanatory extension.

The research reported by Joshi et al. (1984), van Beek (1987) and Zukerman and McConachy (2001)) considers the addition of information to planned discourse to prevent or weaken a user's erroneous inferences from this discourse. Our mechanism for adding explanatory extensions to an interpretation prevents inferences that people find objectionable due to jumps in belief. Since such non-sequiturs may also be present in system-generated arguments, the approach presented here may be incorporated into argument-generation systems.

Several researchers have considered *presuppositions*, a type of suppositions implied by the wording of a statement or query (Kaplan 1982; Motro 1986; Mercer 1991; Gurney et al. 1997). For instance, "How many people passed CS101?" presupposes that CS101 was offered and that students were enrolled in it (Kaplan 1982). Mercer used default logic together with lexical information to identify a speaker's presuppositions. Gurney et al. used active logic plus syntactic and lexical information to update the discourse context presupposed by an utterance. Kaplan considered the information in a database and applied language-driven inferences to identify *presumptions* in database queries, and generate indirect cooperative responses, e.g., "CS101 was not offered" rather than "nobody passed CS101". Motro extended this work using information about the database, such as integrity constraints, in addition to the information in the database. The presuppositions considered by these researchers are typically few and can be unequivocally inferred from the wording of a single statement. In contrast, the suppositions considered in this paper are postulated to justify beliefs stated by a user that differ from what our system thinks that the user believes. Furthermore, there may be several alternative suppositions that explain a user's statements, and their probability depends on the other beliefs held by the user.

Our work on positing suppositions is also related to research on the recognition of flawed plans (Quilici 1989; Pollack 1990; Chu-Carroll and Carberry 2000), in the sense that we attempt to justify a user's statements (i.e., why does the user think X?). Pollack focused on invalid plans in the context of a consultation system that generated helpful responses. In order to infer these plans, her system assumed the availability of a model of the user's beliefs, including erroneous beliefs. Such a model is produced by BIAS from the user's interaction with the system (Zukerman and George 2005) and by positing suppositions. Quilici investigated the recognition of plan-oriented misconceptions in advice-seeking dialogues. This was done by applying a set of justification rules that started from a user's stated belief and arrived at a supporting, possibly erroneous premise. Chu-Carroll and Carberry applied a plan-based approach to identify erroneous beliefs that account for a user's statements during conflict-resolution dialogues. They then selected propositions to be mentioned by their system in its counter-argument based on the envisaged impact of candidate propositions on the user's erroneous statements. The main difference between BIAS and the last two systems is that they infer a plan from a user's utterances and context, and use the plan itself to postulate the user's beliefs that differ from the system's beliefs. In contrast, BIAS infers a Bayesian subnet, rather than a plan, from the user's utterances, and considers its entire user model when postulating discrepant beliefs (i.e., suppositions). In addition, in BIAS, the selection of a supposition is integrated into our probabilistic formalism for choosing an interpretation, instead of being performed as a stand-alone operation.

10 Discussion

We have offered an integrated approach for the generation of argument interpretations in the context of a Bayesian argumentation system. Our interpretations, which are viewed as explanations of a user's argument, comprise the following elements: (a) a Bayesian subnet that connects the propositions mentioned by a user, (b) suppositions that explain the user's beliefs, and (c) explanatory extensions which complement an interpretation with information that the user may have considered (but omitted from his or her argument).

We have presented an anytime algorithm for the generation of candidate interpretations which activates procedures for postulating suppositions, constructing Bayesian subnets (Zukerman and George 2005), and proposing explanatory extensions. We have also extended our formalism for the calculation of the probability of an interpretation (Zukerman and George 2005) to incorporate suppositions and explanatory extensions. This probability reflects the similarity between an interpretation and the domain, and between the argument and the interpretation, rather than the peculiarities of a particular domain BN. Nonetheless, there are domain characteristics that influence this probability—explained below.

The evaluation of our module for positing suppositions shows that our trial subjects found BIAS' suppositions to be both necessary and reasonable, with its preferred suppositions being top-ranked or top-2 ranked by most subjects. The evaluation of our module for generating explanatory extensions shows that interpretations with such extensions were clearly preferred to interpretations without them.

