Bayesian Reasoning in an Abductive Mechanism for Argument Generation and Analysis

Ingrid Zukerman, Richard McConachy and Kevin B. Korb
School of Computer Science and Software Engineering, Monash University
Clayton, Victoria 3168, AUSTRALIA
email: {ingrid, ricky, korb} @csse.monash.edu.au

Abstract
Our argumentation system, NAG, uses Bayesian networks in a user model and in a normative model to assemble and assess arguments which balance persuasiveness with normative correctness. Attentional focus is simulated in both models to select relevant subnetworks for Bayesian propagation. The subnetworks are expanded in an iterative abductive process until argumentative goals are achieved in both models, when the argument is presented to the user.

Introduction
In this paper, we describe the operation of our argument generation-analysis system, NAG (Nice Argument Generator). Given a goal proposition, NAG generates nice arguments, i.e., arguments that are normatively strong while also being persuasive for the target audience. NAG also analyzes users' arguments, and prepares rebuttals if appropriate. The focus of this paper is on the generation aspect of our work.

Figure 1 shows the main modules of NAG. The Strategist may receive as input a goal proposition or a user-generated argument. During argument generation, it activates a generation-analysis cycle as follows (§ Generation-Analysis Cycle). Firstly, it uses semantic activation to quickly form an initial Argument Graph for an argument, or to quickly extend an already existing Argument Graph. An Argument Graph is a network with nodes that represent propositions, and links that represent the inferences that connect these propositions. The Strategist then calls the Generator to continue the argument building process (§ Extending the Argument Graph). The Generator fleshes out the Argument Graph by activating Reasoning Agents to consult several sources of information, and incorporating the inferences and propositions returned by these agents into the Argument Graph. This Argument Graph is returned to the Strategist, which passes it to the Analyzer (§ Argument Analysis) to evaluate its niceness.

To estimate the persuasive power of an argument represented by an Argument Graph, the Analyzer consults a revisable user model that reflects the beliefs of the target audience; a normative model is used to gauge the normative strength of an argument. Belief updating in both the user and the normative model is done by a constrained Bayesian propagation scheme. If the Analyzer reports that the Argument Graph is nice enough, the Strategist presents an argument based upon this graph to the user (§ Argument Presentation). Otherwise, the Analyzer highlights the weaknesses to be fixed in the argument, and the Argument Graph is returned to the Strategist for another cycle of argument extension and analysis. This process is typically performed more than once before the argument is presented to the user. It iterates until a successful Argument Graph is built, or NAG is unable to continue, e.g., due to time running out or failing to find further evidence.

Knowledge Representation
When constructing an argument, NAG relies on a normative model composed of different types of Knowledge Bases (KBs) and a user model also composed of different types of KBs which represent the user's presumed beliefs and inferences. A single KB represents information in one form, e.g., a semantic network (SN), Bayesian network (BN), rule-based system or database. During argument generation, relevant material from several KBs may need to be combined into a common representation. We have chosen BNs for this purpose because of their ability to represent normatively correct reasoning under uncertainty.

When constructing an Argument Graph, NAG develops two BNs: the BN forming one of the KBs in the user model, and the BN forming one of the KBs in the normative model. As arguments are built up, material obtained from other KBs...
may be converted to BN form and added to the appropriate BN, e.g., material from a rule-based system in the user model may be added to the user model BN (§ Extending the Argument Graph). To reduce the amount of information NAG must deal with, we apply a focusing mechanism which highlights the portion of the complete BN in each model that is needed for the current argument (§ Focusing the Argument). Hence, each of the user model and the normative model contains a single Bayesian subnetwork that is in focus. The structural intersection of these Bayesian subnetworks forms the Argument Graph. When analyzing this graph, propagation is performed twice, once over the Bayesian subnetwork in the user model and once over the Bayesian subnetwork in the normative model, each time using probabilistic information sourced from within the model being propagated (§ Argument Analysis). Thus, we measure the strength of the same argument in the user model and the normative model.

### Generation-Analysis Cycle

NAG receives the following inputs: (1) a proposition to be argued for; (2) an initial argument context; and (3) two target ranges of degrees of belief to be achieved (one each for the normative model and the user model). The initial argument context, denoted $context_i$, is composed of the propositions and concepts mentioned in a preamble to the argument plus the argument's goal; this context is expanded as the Argument Graph grows. The degrees of belief to be achieved are expressed as ranges of probabilities, e.g., $[0.5, 0.6]$, in order to be able to represent a variety of goals, e.g., inducing doubt when a strong belief is inappropriate.

