# The Use of Belief Networks for Mixed-Initiative Dialog Modeling

Helen M. Meng, *Member, IEEE*, Carmen Wai, and Roberto Pieraccini, *Member, IEEE*

***Abstract*—This paper proposes the use of Belief Networks (BN) for mixed-initiative dialog modeling. The BN-based framework was previously used for natural language understanding, where BNs infer the informational goal of the user's query based on its parsed semantic concepts. We extended this framework with the technique of backward inference that can automatically detect missing or spurious concepts based on the inferred goal. This is, in turn, used to drive the mixed-initiative dialog model that prompts for missing concepts and clarifies for spurious concepts. Applicability is demonstrated for a simple foreign exchange domain, and our framework's mixed-initiative interactions were shown to be superior to the system-initiative and user-initiative interactions. We also investigate the scalability and portability of the BN-based framework to the more complex air travel (ATIS) domain. Backward inference detected an increased number of missing and spurious concepts, which led to redundancies in the dialog model. We experimented with several remedial measures that showed promise in reducing the redundancies. We also present a set of principles for hand-assigning "degrees of belief" to the BN to reduce the demand for massive training data when porting to a new domain. Experimentation with the ATIS data also gave promising results.**

***Index Terms*—Belief networks, dialog modeling, mixed-initiative.**

## I. Introduction

SPOKEN dialog systems demonstrate a high degree of usability in many restricted domains, and these range from air travel, train schedules, restaurant guides, ferry time tables, electronic automobile classifieds, weather, and e-mail [1]–[9]. The user typically interacts with these systems to retrieve information e.g., train and ferry schedules; or to complete a task, e.g., book a flight, reserve a restaurant table, find an apartment, etc. Dialog modeling in these systems plays an important role in assisting users to achieve their goals effectively. The system-initiative dialog model assumes complete control in providing step-wise guidance for the user through an interaction. At each step, the model elucidates what the user may say or input into the system. The system-initiative model does not permit deviations from the pre-set course of interaction, but such restrictions also help attain high task completion rates. Conversely, the user-initiative model offers maximum flexibility for the user to determine his preferred course of interaction. With this type of interaction, it may be difficult to elucidate the system's scope of competence to the user. Should the user's request fall outside of this scope, the system may fail to help the user fulfill his goal. To strike a balance between the system-initiative and user-initiative models, the mixed-initiative dialog model allows both the user and the system to influence the course of interaction. It is possible to *handcraft* a sophisticated mixed-initiative dialog flow, but the task is expensive, and may become intractable for complex application domains.

Recent research efforts in dialog modeling attempt to reduce manual handcrafting by adopting data-driven approaches. Dialog design is formulated as a stochastic optimization problem, where machine learning techniques are applied to learn the "optimal" dialog strategy from training data. For example, ergodic hidden markov models have been applied [10], and reinforcement learning based on a Markov Decision Process [11], [12] was used to learn the dialog process with states, actions and sequential decisions that is optimal from the perspective of a reward/cost function [13]. This function is dependent on factors such as user satisfaction, task completion rate, user effort, etc.

An alternative approach involves the use of Belief Networks (BN) for modeling dialog interactions. [14] proposed to use BN to model the mental states of dialog participants and changes in the mental states along with incoming evidence connected to utterances in a conversation. He argued that the BNs provide a probabilistic and decision-theoretic framework for modeling dialog, and the framework is plausible for computational implementation. [15] applied BN to infer the user's intention and attention in a mixed-initiative interaction, combined with maximization of expected user utility when the computer selects the "best" response or action. This framework has been demonstrated in the implementation of a virtual front desk receptionist and also an appointment scheduling agent: [16], [17].

This work is our initial attempt to utilize BN in mixed-initiative interactions. We focus on application domains where the user interacts with a spoken dialog system for database retrieval. There is a multitude of ways in which the user can express his domain-specific informational goal(s) in natural (spoken) language. Furthermore, the expression may span multiple utterances in the dialog interaction. BNs are used to infer the user's informational goal based on the semantics of a query expres-

Manuscript received October 22, 2001; revised September 19, 2002. This work was supported in part by a grant from the Research Grants Council of the Hong Kong Special Administrative Region, China under Projects CUHK4177/00E and CUHK4326/02E, and a Direct Grant from The Chinese University of Hong Kong. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Hermann Ney.

H. M. Meng is with the Department of Systems Engineering and Engineering Management, The Chinese University of Hong Kong, Shatin, Hong Kong SAR, China (e-mail: hmmeng@se.cuhk.edu.hk).

C. Wai was with the Department of Systems Engineering and Engineering Management, The Chinese University of Hong Kong, Shatin, Hong Kong SAR, China. She is now with the Technology Services Division, Hong Kong SAR Government, Hong Kong, China.

R. Pieraccini is with the Natural Dialog Group at Speech Works International Ltd., New York, NY 10004 USA.

Digital Object Identifier 10.1109/TSA.2003.814380

sion that is either self-contained or dependent on discourse context. In addition, effective database retrieval for a given informational goal may require a set of necessary attributes. If this set of attributes is fully specified in the user's query, the interaction should conclude with successful task completion. However, if there are attributes *missing* from the query expression, the mixed-initiative dialog model should automatically *prompt* the user for the attribute value. Alternatively, there may be *spurious* attributes in the query expression—these may be optional attributes specified by the user or attributes resulting from speech recognition errors. The mixed-initiative dialog model should automatically *clarify* for such spurious attributes with the user. Hence prompting and clarification are the dialog acts of focus in this work. We believe that the BNs offer several advantages to our problem.

i) The BN probabilities can be automatically trained from available data. Automation eases portability across domains and scalability to more complex domains. The user's informational goal can be identified by probabilistic inference in the BNs.
ii) The BN topology can also be automatically learned from training data. The topology can capture the inter-node dependencies in the BN, where each node represents a semantic concept characterizing the knowledge domain.
iii) Belief propagation within a BN corresponds to computing the probability of events that can be used for reasoning. This procedure enables automatic detection of missing and spurious concepts which can drive the mixed-initiative dialog model. The procedure is also suitable for belief revision as the discourse evolves in the course of the dialog interaction.
iv) The BN framework is amenable to the optional incorporation of human knowledge should training data be sparse. For example, the BN topology may be handcrafted, learned from training data, or both. Similarly, BN probabilities may be trained or assigned/refined by hand according to the developer's "degree of belief" in inter-node dependencies.

We have applied the BN framework to two domain-specific systems—the foreign exchange domain and the air travel domain. The systems will be described in the following.

## II. Overview of the Belief Network Approach

The proposed approach for dialog modeling extends the Belief Network (BN) framework previously used for Natural Language Understanding (NLU). Details of our NLU framework are described in [18], and the framework shares similar objectives with related work that strives to automate grammar development for natural language understanding [19]–[23]. This section presents an overview of our BN-based approach and its extensions. Specific applications of this approach to two application domains (foreign exchange and air travel) are presented in Sections III–V.