Our formalism for calculating the probability of an interpretation is based on the idea of cost versus reward, which balances the cost of adding extra elements (e.g., suppositions and explanatory extensions) to an interpretation against the benefits obtained from these elements. The calculations that implement this idea are based on a few general principles: (1) combinatoric principles for extracting an interpretation graph from the domain BN; (2) known distributions, such as Poisson for the number of nodes in an interpretation graph or explanatory extension, and Zipf for modelling discrepancies in belief; and (3) distributions which model preference heuristics, e.g., for different suppositions in the context of a user's beliefs, and for different types of inferences. The parameterization of these distributions requires specific information. For instance, the mean of the Poisson distribution determines the "penalty" for having too many nodes in an interpretation or explanatory extension. Similarly, the hand-tailored distributions require fine-tuning the probabilities that express people's preferences, or designing experimental studies to gather accurate probabilities that express these preferences (Elzer et al. 2005).

The immediate applicability of our approach is mainly affected by our assumption that the nodes in the BNs are binary, which influences all aspects of our implementation. Other factors that must be considered when applying our formalism are: the characteristics of the domain, the expressive power of BNs, and the ability of users to interact with the system.

Binary nodes in BNs. A limitation to the applicability of our implemented formalism is the assumption that the nodes in the domain BN are binary. The effect of this assumption is that knowing the probability of the value of a node (say $\Pr(\mathsf{TimeOfDeath11}) = 0.6$) automatically implies the probability of its other value ($\Pr(\neg\mathsf{TimeOfDeath11}) = 0.4$). This assumption simplifies our procedure for calculating the probability of suppositions, but it does not invalidate the principles behind the process for generating suppositions and assigning probabilities to them (Sect. 4). The generalization of the procedure for calculating the probability of suppositions is relatively simple. For example, supposing that a node has its most probable value would still have the highest probability, and supposing that it has its least probable value would have the lowest probability. Supposing other values would then have intermediate probabilities. To address this limitation we would have to extend the probability distribution in Table 2 to reflect this range of options.

The binary-node assumption also affects explanatory extensions, as the preferences we identified regarding acceptable inferences were restricted to binary-nodes and monotonic inferences (Sect. 6). The relaxation of the binary-node assumption requires the extension of our definition of acceptable inferences. For example, the preferences we identified could generalize to non-binary nodes with ordinal values (e.g., high, medium and low).

The relaxation of the binary-node assumption has a further impact with respect to the search space, which would be significantly extended. Specifically, for suppositions, each value of a node would be a candidate supposition. Thus, instead of having a search space of 3^N suppositions for a BN with N nodes, now the search space would be $(k_1+1)\times(k_2+1)\times\cdots\times(k_N+1)$, where k_i is the number of values for node i. Experimental studies would then be required to determine the effect of an increase in the search space on the performance of our anytime algorithm, and to appropriately parameterize this algorithm (e.g., how many supposition configurations to select as seeds, Sect. 4).

Domain characteristics. Explanatory extensions are domain dependent, as they are generated to explain surprising outcomes, and what is surprising often depends on the domain. For instance, in some domains what matters is the increase or reduction in probability, rather than its absolute value, e.g., an increase from 6 to 10% in the probability of a patient having cancer may require an explanatory extension, even though in absolute terms both probabilities belong to the VeryProbablyNot belief category. Also, the values of nodes may be elements of a set (e.g., red, white, blue). The idea of inference monotonicity is not applicable to sets, thereby precluding the identification of regularities such as those identified in our survey (Sect. 6). These observations, together with the relaxation of the binary-node assumption, affect several factors in the generation of explanatory extensions, e.g., the probability categories that are considered significant, whether absolute probabilities or relative change from previous values should be considered, which preferences generalize across different nodes and which should be encoded specifically.

Expressive power of BNs. Although the BNs employed in our work were hand designed, BNs can be automatically learned from data (Wallace 2005). Our domain knowledge is represented according to the state of the art for application BNs, which represent ground predicates. This influences our argument interpretation capabilities, which support only such predicates.