Upon completion of the argumentation process, the Strategist produces an Argument Graph which starts from admissible premises and ends in the goal proposition. Admissible premises are propositions that start out being believed by NAG and the user (sufficiently for the argument to work).

The Strategist executes the following algorithm during argument generation. In principle, this procedure is applicable to any proposition, and hence also to special forms such as premises and modal propositions. However, it does not currently have facilities to treat these forms in any special way.

### Generation-Analysis Algorithm

1. $i \leftarrow 0$.
2. Clamp any items in the current context, $context_i$, and perform spreading activation. This yields an Argument Graph containing: the clamped nodes, the activated nodes (whose activation exceeds a threshold), plus the links connecting these nodes (§ Focusing the Argument).
3. Identify new subgoals in the current Argument Graph (§ Choosing Argument Subgoals).
4. Pass the argument subgoals identified in Step 3 to the Generator, which adds the new information returned by its Reasoning Agents to the current Argument Graph (§ Extending the Argument Graph).
5. Pass the Argument Graph generated in Step 4 to the Analyzer for evaluation (§ Argument Analysis).
6. If the Analyzer reports that the current Argument Graph is sufficiently nice, then present an argument based on this graph to the user, and wait for a response (§ Argument Presentation). Otherwise, continue.
7. $i \leftarrow i + 1$.
8. $context_i \leftarrow context_{i-1} +$ new nodes connected to the goal during cycle $i-1$.
9. Go to Step 2.

When receiving a user's argument, an analysis-generation cycle is activated. This cycle begins in Step 5, which results in the acceptance of the user's argument if no flaws are detected. Otherwise, the cycle is completed, and the generation part of the cycle is performed (Steps 2, 3 and 4) to try to bridge small inferential gaps in the user's argument. This cycle is repeated only a few times, since large gaps in a user's argument make it more likely that NAG and the user are using different lines of reasoning.

### Focusing the Argument

Bayesian network propagation (Pearl 1988) is an NP-hard problem in the general case (Cooper 1990). NAG is designed to be an interactive system, potentially drawing upon very large knowledge bases, so complete propagation over large BNs is not feasible. In addition, NAG's user model is designed to model human cognitive abilities, and humans normally cannot absorb and analyze all data relevant to a complex problem. To cope with both of these limits on complexity we emulate the principal means available to humans for applying limited cognitive capacity to problem solving, namely attention (see, for example, Baars 1987).

NAG uses two hierarchical SNs, one built on top of the user model BN and one built on top of the normative model BN, to capture associative connections between information items (Figure 2 illustrates a semantic-Bayesian network). The initial semantic-Bayesian networks are currently built manually, but they may be automatically extended during argument generation (§ Extending the Argument Graph). Both the SN and the BN are used by NAG to simulate attentional focus in each model. However, the resulting Argument Graph contains only propositions and links from the BN.

NAG takes the context in which the argument occurs as an initial set of salient objects. For example, if the user presents an argument to NAG, the concepts occurring in the propositions within the argument or in the preceding...
For example, assume NAG’s qualitative rule-based system agent finds a rule stating “If D then E is possible”. If the agent responsible for quantitative rule-based systems also finds the rule “If D then E with prob = \( p \)”, which NAG

### Extending the Argument Graph

The initial Argument Graph consists of the subset of the BNs which was activated by the attentional mechanism. The Generator then activates the Reasoning Agents to collect information relevant to each subgoal in the current Argument Graph. Upon their return, the Generator must determine: (1) which newly returned inferences should be integrated into the Argument Graph; (2) the structure of the additions to the Argument Graph representing the new inferences; and (3) the parameters of the new inferences and propositions.