Understanding natural language queries for a specific application domain involves parsing the input query into a series of domain-specific *semantic concepts*, and from these we infer the

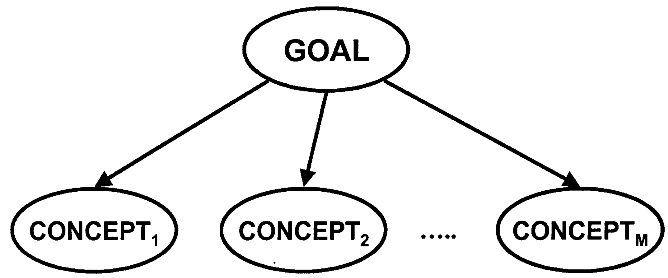


Fig. 1. Basic topology for our Belief networks. This topology assumes conditional independence among concepts. The arrows of the acyclic graph are drawn form cause to effect. This topology is equivalent to the Naïve Bayes formulation [24].

*informational goal* of the user's query. Semantic concepts correspond to the pieces of information that are relevant to the application. An information goal is the service or the information requested by the user. It is assumed that within a restricted application domain, there is a finite set of $(M)$ semantic concepts as well as a finite set of $(N)$ informational goals. The goals $G_i$ and concepts $C_j$ are all binary, and the concept $C_j$ is true if it appears in the utterance. Hence, we can formulate the NLU problem as making $N$ binary decisions with $N$ BNs, one for each informational goal. The BN for goal $G_i$ takes as input a set of semantic concepts $\underline{C}$ [1] extracted from the user's query. The BN gives the *a posteriori* probability $P(G|C)$ and from this the binary decision is made by thresholding. The topology of the BN may assume conditional independence among the concepts in $\underline{C}$ i.e., there are direct links between the goal and the concept nodes, but no linkages among the concepts nodes. This is equivalent to a naïve Bayes formulation, and is illustrated in Fig. 1.

Goal inference based on $P(G|\underline{C})$ may be computed as shown in (1) with the conditional independence assumption. As mentioned, this is equivalent to the Naïve Bayes formulation and (1) simply applies Bayes' rule. We assume that the goal $G_i$ is present if $P(G_i|\underline{C})$ is greater than a threshold $\theta$; and that the goal $G_i$ is absent otherwise. $\theta$ may be set to 0.5 for simplicity since $P(G_i=1|\underline{C})+P(G_i=0|\underline{C})$. This formulation provides us with a means of rejecting out-of-domain (OOD) queries—a query is classified as OOD when all BNs vote negative for their corresponding goals. In addition, the formulation also accommodates queries with multiple goals, i.e., when multiple BNs vote positive. We may also force the selection of a single goal (even when multiple BNs vote positive) by applying the maximum *a posteriori* rule. See (1) at the bottom of the next page.

In this work, we use an *enhanced* topology for the BN that adds linkages in between concept nodes to model the inter-concept dependencies for goal inference. We constrain ourselves to topologies that belong to the classification-based network structures. A classification-based network has a root node (with no parents) which represents our goal $G_i$. Fig. 2 shows an example of our enhanced topology. [25] presented a method for learning such linkages automatically from training data according to the Minimum Description Length (MDL) principle. A brief description is as follows: Every node in the BN contributes toward the complexity of the network by a magnitude $L_{\text{network}}$ (to which we also refer as the *network*

[1] $\underline{C}$ represent a concept.

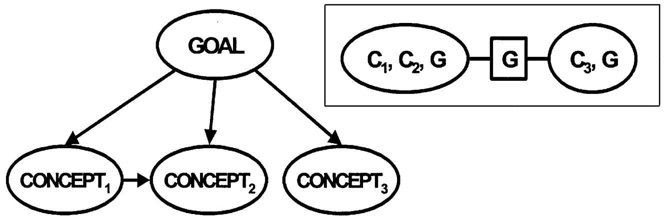


Fig. 2. Trained topology for our Belief networks. This topology captures the causal dependencies between the goal and a concept as well as between two concepts. The arrows of the acyclic graph are drawn from cause to effect. Dependencies among concepts are automatically learned from training data according to the minimum description length (MDL) principle. The inset shows the cliques of the network. The two cliques are $(C_1, C_2, G)$ and $(C_3, G)$. $G$ is the separator node between the two cliques.

*description length*). Lower values for $L_{\mathrm{network}}$ reflect lower network complexities. Each node also contributes toward the accuracy in modeling the data by a magnitude of $L_{\mathrm{data}}$ (to which we also refer as the *data description length*). Lower values for $L_{\mathrm{data}}$ reflect higher accuracy. Consider the BN for goal $G_j$, with concept nodes $\{C_{j1}, C_{j2} \ldots C_{jM}\}$. For a given concept node $C_{ji}$ in the BN, both $L_{\mathrm{network}}$ and $L_{\mathrm{data}}$ are functions of the concept node itself, its parents *Parents* $(C_{ji})$ and their instantiations in the training data. Hence the total description length $(L_{\mathrm{total}})$ contributed by a given node is defined as $L_{\mathrm{total}} = L_{\mathrm{network}} + L_{\mathrm{data}}$. The total description length of a network is the sum of all the concept nodes in the network. Similarly, the total description length of an arc in a network is the sum of the two nodes linked by the arc. As mentioned earlier, the MDL principle aims to find the simplest network that can model the training data most accurately. We used a best-first search that begins by computing the $L_{\mathrm{total}}$ for each possible arc that can be added to our pre-defined network structure. The arcs are sorted in increasing order of $L_{\mathrm{total}}$ to form the sorted arc list and each arc in the list is paired with the initial network topology $(T_0)$ to form a search list of network-arc pairs, i.e., $\{(T_0, \mathrm{arc}_1), (T_0, \mathrm{arc}_2), \ldots, (T_0, \mathrm{arc}_n)\}$. These pairs are sorted according to the sum of the total description lengths of the original network and the arc. The pair with minimum description length is popped off the list and the network and arc are combined to form a new topology $(T_1)$, an enhanced topology with minimum description, i.e., thus far $T_{\mathrm{MDL}} = T_1$. $T_{\mathrm{MDL}}$ is then paired with other possible arcs and the search process continues for a fixed number of iterations. Upon completion of this machine learning process, each goal has its own BN with an enhanced topology. We also refer to this as the "trained" topology. Fig. 2 provides an illustration.