Two complementary avenues for resolving this issue have been considered in the literature. Charniak and Goldman (1993), Gertner (1998), and Elzer et al. (2005) built BNs on the fly based on the features of a situation. Getoor et al. (2001) and Taskar et al. (2002) studied probabilistic relational models, which combine advantages of relational logic and BNs, and can generalize over a variety of situations. The first approach is suitable for our situation, where different instances of a predicate can be dynamically instantiated in a BN. However, such an approach still does not support arguments based on quantified predicates. The interpretation of such arguments requires both further research on BNs, and corresponding enhancements to our approach.

Interacting with BNs. Although this paper focuses on argumentation, our formalism enables users to interact with BNs outside the argumentation context. That is, our formalism allows users to reach a “mutual understanding” with domain BNs, in the sense that users can present their thoughts about a situation, and the system can attempt to cast these thoughts in the context of a domain BN. However, this requires an underlying assumption of a common ground between the users and the knowledge representation. In addition, users must be able to input their ideas. As indicated in Sect. 2, we have addressed this problem by providing a web interface that restricts users to the propositions known to the system. However, usability studies are required to determine the general feasibility of this approach.

Summary. The main contributions of this article are: an extension of the definition of an interpretation that includes suppositions and explanatory extensions, a general formalism for calculating the probability of an interpretation, and procedures for postulating suppositions and including explanatory extensions. Explanatory extensions are influenced by the characteristics of the domain and require further examination. However, the probabilistic formalism and the procedure for positing suppositions are domain independent, and, with the relaxation of the binary-node assumption, generally applicable. We are currently in the process of applying our formalism for calculating the probability of an interpretation to spoken interactions with robots, which requires us to represent open-ended settings (where all the propositions in the domain cannot be encoded in advance).

Appendix

A Calculating the effect of explanatory extensions on the probability of an interpretation

The influence of explanatory extensions on the probability of an interpretation is represented by the following product, which expresses the cost-reward balance obtained by including a (possibly empty) explanatory extension in an interpretation.

$$\Pr(\text{beliefs in } IG_i|SC_i, \text{struct of } IG_i, EE_i) \times \Pr(EE_i|SC_i, \text{struct of } IG_i) \tag{A.1}$$

The calculation of the two probabilities in this product is described in the following sub-sections.

A.1 Estimating Pr(beliefs in $IG|SC$, struct of IG, EE)

As stated in Sect. 6.2.1, the probability of the beliefs in IG is a function of the discrepancy between peoples' preferences regarding which beliefs are appropriate for the consequent of the inferences in IG given the antecedents, and the actual belief in the consequent.

Following (Zukerman and George 2005), we use the Zipf distribution to model these discrepancies in belief for each inference in an interpretation graph, where the parameter of the distribution is the difference between the belief category of the consequent and a desired belief category. However, here we do not have a single desired category, but a range of desired belief categories, denoted as *acceptable range*, which includes the beliefs specified by *BothSides, SameSide* and *AlmostSame* inferences (Sect. 6).

Definition 1 *(***Acceptable Range***)*
$AcceptableRange = [\min_{A \in Anteced's}\{BelCat(A)\}, \max_{A \in Anteced's}\{BelCat(A)\}]$
If *AcceptableRange* is on the same side of EvenChance
Then extend *AcceptableRange* by one category towards EvenChance

The parameter of our Zipf distribution is 0 if the consequent of the inference is inside the acceptable range. Otherwise, it is the absolute value of the difference between the belief

category of the consequent and the closest belief category in the acceptable range. This parameter, called *difRange*, is calculated as follows.
For *Inference* = [{*Antecedents*}→*Cons*],
difRange(*Inference*)

$$= \begin{cases} |BelCat(Cons) - \min Acct'bleRange| & \text{if } BelCat(Cons) < \min Acct'bleRange \\ |BelCat(Cons) - \max Acct'bleRange| & \text{if } BelCat(Cons) > \max Acct'bleRange \\ 0 & \text{otherwise} \end{cases}$$

The following formula is then used to calculate the probability of the beliefs in *Inference*.

$$\Pr(\text{beliefs in } \mathit{Inference}) = \begin{cases} \dfrac{|AcceptableRange| \times \theta}{(difRange(Inference) + 1)^{\gamma}} & \text{if } BelCat(Cons) \in AcceptableRange \\ \dfrac{\theta}{(difRange(Inference) + 1)^{\gamma}} & \text{otherwise} \end{cases} \quad \text{(A.2)}$$

where $|AcceptableRange|$ is the size (number of belief categories) of the acceptable range, $\gamma = 2$ is an empirically determined parameter, and θ is a normalizing constant.[9] This probability distribution views the union of the belief categories in the acceptable range as a single block (the belief categories inside this block do not compete with each other). A side effect of this distribution is that a consequent inside a large acceptable range has a higher probability than a consequent inside a small range, and a consequent outside a large acceptable range has a lower probability than a consequent outside a small range. Alternative formulations for the impact of the size of an acceptable range on the probability of an inference are left for future research.