#### Which propositions and inferences to integrate.

New propositions returned by the Reasoning Agents are added to the current Argument Graph as new nodes. NAG decides whether to introduce new inferences returned by the Reasoning Agents into the Argument Graph (or to replace existing inferences with new ones) by applying the following rules, which ensure that each relationship between propositions in the Argument Graph is represented only once:

1. **At most one inference may directly connect any two propositions in the Bayesian subnetwork in each of the user model and the normative model.**

2. **When selecting from multiple candidate inferences, prefer inferences sourced from more expressive representations, where expressiveness means how much probabilistic information can be expressed by the representation.**

For example, assume NAG’s qualitative rule-based system agent finds a rule stating “If D then E is possible”. If the agent responsible for quantitative rule-based systems also finds the rule “If D then E with prob = \( p \)”, which NAG
Argumentation

there is no reason to prefer the prior probabilities obtained from one

Extending the Graph - Example Continued

Structure of the new propositions and inferences. The

Adding parameters for the propositions and inferences. Normally,

Adding information to the Argument Graph about joint
c conditional probabilities associated with new inferences is
done as follows. If a Reasoning Agent can provide

Adding to the Argument Graph about joint
c conditional probabilities associated with new inferences is
done as follows. If a Reasoning Agent can provide

Extending the Graph - Example Continued

In this step, the information returned by the Reasoning

(1) NAG does not try to merge information gleaned from more

than one available source since it is unclear how to do so.

Since we are not modeling the reliability of the various KBs,

there is no reason to prefer the prior probabilities obtained from one

KB to conflicting priors obtained from another. Thus, we retain

whatever information is already in the BN.

Figure 4: Argument Graph for the Asteroid Example

N6 \rightarrow N2 have not been discovered yet). All the links

returned by the Reasoning Agents are causal or evidential, as

these are the only types of relations incorporated at present

in the arguments generated by NAG. Some of this information

will be included in the final Argument Graph presented
to the user, e.g., the newly found node N6 and the link

connecting N4 \rightarrow N6, while other information, e.g., node N5,

will be eventually excluded (§ Argument Presentation).

Upon completion of this step, the Argument Graph consists of
two separate subgraphs: one containing nodes N5-N14 and

another containing nodes N1-N4.

Argument Analysis

The process of computing the anticipated belief in a goal

proposition as a result of presenting an argument starts with

the belief in the premises of the Argument Graph and ends

with a new degree of belief in the goal proposition. The

Analyzer computes the new belief in a proposition by com-
bining the previous belief in it with the result of applying

the inferences which precede this proposition in the Argu-

ment Graph. This belief computation process is performed

by applying propagation procedures to the Bayesian sub-

network corresponding to the current Argument Graph in the

user model and separately to the subnetwork corresponding
to the current Argument Graph in the normative model.

In propagating only over the subnetworks initially seed-
ed by the focusing mechanism (§ Focusing the Argument)

and extended with information returned by the Reasoning

Agents (§ Extending the Argument Graph), NAG ignores

those parts of the complete BNs in the user and normative

models not deemed relevant to the current argument. Prop-

agating over the subnetwork corresponding to the current

Argument Graph in the user and normative models is much

faster than having to perform propagation over the complete

BN in each model, but the trade off is a less accurate

estimate of the final belief in the goal proposition. Still, in a

system designed to be interactive, some such trade off is

necessary in view of the complexity of Bayesian propagation.

After propagation, the Analyzer returns the following

measures for an argument: its normative strength, which

is its effect on the belief in the goal proposition in the nor-

836 Argumentation
mative model, and its persuasiveness, which is its effect on the user’s belief in the goal proposition (estimated according to the user model). Of course, an argument’s persuasiveness may be quite different from its normative strength.

After the Analyzer has evaluated the normative strength and persuasiveness of the Argument Graph it returns an assessment, which points out any propositions within the Argument Graph that are not sufficiently supported. The generation of support for such propositions is automatically handled by the Generation-Analysis algorithm as follows. Propositions that became connected to the goal during the current cycle are automatically added to the context (Step 8 of the Generation-Analysis algorithm). These propositions are clamped in Step 2 of the next cycle, and those which have not been previously passed to the Reasoning Agents are identified as subgoals (Step 3). It is possible that some propositions will remain insufficiently supported after being investigated by the Reasoning Agents. Often, these propositions are eventually removed from the Argument Graph after alternative, stronger subarguments have been found (§ Argument Presentation).

After integrating the new subarguments into the Argument Graph, the now enlarged Argument Graph is again sent to the Analyzer for inspection. Thus, by completing additional focusing-generation-analysis cycles, NAG can often improve Argument Graphs that are initially unsatisfactory.