For trained BN topologies similar to that shown in Fig. 2, probability propagation for goal inference is more complex than was shown in (1).[2] We provide a brief explanation here. Take the BN in Fig. 2 as an example, there are two *cliques* (i.e., maximal sets of nodes that are all pairwise linked)—$(G, C_1, C_2)$ and $(G, C_3)$. This is illustrated in the inset of the figure, which also shows that the cliques can communicate through the separator node $G$. Each clique relates to a joint probability $P(G, \underline{C})$. For example, in Fig. 2 the clique $(G, C_1, C_2)$ relates to the joint probability $P(G, C_1, C_2)$ and the clique $(G, C_3)$ relates to the joint probability $P(G, C_3)$. Given a user's query, we derive the presence and absence of the various concepts $\underline{C}$, and update the joint probability according to (2). The updated joint probability is eventually marginalized to produce a probability for goal identification $(P^*(G))$

$$P^*\left(G_i, \vec{C}\right) = P\left(G_i | \vec{C}\right) P^*\left(\vec{C}\right) = P\left(G_i, \vec{C}\right) \frac{P^*\left(\vec{C}\right)}{P\left(\vec{C}\right)} \tag{2}$$

where

| | |
|---|---|
| $P^*(C)$ | instantiated according to the presence or absence of the concepts in the user's query; |
| $P(Gi, C)$ | joint probability obtained from the training set; |
| $P^*(Gi, C)$ | updated joint probability. |
| $*$ | denotes an updated probability with knowledge about the presence/absence of the various concepts in the user's query. |

Fig. 3 illustrates the process of computing the updated probability $P^*(G)$ for goal identification, using the BN in Fig. 2 as an example. A more detailed description is included in the Appendix.

### A. Extension for Dialog Modeling—Backward Inference

We extend the BN framework from NLU to mixed-initiative dialog modeling. The main idea is to enable BNs to automatically detect missing or spurious concepts according to domain-specific constraints captured by their probabilities. Should a *missing* concept be detected, the BN will drive the dialog model to *prompt* the user for the necessary information. Should a *spurious* concept be detected the BN will drive the dialog model to *clarify* with the user regarding the unnecessary information. Automatic detection of missing and spurious concepts is achieved by the technique of *backward inference*.

[2]For an introduction to Bayesian Networks and probability propagation, we refer the reader to [26]

$$\begin{aligned} P(G_i = 1|\vec{C}) &= \frac{P(\vec{C}|G_i = 1)P(G_i = 1)}{P(\vec{C})} \\ &= \frac{\prod_{k=1}^{M} P(C_k|G_i = 1)P(G_i = 1)}{\prod_{k=1}^{M} P(C_k|G_i = 0)P(G_i = 0) + \prod_{k=1}^{M} P(C_k|G_i = 1)P(G_i = 1)} \end{aligned} \tag{1}$$

Assume we know $C_1=1$, $C_2=1$, $C_3=0$ from the user's query, we would like to find the probability of $G=1$ for the query as follows:

Update the joint probability in the first clique. Since $C_1$ is present,

$P^*(C_1=1)=1$

$$P^*(C_1=1,C_2=1,G=1)=P(C_1=1,C_2=1,G=1)\frac{P^*(C_1=1)}{P(C_1=1)}$$

↓

Since $C_1$ is known, we obtain $P^*(C_2=1, G=1)$ from $P^*(C_2=1, G=1, C_1)$

Marginalize $P^*(C_2=1, G)$ to obtain $P^*(C_2=1)$

↓

Update the joint probability in the first clique, since $C_2$ is present,

$P^{**}(C_2=1)$

$$P^{**}(C_2=1,G=1)=P^*(C_2=1,G=1)\frac{P^{**}(C_2=1)}{P^*(C_2=1)}$$

↓

Marginalize $P^{**}(G=1, C_2)$ to obtain $P^*(G=1)$

↓

Propagate $P^*(G=1)$ through the connecting node $G$ to the second clique

↓

Update the joint probability in the second clique with $P^*(G=1)$

$$P^*(C_3=0,G=1)=P(C_3=0,G=1)\frac{P^*(G=1)}{P(G=1)}$$

↓

Marginalize $P^*(C_3=0, G)$ to obtain $P^*(C_3=0)$

↓

Update the joint probability in the second clique with $P^{**}(C_3=0)$ because $C_3$ is absent

$$P^{**}(C_3=0,G=1)=P^*(C_3=0,G=1)\frac{P^{**}(C_3=0)}{P^*(C_3=0)}$$

↓

Marginalize $P^{**}(G=1, C_3)$ to obtain $P^{**}(G=1)$

Fig. 3. Flow chart illustrating probability propagation through the trained Belief network topology as Fig. 2, for inferring the information goal $G$ based on the input concepts $C_1$, $C_2$, and $C_3$.

Backward inference involves probability propagation within the BN. Having inferred the informational goal $(G_i)$ for a given user's query, the goal node of the corresponding BN is instantiated (to either 1 or 0) to test the network's confidence in each of the input concepts. If the BN topology assumes conditional independence among the concepts, the updated probability of the concepts will be simply $P(C_j|G)$. However, in our BN in which the concepts depend on each other, the updated goal probability $P^*(C_i)$ will propagate to update the joint probabilities of each clique $P^*(\underline{C}, G_i)$. Thereafter we may obtain each $P^*(C_j)$ by marginalization. This procedure is described by (3), and it is similar to the procedure described by (2) for updating concept probabilities[3]

$$P^*\left(\vec{C},G_i\right)=P\left(\vec{C}|G_i\right)P^*\left(G_i\right)=P\left(\vec{C},G_i\right)\frac{P^*\left(G_i\right)}{P\left(G_i\right)} \quad (3)$$

[3]For a detailed illustration of this computation, please refer to the Appendix.

where

- $P^*(G_i)$ is updated from instantiating the goal node;
- $P(\underline{C}, G_i)$ joint probability of the clique obtained from the training set;
- $P^*(\underline{C}, G_i)$ updated joint probability of the clique.

Based on the value of $P^*(C_j)$, we make a binary decision (by thresholding) regarding whether $C_j$ should be present or absent. This decision is compared with the actual occurrence of $C_j$ in the user's query. If the binary decision indicates that $C_j$ should be absent but it is actually present in the input query, the concept is labeled spurious and the dialog model will invoke a clarification act. If the binary decision indicates that $C_j$ should be present but it is absent from the query, the concept is labeled missing and the dialog model will invoke a prompting act. In Section III, we demonstrate the applicability of this BN-based dialog model to a spoken language system in the foreign exchange domain.

## III. APPLICABILITY TO THE CU FOREX SYSTEM

We have chosen to investigate the feasibility the BN-based dialog model within the context of the CU FOREX system [27]. This is a trilingual (English, Putonghua and Cantonese) conversational hotline that supports inquiries about foreign exchange information. It supports inquiries regarding the exchange rates between a currency pair, as well as the interest rates for various deposit duration for a currency. The domain is relatively simple, and is characterized by two query types (or informational goals—Exchange Rates and Interest Rates); and five domain-specific concepts (two CURRENCYA concepts, TIME DURATION, EXCHANGE RATE and INTEREST RATE). For database retrieval, there are two domain-specific constraints:

i) an exchange rate inquiry requires that the currencies to be bought and sold be specified;
ii) an interest rate inquiry requires that a currency and a time duration be specified.