The probability of the beliefs in an interpretation graph is obtained by multiplying the belief probability for each inference in the graph.

$$\Pr(\text{beliefs in } IG|SC, \text{struct of } IG, EE) = \prod_{j=1}^{NF} \Pr(\text{beliefs in } \mathit{Inference}_j)$$

where NF is the number of inferences in IG.

A.2 Estimating $\Pr(EE|SC, \text{struct of } IG)$

As stated in Sect. 6.2.2, the calculation of $\Pr(EE|SC, \text{struct of } IG)$ is based on the following heuristics.

- Users prefer short explanations, but not at the expense of persuasiveness.
- Explanatory extensions that yield *BothSides* inferences are preferred to explanatory extensions that yield *SameSide* or *AlmostSame* inferences, which are much preferred to explanatory extensions that produce *SmallNonSeq* inferences.

These heuristics are implemented in the following equation.

$$\Pr(EE|SC, \text{struct of } IG) = \prod_{j=1}^{NF} \Pr(InfCategory(EE_j), s(EE_j)|SC, \text{struct of } IG) \quad \text{(A.3)}$$

[9] The denominator is incremented by 1 to avoid division by 0. If the belief category of the consequent is in the acceptable range, the denominator is $(0+1)^{\gamma} = 1$.

where NF is the number of inferences in IG, $s(EE_j)$ is the number of siblings included in the explanatory extension for the jth inference in IG, and $\mathit{InfCategory}(EE_j)$ is the category of the inference resulting from the inclusion of these siblings (*BothSides, SameSide, AlmostSame* or *SmallNonSeq*).

Applying the chain rule of probability yields

$$\Pr(EE|SC, \text{struct of } IG) = \prod_{j=1}^{NF} \left\{ \begin{array}{l} \Pr(\mathit{InfCategory}(EE_j)|s(EE_j), SC, \text{struct of } IG) \\ \times \Pr(s(EE_j)|SC, \text{struct of } IG) \end{array} \right\} \tag{A.4}$$

These probabilities are calculated as follows.

– $\Pr(s(EE_j)|SC, \text{struct of } IG)$ is the probability of having $s(EE_j)$ siblings in an explanatory extension for inference j in IG. As in (Zukerman and George 2005), we model this probability by means of a truncated Poisson distribution, *Poisson*(μ), where μ is the average number of nodes in an explanatory extension. According to our first heuristic, explanatory extensions should be concise, but not necessarily minimal (Sect. 6). Hence, we use $\mu = 1.5$, which penalizes explanatory extensions with many nodes.

$$\Pr(s(EE_j)|SC, \text{struct of } IG) = \begin{cases} \delta \dfrac{e^{-\mu} \mu^{s(EE_j)}}{s(EE_j)!} & \text{if } s(EE_j) \le S \\ 0 & \text{otherwise} \end{cases} \tag{A.5}$$

where δ is a normalizing constant, and S is the maximum number of siblings for a node in the domain BN. We use the maximum number of siblings in the BN, instead of the actual number of siblings of the antecedents of the inference in question, so that we can have an absolute measure of the cost of an explanatory extension. Such a measure expresses our users' preference for short explanations (irrespective of the number of propositions that could be included in an explanation), and enables us to compare different interpretations where explanatory extensions have been added to different inferences.