Analyzing the Graph – Example Continued

The argument that can be built at this stage has three main branches: (1) from nodes $N_5$, $N_6$ and $N_8$ to $N_9$ and then $N_{12}$, (2) from $N_7$ to $N_{10}$, then $N_9$ and then $N_{12}$, and (3) from $N_{11}$ and $N_{14}$ to $N_{13}$ and then $N_{12}$. However, only $N_7$ is currently believed by the user, hence it is the only admissible premise among the potential premise nodes. Thus, the anticipated final belief in the goal node in both the normative and the user model falls short of the desired ranges. This is reported by the Analyzer to the Strategist. Nodes $N_3$, $N_6$, $N_{8}$, $N_{11}$, $N_{13}$ and $N_{14}$ are now added to the context (which initially consisted of $N_1$, $N_3$, $N_7$ and $N_{12}$), and the next cycle of the Generation-Analysis algorithm is activated. After spreading activation (Step 2), several nodes become active. However, the main node of interest in this example is $N_2$, which is activated by $N_1$, $N_3$ and $N_6$. The activation from $N_6$ results in the argument fragment composed of nodes $N_1$–$N_4$ being linked to the goal. The subgoal selection step (Step 3) now identifies nodes $N_2$, $N_3$, $N_5$, $N_6$, $N_8$, $N_{10}$, $N_{11}$ and $N_{14}$ as subgoals to be passed to the Generator, since these nodes now have a path to the goal node and have not been previously passed to the Reasoning Agents. These agents can find the following new information only: node $N_{13}$ and links $N_6 \rightarrow N_2$ and $N_{11} \rightarrow N_{15}$. The resulting Argument Graph is returned to the Analyzer again (Step 5), which determines that the anticipated belief in the goal is now within the target ranges in both models. Thus, the Argument Graph can be passed to the presentation step.

During argument presentation, NAG attempts to minimize the size of the Argument Graph by searching for the subgraph with the fewest nodes which still yields a sufficiently nice argument. During this process, it tries to make the argument more concise by iteratively deleting nodes and invoking the Analyzer to determine whether the belief in the goal proposition in the now smaller Argument Graph still suffices. This process is desirable since, upon completion of the generation-analysis cycles, some of the propositions in the Argument Graph may be supported more strongly than is necessary for the argument to work.

The subgraph corresponding to an argument generated for the asteroid example is outlined with a dashed box in Figure 4. Propositions $N_4$, $N_5$, $N_9$ and $N_{14}$ are omitted because of their weak contribution to the goal. The subgraph composed of $N_1 \rightarrow N_7 \rightarrow N_{12}$ is omitted because the desired belief ranges can be achieved without it.

At present, the resulting Argument Graph can be rendered graphically through a graphical interface which allows users to build and receive arguments in an annotated network form. Methods for rendering NAG’s output in English, such as those described in (Huang & Fiedler 1997; Reed & Long 1997), are also being considered.

Evaluation

A preliminary Web-based evaluation of NAG’s output was conducted by a pre-test elicitation of subjects’ beliefs regarding the following key propositions in the argument in Figure 4: $(N_{12})$ a large asteroid struck the Earth about 65 million years ago, $(N_2)$ there was a sudden cooling of the Earth’s climate about 65 million years ago, $(N_{11})$ iridium is abundant in the Earth’s crust, $(N_7)$ iridium is abundant in asteroids (the last two factors are related to node $N_7$). According to the replies, an argument was selected among several options previously generated by NAG. For instance, if a respondent indicated belief in $N_2$, then a subargument supporting this proposition was omitted. After presenting an argument to a respondent, a post-test was administered to assess changes in belief in the pre-test propositions.

Among the 32 respondents, there was a clear tendency to shift belief towards the (final and intermediate) targets in response to NAG’s argument. The following percentages of the respondents who had no opinion or a previous incorrect belief shifted to a correct belief: 58% for $N_{12}$, 36% for $N_2$, 83% for $N_7$, and 92.5% for $N_{11}$ (which was sourced from the Encyclopedia Britannica). These shifts represent 50%, 32%, 84% and 181% of a standard deviation unit respectively, indicating that NAG’s arguments were reasonably persuasive. In future, we shall undertake more rigorous testing in order to compare NAG’s arguments against human-generated arguments.

NAG was tested on five sample scenarios in order to assess the effect of using spreading activation to simulate attention.