### A. Two Interaction Modalities

CU FOREX has been made available for experimentation for the public since August 1999. We receive a few hundred calls per month on average. The system currently supports two interaction modalities based on SpeechWorks 4.0:[4]

i) The Directed Dialog (DD) is designed for novice users. This type of interaction guides the user through a session, and elucidates what may be said at various stages of the interaction. Table I shows a DD interaction between the system and the user. Notice that at every dialog turn, the system guides the user to provide a specific information attribute.
ii) The Natural Language Shortcut (NLS) is designed for expert users who want to expedite the inquiry session by uttering a full query, and traverse the entire session within one single dialog turn. Hence the query may carry multiple attributes for database retrieval, as opposed to a single attribute per utterance in the DD case. An example dialog for an NLS interaction is shown in Table II.

### B. Belief Networks in CU FOREX

We developed two BNs, one for each informational goal. Each BN receives all of the five domain-specific concepts as input. We have also used the trained topology automatically learned according to the MDL principle [25]. The training data used here consists of 523 transcribed utterances collected from the NLS hotline, equally distributed between the exchange rate and interest rate inquiries. The resulting topology is illustrated in Fig. 4. The dotted arrow shows the causal dependency between the concepts CURRENCY_1 and CURRENCY_2 learned from data. This network contains the cliques (GOAL, CURRENCY_1, CURRENCY_2), (GOAL, DURATION), (GOAL, EX_RATE) and (GOAL, INT_RATE).

Goal inference proceeds as described in Section II. The decisions across the two BNs are combined to identify the output goal of the input query. Typical values of *a posteriori* probabilities obtained from goal inference are shown in Table III. These values are compared with the threshold $\theta = 0.5$ for making binary decisions.

The single asterisk in $P^*(G)$ indicates that the probability of goal $G$ has been updated during probability propagation in the

[4]http://www.speechworks.com

TABLE I
DIRECTED DIALOG (DD) INTERACTION BETWEEN THE CU FQREX SYSTEM AND THE USER REGARDING AN INTEREST RATE INQUIRY. NOTICE THAT AT EVERY DIALOG TURN, THE SYSTEM GUIDES THE USER TO PROVIDE A SPECIFIED INFORMATION ATTRIBUTE

| | | |
|---|---|---|
| **System** | : | Welcome to CU FOREX. Which language would you prefer, Cantonese, Putonghua or English? |
| **User** | : | *English* |
| **System** | : | Would you like exchange rates or interest rates? |
| **User** | : | *Interest rates* |
| **System** | : | Which currency are you interested in? |
| **User** | : | *Hong Kong dollar* |
| **System** | : | Would you like deposit durations for twenty-four hours, one month.... or one year? |
| **User** | : | *One month* |
| **System** | : | The quotes we provide are for reference only. The interest rate of the Hong Kong dollar for one month is XXX. |
| **System** | : | Welcome to CU FOREX. Which language would you prefer, Cantonese, Putonghua or English? |
| **User** | : | *English* |
| **System** | : | Would you like exchange rates or interest rates? |
| **User** | : | *Interest rates* |
| **System** | : | Which currency are you interested in? |
| **User** | : | *Hong Kong dollar* |
| **System** | : | Would you like deposit durations for twenty-four hours, one month.... or one year? |

TABLE II
NATURAL LANGUAGE (NLS) INTERACTION BETWEEN THE CU FQREX SYSTEM AND THE USER REGARDING AN EXCHANGE RATE INQUIRY. NOTICE THAT THE USER IS ABLE TO SPECIFY SEVERAL INFORMATION ATTRIBUTES WITHIN A SINGLE DIALOG TURN

| | | |
|---|---|---|
| **System** | : | Welcome to CU FOREX. Which language would you prefer, Cantonese Putonghua or English? |
| **User** | : | *English* |
| **System** | : | What kind of currency information are you interested in? |
| **User** | : | *I'd like to know the exchange rate between the greenback and the HK dollar please.* |
| **System** | : | The quotes we provide are for reference only. Exchange rate. US Dollar to HK Dollar. The buying rate is XXX. The selling rate is XXX. |
| **System** | : | Welcome to CU FOREX. Which language would you prefer, Cantonese Putonghua or English? |
| **User** | : | *English* |
| **System** | : | What kind of currency information are you interested in? |

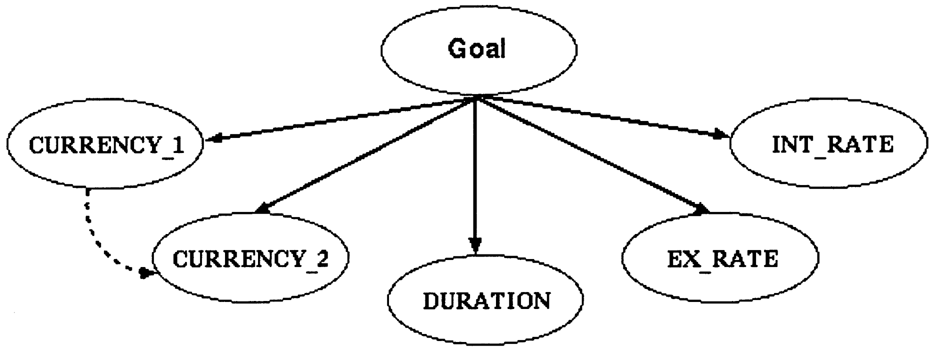


Fig. 4. Trained topology of our BNs in the CU FOREX domain. The EXCHANGE_RATE and INTEREST_RATE BNs have the same topology.

TABLE III
TYPICAL VALUES OF UPDATED PROBABILITIES OBTAINED FROM GOAL INFERENCE USING BNS IN THE CU FOREX DOMAIN

**User: Can I have the exchange rate of the yen please?**
- Inference by the BN for Exchange Rates:
  *P*(Goal* =Exchange Rates*)* = 0.823 ➔ goal *present*
- Inference by the BN for Interest Rates:
  *P*(Goal* =Interest Rates*)* = 0.256 ➔ goal *absent*

Hence, the inferred goal is Exchange Rates.

**User: Tell me about stock quotes**
- Inference by the BN for Exchange Rates:
  *P*(Goal* =Exchange Rates*)* = 0.14 ➔ goal absent
- Inference by the BN for Interest Rates:
  *P*(Goal* =Interest Rates*)* = 0.13 ➔ goal absent

Hence, the user's query is considered out-of-domain (OOD).