– $\Pr(\mathit{InfCategory}(EE_j)|s(EE_j), SC, \text{struct of } IG)$ is the probability of the category of an inference after adding an explanatory extension.[10] We devised the following probability distribution to represent people's preferences for different categories of inferences, as specified by the second heuristic above.

$$\Pr(\mathit{InfCategory}(EE_j)|s(EE_j)) = \begin{cases} 0.5 \text{ if } \mathit{InfCategory}(EE_j) \text{ is } \mathit{BothSides} \\ 0.4 \text{ if } \mathit{InfCategory}(EE_j) \text{ is } \mathit{SameSide} \text{ or } \mathit{AlmostSame} \\ 0.1 \text{ if } \mathit{InfCategory}(EE_j) \text{ is } \mathit{SmallNonSeq} \end{cases} \tag{A.6}$$

This distribution pertains to beliefs in the positive range (a similar probability distribution exists for beliefs in the negative range). According to this distribution, an explanatory extension yielding a *BothSides* inference has a higher probability than one yielding a *SameSide* or *AlmostSame* inference, which in turn has a significantly higher probability than one yielding a *SmallNonSeq* inference.[11] These probabilities were generated manually on the

---

[10] If the explanatory extension is empty, the category of the inference is irrelevant. In this case, we assign it a probability of 1.

[11] We have also considered an approach where the probability of including a sibling in an explanatory extension balances the sibling's salience against its effect on the inference. However, the nature of this trade-off is unclear, e.g., should a *SameSide* and salient sibling be included in preference to a *BothSides* and non-salient sibling? Therefore, at present we revert to a simpler approach.

basis of our surveys. However, these surveys were qualitative, hence similar distributions may work just as well.

References

Charniak, E., Goldman, R.: A Bayesian model of plan recognition. Artif. Intell. **64**(1), 53–79 (1993)

Chu-Carroll, J., Carberry, S.: Conflict resolution in collaborative planning dialogues. Int. J. Human Comput. Stud. **6**(56), 969–1015 (2000)

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

Druzdzel, M.: Qualitative verbal explanations in Bayesian belief networks. Artif. Intell. Simul. Behav. Quarterly **94**, 43–54 (1996)

Elzer, S., Carberry, S., Zukerman, I., Chester, D., Green, N., Demir, S.: A probabilistic framework for recognizing intention in information graphics. In: IJCAI05 Proceedings—the Nineteenth International Joint Conference on Artificial Intelligence, pp. 1042–1047. Edinburgh, Scotland (2005)

Gardent, C., Kow, E.: Generating and selecting grammatical paraphrases. In: ENLG-05 — Proceedings of the Tenth European Workshop on Natural Language Generation, pp. 49–57. Aberdeen, Scotland (2005)

George, S., Zukerman, I., Niemann, M.: An anytime algorithm for interpreting arguments. In: PRICAI2004—Proceedings of the Eighth Pacific Rim International Conference on Artificial Intelligence, pp. 311–321. Auckland, New Zealand (2004)

George, S., Zukerman, I., Niemann, M.: Modeling suppositions in users' arguments. In: UM05—Proceedings of the 10th International Conference on User Modeling, pp. 19–29. Edinburgh, Scotland (2005)

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

Getoor, L., Friedman, N., Koller, D., Taskar, B.: Learning probabilistic models of relational structure. In: Proceedings of the Eighteenth International Conference on Machine Learning, pp. 170–177. Williamstown, Massachusetts (2001)

Grice, H.P.: Logic and conversation. In: Cole P., Morgan J. (eds.) Syntax and Semantics, Volume 3: Speech Acts, pp. 41–58. Academic Press (1975)

Gurney, J., Perlis, D., Purang, K.: Interpreting presuppositions using active logic: from contexts to utterances. Comput. Intell. **13**(3), 391–413 (1997)

Hobbs, J.R., Stickel, M.E., Appelt, D.E., Martin, P.: Interpretation as abduction. Artif. Intell. **63**(1–2), 69–142 (1993)

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

Horvitz, E., Suermondt, H., Cooper, G.: Bounded conditioning: flexible inference for decision under scarce resources. In: UAI89—Proceedings of the 1989 Workshop on Uncertainty in Artificial Intelligence, pp. 182–193. Windsor, Canada (1989)

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

Joshi, A., Webber, B.L., Weischedel, R.M.: Living up to expectations: computing expert responses. In: AAAI84—Proceedings of the Fourth National Conference on Artificial Intelligence, pp. 169–175. Austin, Texas (1984)

Kahneman, D., Slovic, P., Tversky, A.: Judgment under Uncertainty: Heuristics and Biases. Cambridge University Press (1982)