\footnote{Argument flaws such as reasoning cycles and weak inferences are also detected by the Analyzer, and are corrected by the Strategist and the Generator if possible. However, discussion of the correction procedures is beyond the scope of this paper.}

\footnote{Technically, due to the high variation in the responses, only the largest of these shifts is statistically significant with $p = 0.035$ (when a normal distribution is assumed).}
Table 1: Test scenarios for NAG

<table>
<thead>
<tr>
<th>Name</th>
<th># nodes in SN</th>
<th># nodes in BN</th>
<th>average connect.</th>
<th>ave. time with SA</th>
<th>ave. time w/o SA</th>
</tr>
</thead>
<tbody>
<tr>
<td>asteroid</td>
<td>100</td>
<td>50</td>
<td>~3.25</td>
<td>12.5(4)</td>
<td>25(4)</td>
</tr>
<tr>
<td>finance</td>
<td>100</td>
<td>120</td>
<td>~4</td>
<td>32.8(5)</td>
<td>131.4(5)</td>
</tr>
<tr>
<td>alphabet</td>
<td>50</td>
<td>50</td>
<td>~4</td>
<td>8.5(4)</td>
<td>25.8(4)</td>
</tr>
<tr>
<td>phobos</td>
<td>20</td>
<td>40</td>
<td>~3</td>
<td>3.5(2)</td>
<td>6.0(2)</td>
</tr>
<tr>
<td>papers</td>
<td>20</td>
<td>20</td>
<td>~3</td>
<td>3.3(2)</td>
<td>6.5(2)</td>
</tr>
</tbody>
</table>

Table 1 shows the number of nodes and average connectivity in these scenarios, and the average time (in cpu seconds) required for generating arguments with and without spreading activation (columns 5 and 6 respectively; the number of runs performed appears in parenthesis). These results were obtained using mid-range (spreading activation) parameter values for a variety of goals (one goal per run). In all but one run the same arguments were generated with and without spreading activation. A slower decay and a lower activation threshold (§ Focusing the Argument) resulted in the incorporation of more nodes into the Argument Graph. In extreme cases this yielded longer argument generation times than without spreading activation due to the need to inspect nodes that were only marginally related to the goal. A quick decay and a high activation threshold resulted in the incorporation of fewer nodes into the Argument Graph. In extreme cases this also extended argument generation times, since the search for an argument became mainly goal based.

Related Research

The mechanism presented in this paper uses Bayesian reasoning to perform abduction during argument generation, and performs spreading activation to focus the argument. This use of spreading activation resembles Charniak and Goldman's (1993) use of a marker passing mechanism to focus attention in a Bayesian plan recognition system.

The approach of "interpretation as abduction" used in (Hobbs et al. 1993) aims to recover the premises and inferential links which lead to the conclusion of some given argument. This is similar to NAG's analysis-generation cycle. However, NAG is a system that reasons under uncertainty and can generate as well as analyze its own arguments. A generative system based on the work of Hobbs et al. is described in (Thomason, Hobbs, & Moore 1996). That system deals with what can be readily inferred, and so deleted, during communication, but the generated discourse does not present an argument in support of a proposition. Horacek (1997) and Mehl (1994) describe systems that turn an explicit argument into one where easily inferred information is left implicit. However, both of these systems require a complete argument as input, while NAG constructs its own arguments.

NAG's generation-analysis cycle resembles the propose-evaluate-modify cycle in (Chu-Carroll & Carberry 1995). However, NAG uses Bayesian reasoning to determine the impact of an argument on an addressee's beliefs, and it may combine several lines of reasoning to achieve its goal, rather than selecting a single proposition.

Conclusion

We have offered a mechanism for argument generation and analysis which uses a series of focusing-generation-analysis cycles to build two BNs (one in the normative model and another in the user model) that contain the information required to construct a nice argument. This use of two models enables us to distinguish between normatively correct and persuasive arguments. An attentional mechanism is used to focus the search during argument generation, and partial propagation, performed over the Bayesian subnetworks in focus (the current Argument Graph), is used to estimate the impact of the resultant argument on an addressee's beliefs. A preliminary evaluation of NAG's arguments yielded encouraging results, an evaluation of NAG's attentional mechanism shows that it substantially reduces argument generation times without appreciable effects on argument quality.

Acknowledgments

This research was supported in part by Australian Research Council grant A49531227.

References