TABLE IV
INPUT QUERY IS *"CAN I HAVE THE INTEREST RATES OF THE YEN FOR ONE MONTH PLEASE?"* GOAL INFERENCE IDENTIFIES THAT THE UNDERLYING GOAL IS `Interest Rates`. THE GOAL NODE OF THE CORRESPONDING BN IS INSTANTIATED, AND BACKWARD INFERENCE PRODUCES $P^*(C_j)$ FOR EACH CONCEPT. THESE PROBABILITIES ARE COMPARED WITH THE THRESHOLD $\theta = 0.5$ TO MAKE A BINARY DECISION REGARDING THE PRESENCE/ABSENCE FOR EACH CONCEPT. THE BINARY DECISIONS AGREE WITH THE ACTUAL OCCURRENCES OF THE CONCEPTS

| *Concept $C_j$* | *$P^*(C_j)$* | **Binary Decision for $C_j$** | **Actual Occurrence for $C_j$** |
|---|---|---|---|
| CURRENCY_1 | 0.910 | Present | present |
| CURRENCY_2 | 0.006 | absent | absent |
| DURATION | 0.770 | present | present |
| EX_RATE | 0.011 | absent | absent |
| INT_RATE | 0.867 | present | present |

BN. In reality, the $P^*(G)$ for this example has been updated four times actually (once for each clique). A brief example illustrating this process is included in the Appendix, in which the number of asterisks indicate the number of updates. However, for the sake of simplicity in the main body of this paper, we use a single asterisk to indicate that a probability has been updated. Having instantiated the inferred goal, backward inference verifies the validity of the input query against domain-specific constraints. In this way we can test for cases of missing or spurious concepts,[5] and generate an appropriate system response. Consider the example of an interest rate inquiry, *"Can I have the interest rates of the yen for one month please?"* We instantiated the goal node of the BN (for Interest Rates) to 1, and performed backward inference for each input concept $C_j$ to obtain $P^*(C_j)$. This probability is compared with the threshold $\theta = 0.5$ to determine whether the concept should be present or absent (see equation at the bottom of this page).

The probabilities and binary decisions obtained in this example are shown in Table IV. The binary decision for each concept *agrees* with their actual occurrence in the input query. Thus

[5]These may be due to speech recognition errors in an integrated spoken dialog system.

| | |
|---|---|
| GOAL: | Interest_Rate |
| CURRENCY: | yen |
| DURATION: | one month |

Fig. 5. Illustration of the semantic frame corresponding to the query *"Can I have the interest rates of the yen for one month please?"*

we can use the concept-value pairs to form a semantic frame (see Fig. 5), which can be further processed for database retrieval.

However, if the binary decision for a concept disagrees with its actual occurrence, a prompt will be invoked to request the missing concept or to clarify spurious concepts. We illustrate these two cases by the following examples.

Case 1) **Prompt for a Missing Concept**

Consider the interest rate query *"Can I have the interest rate of the yen?"*. Backward inference gives $P^*(C_j)$ for the concept ⟨DURATION⟩ which equals 0.770. This is greater than the threshold $\theta = 0.5$, which suggests that the ⟨DURATION⟩ should be

$$P^*(C_i) \begin{cases} \geq \theta \rightarrow & C_j \text{ should be present in the given query with goal } G_i \\ < \theta \rightarrow & C_j \text{ should be absent in the given query with goal } G_i \end{cases}$$

TABLE V

INPUT QUERY IS "*CAN I HAVE THE INTEREST RATE OF THE LIRA AGAINST THE YEN*." GOAL INFERENCE IDENTIFIES THAT THE UNDERLYING GOAL IS `Exchange Rates`. THE GOAL NODE OF THE CORRESPONDING BN IS INSTANTIATED, AND BACKWARD INFERENCE PRODUCES $P^*(C_j)$ FOR EACH CONCEPT. THESE PROBABILITIES ARE COMPARED WITH THE THRESHOLD $\theta = 0.5$ TO MAKE A BINARY DECISION REGARDING THE PRESENCE/ABSENCE FOR EACH CONCEPT. THE BINARY DECISION OF THE CONCEPT ⟨INT_RATE⟩ FROM BACKWARD INFERENCE (BN OF INTEREST RATES) DOES NOT AGREE WITH ITS ACTUAL OCCURRENCE (SHADED). HENCE ⟨INT_RATE⟩ IS DEEMED SPURIOUS AND THE DIALOG MODEL ISSUES A CLARIFICATION RESPONSE

| *Concept* $C_j$ | $P^*(C_j)$ | Binary Decision for $C_j$ | Actual Occurrence for $C_j$ |
|---|---|---|---|
| CURRENCY_1 | 0.910 | Present | present |
| CURRENCY_2 | 0.920 | Present | present |
| DURATION | 0.017 | Absent | absent |
| EX_RATE | 0.840 | Present | absent |
| **INT_RATE** | **0.023** | **Absent** | **present** |

present but is missing. Hence the system prompts the user with "Please specify the deposit duration"

Case 2) **Clarify for a Spurious Concept**

Consider the query "*Can I have the interest rate of the lira against the yen*." The inferred goal is `Exchang Rates` and results from backward inference are shown in Table V. Comparison between the binary decisions for each concept with its actual occurrence in the query results in the automatic detection that the concept ⟨INT_RATE⟩ is spurious. This invokes the dialog model to generate the clarification response: *"Are you referring to the exchange rate between the lira and the yen?"*

### C. Evaluation of the BN-Based Dialog Model

Evaluation of the BN-based dialog model is based on 550 dialog sessions collected using the CU FOREX system during the period between November and December 1999. Recorded dialogs come from two hotlines—one with a directed dialog (DD) mode of interaction and the other with natural language shortcuts (NLS). Approximately 17% were rejected manually as the users were clearly attempting to break the system. Of the remaining queries, 285 calls were obtained from the Directed Dialog hotline while 170 calls were obtained from the NLS hotline. The task completion rates of the DD and NLS models are shown in Table VI. Our original DD model only accepts one attribute for each dialog turn, hence failures in the DD model are mainly caused by queries with multiple information attributes or OOD input. Failures in the NLS model are due to missing concepts, spurious concepts or OOD queries.[6] Detail statistics are tabulated in Table VII. We input these failed queries into our BN-based mixed-initiative dialog model to simulate online processing by this new model, and observed that the BN-based dialog model can automatically reject OOD input, and offered a continuation option at the failure points of *all* the dialogs.

[6]Note that we could have handcrafted some heuristics in a dialog manager to catch these errors in the original system but instead we have chosen to use the BN to work out the answer.

TABLE VI

TASK COMPLETION RATES OF THE ORIGINAL CU FOREX SYSTEM WITH DIRECTED DIALOG (DD) AND NATURAL LANGUAGE SHORTCUTS (NLS) INTERACTIONS, IN COMPARISON WITH TASK COMPLETION RATES FROM THE BN-BASED. THE COMPARISON IS BASED ON 550 DIALOG SESSIONS (1 SESSION PER PHONE CALL)

| | **Task completion rates of the original CU FOREX system** | **Task completion rates of the new BN-based dialog model** |
|---|---|---|
| **Directed Dialog Interactions** (total: 285 phone calls) | 85% | 96% (remaining 4% is OOD rejection) |
| **Natural Language Shortcuts** (total: 170 phone calls) | 63% | 97% (remaining 3% is OOD rejection) |

TABLE VII

CAUSES OF FAILURE FOR THE ORIGINAL CU FOREX DIALOG MODEL. PERCENTAGES REFER TO THE PROPORTION OF THE EVALUATED QUERIES (285 DD AND 170 NLS)

| | **Causes of failure for the original CU FOREX system** | | | |
|---|---|---|---|---|
| | Multiple Attributes | Missing Concepts | Spurious Concepts | Out-of-domain Input |
| **Directed Dialog** | 11% | --- | --- | 4% |
| **Natural Language Shortcuts** | --- | 30% | 4% | 3% |

Table VIII shows typical examples of failure in the DD and NLS models, and how these are handled by the BN-based dialog model.