Kaplan, S.J.: Cooperative responses from a portable natural language query system. Artif. Intell. **19**, 165–187 (1982)

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

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

Mercer, R.E: Presuppositions and default reasoning: a study in lexical pragmatics. In: Pustejovski, J., Bergler, S. (eds.) ACL SIG Workshop on Lexical Semantics and Knowledge Representation (SIGLEX), pp. 321–339. Berkeley, California (1991)

Motro, A.: SEAVE: a mechanism for verifying user presuppositions in query systems. ACM Trans. Inform. Syst. (TOIS) **4**(4), 312–330 (1986)
Ng, H., Mooney, R.: On the role of coherence in abductive explanation. In: AAAI-90—Proceedings of the Eighth National Conference on Artificial Intelligence, pp. 337–342. Boston, Massachusetts (1990)
Paiva, D.S.: Investigating NLG architectures: taking style into consideration. In: EACL'99—Proceedings of the Ninth Conference of the European Chapter of the Association for Computational Linguistics, pp. 237–240. Bergen, Norway (1999)
Pearl, J.: Probabilistic Reasoning in Intelligent Systems. Morgan Kaufmann Publishers, San Mateo, California (1988)
Pollack, M.: Plans as complex mental attitudes. In: Cohen, P., Morgan, J., Pollack, M. (eds.) Intentions in Communication, pp. 77–103. MIT Press (1990)
Quilici, A.: Detecting and responding to plan-oriented misconceptions. In: Kobsa, A., Wahlster, W. (eds.) User Models in Dialog Systems, pp. 108–132. Springer-Verlag (1989)
Taskar, B., Abbeel, P., Koller, D.: Discriminative probabilistic models for relational data. In: Proceedings of the Eighteenth Conference on Uncertainty in Artificial Intelligence (UAI), pp. 485–490. Alberta, Canada (2002)
van Beek, P.: A model for generating better explanations. In: Proceedings of the Twenty-Fifth Annual Meeting of the Association for Computational Linguistics, pp. 215–220. Stanford, California (1987)
Wallace, C.: Statistical and Inductive Inference by Minimum Message Length. Springer, Berlin, Germany (2005)
Zukerman, I., George, S.: A probabilistic approach for argument interpretation. User Model. User-Adapted Interact. Special Issue Lang-Based Interact. **15**(1–2), 5–53 (2005)
Zukerman, I., George, S., George, M.: Incorporating a user model into an information theoretic framework for argument interpretation. In: UM03—Proceedings of the Ninth International Conference on User Modeling, pp. 106–116. Johnstown, Pennsylvania (2003)
Zukerman, I., Jitnah, N., McConachy, R., George, S.: Recognizing intentions from rejoinders in a Bayesian interactive argumentation system. In: PRICAI2000—Proceedings of the Sixth Pacific Rim International Conference on Artificial Intelligence, pp. 252–263. Melbourne, Australia (2000)
Zukerman, I., McConachy, R.: WISHFUL: a discourse planning system that considers a user's inferences. Comput. Intell. **1**(17), 1–61 (2001)
Zukerman, I., Niemann, M., George, S.: Improving the presentation of argument interpretations based on user trials. In: AI'04—Proceedings of the 17th Australian Joint Conference on Artificial Intelligence, pp. 587–598. Cairns, Australia (2004)

Authors' vitae

Sarah George completed a Computer Science degree with Honours at Monash University. During and after her studies she worked as a Research Fellow for Ingrid Zukerman. She is a "programming generalist" interested in systems, abstraction and language. Her curiosity about language as knowledge representation led her to her work in user modelling and natural language. Sarah now works on a different kind of knowledge representation and retrieval, with the project hosting service CVSDude.

Ingrid Zukerman is a 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. Since then, she has been working in the Faculty of Information Technology at Monash University. Her areas of interest are discourse planning and interpretation, plan recognition, dialogue, and question answering.

Michael Niemann is currently a Research Fellow working with Ingrid Zukerman. He received a B.Sc. in Mathematics and Computer Science and B.A. in Linguistics from the University of Melbourne. His subsequent Arts Honours year focused on Computational Linguistics. Before joining Monash, he played a major part in the design and development of two commercial machine translation systems. His fields of interest include syntactic and semantic formalisms and human–computer interaction.