## IV. SCALABILITY TO THE ATIS DOMAIN

Thus far we have demonstrated the applicability of the BN-based dialog model to the simple domain of foreign exchange in the CU FOREX system. This section reports on our investigation on the application of our BN-based dialog model to a more complex domain. We have chosen the ATIS (Air Travel Information Service) [28] domain due to data availability. ATIS is a common task in the DARPA Defense Advanced Research Projects Agency) Speech and Language Program in the US. We used Class A (context-independent) as well as Class D (context-dependent) queries of the ATIS-3 corpus. The disjoint training and test sets consist of 2820 and 773 (1993 test) transcribed utterances respectively. Each utterance is accompanied by its corresponding SQL query for database retrieval.

We treat the main attribute label of the SQL query as the informational goal of the input utterance. Inspection of the Class A training data reveals that out of the 32 query types (or informational goals, e.g., flight identification, fare identification, etc.), only 11 types have ten or more occurrences.

These 11 goals cover over 95% of the training set, and 94.7% of the testing set (1993 test). Consequently, we have developed 11 BNs to capture the domain-specific constraints for each informational goal. Also, with the reference to the attribute labels identified as key semantic concepts from the SQL query, we have designed our semantic tags for labeling the input utterance. We have a total of 60 hand-designed semantic tags, which include the concepts/attributes needed for database access (e.g., ⟨DAY_NAME⟩, ⟨FLIGHT_NUMBER⟩), and others that play a syntactic role for NLU (e.g., ⟨PREPOSITION⟩, ⟨SUPERLATIVE⟩). Hence, ATIS presents greater domain complexity characterized by 11 query types and total 60 domain-specific concepts.

The current investigation on scalability focuses on the ability of the trained BN's to correctly identify the goals and concepts in an input query, including those inherited from discourse history. We also focus on the ability of the BNs, using the same set of trained probabilities, to detect missing and spurious concepts based on domain-specific constraints. Such detection is critical for the mixed-initiative dialog interactions that lie within the scope of our study. However, since the ATIS corpora provides reference semantic frames but not mixed-initiative interactions, we can only evaluate the BNs based on goal/concept identification.

### A. Belief Networks in ATIS

As mentioned earlier, we developed 11 BNs for ATIS. In order to constrain computation time for goal inference, each BN only has $N(=20)$ concept nodes that are selected automatically for its goal $G_j$ using the Information Gain criterion. The selection aims to optimize on overall goal identification accuracy with reference to the Class A training utterances [18]. The BNs also adopt trained topologies that model inter-concept dependencies according to the MDL principle. Inclusion of such dependencies has brought further improvements in goal identification performance [25]. Fig. 6 shows the BN in the ATIS domain for the goal `Aircraft_Code`.

Goal inference proceeds as described in Section II for the trained BN topology. Thresholding $P^*(G_i)$ with $\theta = 0.5$ decides between the presence or absence of the goal $G_i$. In these experiments, we always apply the maximum *a posteriori* rule to identify a single goal for an input utterance. Utterances are labeled OOD when all BNs vote negative.

### B. Detecting Missing and Spurious Concepts in ATIS

As described previously, instantiating the inferred goal node followed by backward inference produces $P^*(C_j)$. Comparing this probability with the threshold ($\theta = 0.5$) decides whether the concept $C_j$ should be present or absent according to domain-specific constraints. The binary decision may be compared with the actual occurrence of the concept $C_j$ in the input query for detecting missing or spurious concepts. These are, in turn, used to drive the dialog model. However, as we migrated from CU FOREX to ATIS, we discovered that this methodology often produces *several* missing or spurious concepts for an input query. For example, consider the query:

**Query**: *what type of aircraft is used in american airlines flight number seventeen twenty three?*

**Concepts**: ⟨WHAT⟩ ⟨TYPE⟩ ⟨AIRCRAFT⟩ ⟨AIRLINE_NAME⟩ ⟨FLIGHT_NUMBER⟩

**Goal**: `Aircraft_Code`

Backward inference in the BN for `Aircraft_Code` produces updated probabilities $P^*(C_j)$ as shown in Table IX.[7] Our detection algorithm labels the concepts ⟨CITY_ORIGIN⟩ and ⟨CITY_DESTINATION⟩ to be missing, and ⟨FLIGHT_NUMBER⟩ to be spurious. One reason is because in the training corpus, most queries with the goal `Aircraft_Code` included the city pair instead of the flight number, e.g., "*what is the smallest aircraft available flying from pittsburgh to baltimore arriving on may seventh*." However, the constraints provided by an AIRLINE_NAME and a FLIGHT_NUMBER should serve equally well for database access to retrieve an `Aircraft_Code`.

If our dialog model follows through with the prompting for the missing concepts and clarifying for the spurious concepts in Table IX, the system will first prompt the user for the CITY_ORIGIN, then CITY_DESTINATION; and then clarify for FLIGHT_NUMBER. Such a dialog model has too much redundancy, and fails to realize that the attribute pair (AIRLINE_NAME, FLIGHT_NUMBER) provides an equivalent amount of constraints as the attribute pair (CITY_ORIGIN, CITY_DESTINATION) for retrieving an Aircraft_Code. We attempted to solve this problem in two ways:

TABLE VIII
EXAMPLE DIALOGS ILLUSTRATING THE CAUSES OF FAILURE IN THE ORIGINAL CU FOREX SYSTEMS WITH DD AND NLU INTERACTIONS, AND HOW THESE CAN BE HANDLED APPROPRIATELY BY THE NEW BN-BASED DIALOG MODEL

| **Dialog 1: Directed Dialog (DD) Interaction** | | |
|---|---|---|
| **System** | : | Would you like exchange rates or interest rates? |
| **User** | : | *Exchange rates* |
| **System** | : | Which currency you would like to sell? |
| **User** | : | *Yen to Hong Kong please.* |
| **System (Original CU FOREX)** | : | Failed (Reason: Multiple information attributes in a DD interaction) |
| **System (New BN-based Dialog Model)** | : | *The exchange rate between the Yen and the Hong Kong dollar is XXX.* (Concepts pass the domain constraints) |
| **Dialog 2: Natural Language Shortcut (NLS) Interaction** | | |
| **System** | : | What kind of currency information are you interested in? |
| **User** | : | *Tell me about interest rates.* |
| **System (Original CU FOREX)** | : | Failed (Reason: Missing the concepts <CURRENCY_1> and <DURATION>) |
| **System (New BN-Based Dialog System)** | : | *Please tell me the currency you are interested in* (Prompt for the missing concepts.) |

[7]When there are no inter-concept linkages, these updated probabilities $P^*(C_j)$ can be computed once and stored. The situation changes later when $P^*(C_j)$ also depends on the instantiation of its parent concepts.

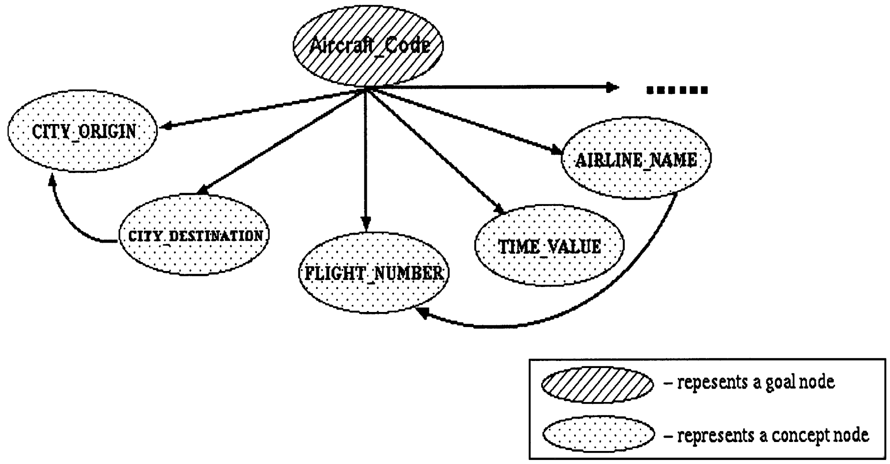


Fig. 6. Example of a BN in the ATIS domain. The goal is Aircraft_Code and the goal node has been labeled. The others are concept nodes. The trained topology is used to model concept-concept dependencies (see the arrows connecting CITY_DESTINATION with CITY_ORIGIN, and AIRLINE_NAME with FLIGHT_NUMBER).

TABLE IX
THE INPUT QUERY IS "*WHAT TYPE OF AIRCRAFT IS USED IN AMERICAN AIRLINES FLIGHT NUMBER SEVENTEEN TWENTY THREE*". UPDATED CONCEPT PROBABILITIES $P^*(C_j)$ OBTAINED FROM BACKWARD INFERENCE IN THE BN FOR `Aircraft_Code` ARE SHOWN. USING THRESHOLD $\theta = 0.5$ WE CAN DECIDE WHETHER THE CONCEPT $C_j$ SHOULD BE PRESENT OR ABSENT. THIS BINARY DECISION IS COMPARED AGAINST THE ACTUAL OCCURRENCE OF THE CONCEPT $C_j$ IN THE INPUT QUERY, IN ORDER TO DETECT MISSING AND SPURIOUS CONCEPTS (SHADED)

| ***Concept $C_j$ (subset)*** | ***$P^*(C_j)$*** | **Binary Decision for $C_j$** | **Actual Occurrence for $C_j$** |
|---|---|---|---|
| AIRCRAFT | 1.000 | present | present |
| AIRLINE_NAME | 0.538 | present | present |
| **CITY_ORIGIN** | **0.645** | **present** | **absent** |
| **CITY_DESTINATION** | **0.615** | **present** | **absent** |
| TIME_VALUE | 0.201 | absent | absent |
| **FLIGHT_NUMBER** | **0.420** | **absent** | **present** |

*1) Double Thresholding:* In order to prevent dialog redundancies in prompting for missing concepts or clarifying for spurious concepts, we defined two thresholds for backward inferencing; see equation at the bottom of this page.

Concepts whose probabilities (from backward inference) score between $\theta_{\text{upper}}$ and $\theta_{\text{lower}}$ will not take effect in mixed-initiative response generation (i.e., prompting/clarification). Concepts whose scores exceed $\theta_{\text{upper}}$, and also correspond to an SQL attribute will be prompted if missing; and concepts whose scores is lower than $\theta_{\text{lower}}$, and correspond to an SQL attribute will be clarified if spurious. Based on the training data, we have empirically adopted 0.7 and 0.2 for $\theta_{\text{upper}}$ and $\theta_{\text{lower}}$ respectively. The double threshold scheme enables the dialog model to prompt for missing concepts that are truly needed, and clarify for spurious concepts that may confuse the query's interpretation.

*2) Handcrafted Topology for BNs:* BN topologies automatically learned from training data may not be able to capture some equivalence classes for dialog modeling. This is partially because the actual constraints depend much more on the actual "logic" of the application domain rather than the corpus of training sentences. In capturing the equivalence classes from training data, we assume that the users are aware of the related domain-specific constraints that will be reflected in their queries. In principle, the logic of the domain is also obtainable from the database schema or from some representation of the "business logic". Referring to the example in Section IV-B, the equivalence classes for the Aircraft_Code BN may be expressed as

((CITY_ORIGIN AND CITY_DESTINATION OR
FLIGHT_NUMBER AND AIRLINE_NAME))

In other words, providing either the origin and destination cities, or providing the airline name with the flight number are sufficient constraints for locating the aircraft code from the ATIS database. The occurrence of the second attribute pair (FLIGHT_NUMBER AND AIRLINE_NAME) is relatively sparse in the training corpus, and the original BN topology (Fig. 6) is unable to capture the logic between these two attribute pairs.

We attempt to refine the BN topologies by adding linkage(s) that are obvious but have not been learned due to sparseness in training data. For example, the refined topology for the Aircraft_Code BN is shown in Fig. 7. Hence we have added a link between the two attribute pairs (see dotted arrow).

The BNs with the handcrafted topology are trained with the training corpus. We investigate how the previous example

$$P^*(C_j) \begin{cases} \geq \theta_{\text{upper}} \rightarrow & C_j \text{ should be } present \text{ in the given } G_i \text{ query} \\ < \theta_{\text{upper}} \text{ and } \geq \theta_{\text{lower}} \rightarrow & C_j \text{ is } optional \text{ in the given } G_i \text{ query} \\ < \theta_{\text{upper}} \rightarrow & C_j \text{ should be } present \text{ in the given } G_i \text{ query} \end{cases}$$

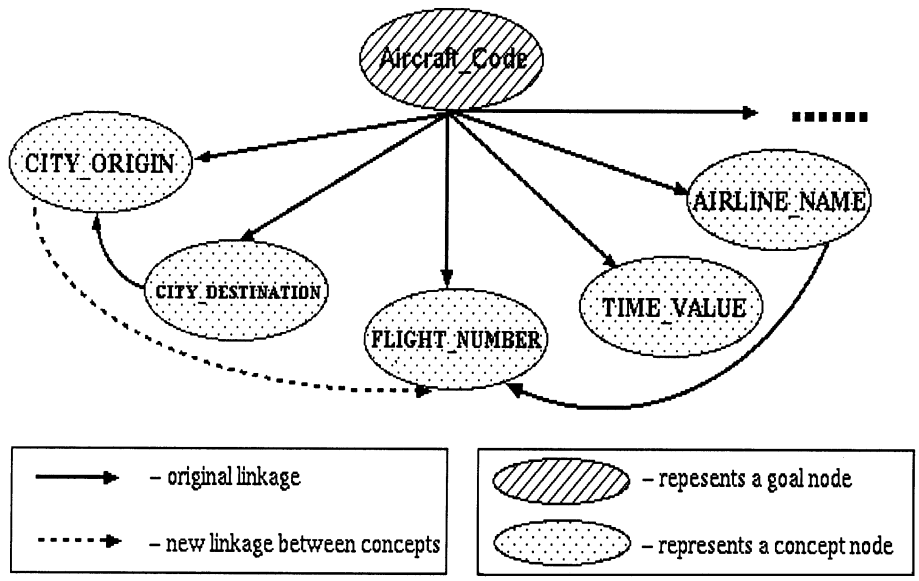


Fig. 7. Refined topology for the `Aircraft_Code` BN. A link is inserted (dotted arrow) to capture the equivalence between the attribute pairs (CITY_ORIGIN, CITY_DESTINATION) and (flight_number, airline_name). They provide the same degree of constraints for the inference of the goal `Aircraft_Code`.

TABLE X
REVISED PROCEDURE OF BACKWARD INFERENCE FOR THE BN TOPOLOGY WITH CONCEPTS DEPENDENCE. FOR THE CONCEPT NODE $C_j$, WE INSTANTIATE ITS PARENT NODE(S) ONLY WHILE $C_j$ IS BEING EVALUATED

| |
|---|
| For the BN of inferred goal $G_i$ with concepts $C_j$: |
| 1. Instantiate the goal node $G_i$ to 1 |
| 2. For each concept $C_j$ |
| Identify its parent(s) $C_k$ (i.e. there is a link going from $C_k$ to $C_j$) |
| Instantiate the node for $C_k$ node according to its occurrence in the user's query |
| 3. Obtain the updated probability $P^*(C_j)$ |

"*what type of aircraft is used in american airline flight number seventeen twenty three*" is handled with the hand-refined BN topology. Previously, backward inference involves only the instantiation of the inferred goal for the sake of simplicity. Here since our BN topology contains inter-concept linkages, backward inference regarding a particular concept involves not only the instantiation of the inferred goal, but also the instantiation of the occurrence of parent concepts. This is shown in Table X.

We replaced the double threshold in Section IV-B-I with the single threshold $\theta = 0.5$ for $P^*(C_j)$, in order to detect missing and spurious concepts. Results are tabulated in Table XI. Comparison with Table IX shows the effect of using the hand-refined BN topology to our previous example sentence. The handcrafted topology has captured the equivalence classes (AIRLINE_NAME, FLIGHT_NUMBER) and (CITY_ORIGIN, CITY_DESTINATION) and avoided redundancies in the two-threshold dialog model. Using the single threshold may be more desirable than using two thresholds that are set in an *ad hoc* way. It should be noted that for the hand-refined BN, linkage insertion requires some human knowledge and inspection. Thereafter, the procedure of training to acquire network probabilities remains identical to the previous BN-based model that does not involve handcrafting. Table XII shows another query of Aircraft_Code that shows similar result.

TABLE XI
INPUT QUERY IS "*WHAT TYPE OF AIRCRAFT IS USED IN AMERICAN AIRLINES FLIGHT NUMBER SEVENTEEN TWENTY THREE*". THE BN TOPOLOGY FOR THE INFERRED GOAL `Aircraft_Code` HAS BEEN HAND-REFINED (SEE FIG. 7). UPDATED CONCEPT PROBABILITIES $P^*(C_j)$ OBTAINED FROM BACKWARD INFERENCE ARE SHOWN. A SINGLE THRESHOLD $\theta = 0.5$ IS USED TO DECIDE WHETHER THE CONCEPT $C_j$ SHOULD BE PRESENT OR ABSENT. THIS BINARY DECISION IS COMPARED AGAINST THE ACTUAL OCCURRENCE OF THE CONCEPT $C_j$ IN THE INPUT QUERY, IN ORDER TO DETECT MISSING AND SPURIOUS CONCEPTS (SHADED). COMPARISON WITH TABLE IX SHOWS THAT THE HAND-REFINED TOPOLOGY HAS AVOIDED REDUNDANCIES IN TWO-THRESHOLD BN DIALOG MODEL)

| ***Concept $C_j$ (subset)*** | ***$P^*(C_j)$*** | **Binary Decision for $C_j$** | **Actual Occurrence for $C_j$** |
|---|---|---|---|
| AIRCRAFT | 1.000 | present | present |
| AIRLINE_NAME | 0.538 | present | present |
| **CITY_ORIGIN** | **0.000** | **absent** | **absent** |
| **CITY_DESTINATION** | **0.000** | **absent** | **absent** |
| TIME_VALUE | 0.201 | absent | absent |
| **FLIGHT_NUMBER** | **0.600** | **present** | **present** |

*3) Optional Concepts:* The example in Table XII classified the concept ⟨TIME_VALUE⟩ as spurious. In a complex domain with many attributes/concepts (e.g., 60 for ATIS), the user may freely provide additional specifications for database access. For these additional concepts, backward inference usually produces low probabilities, but our dialog model may not need to clarify for all of them. Hence we have defined a set of "optional concepts" by the following heuristic: For a given goal $G_i$, if the occurrence frequency of the concept in the training corpus is lower than the threshold $\theta(G_i)$, the concept will be classified as optional. $\theta(G_i)$ is defined as

$$\theta(G_i) = \frac{\#\text{training sentences with goal } G_i}{2}. \qquad (4)$$

Optional concepts will not be considered spurious during backward inference, and the BN-based dialog model will not generate a clarification response for the concepts.

TABLE XII

INPUT QUERY IS "*DISPLAY TYPES OF AIRCRAFT DEPARTING FROM CLEVELAND TO DALLAS BEFORE NOON*". THIS TABLE SHOWS THE RESULTS OF BACKWARD INFERENCE, ANALOGOUS TO TABLE X. WE OBSERVE THAT WITH THE HAND-REFINED TOPOLOGY AS SHOWN IN FIG. 7, THE OCCURRENCE OF THE ATTRIBUTE PAIR (CITY_ORIGIN, CITY_DESTINATION) AUTOMATICALLY LOWERS THE PROBABILITIES $P^*(C_j)$ FOR THE CONCEPTS ⟨AIRLINE_NAME⟩ AND ⟨FLIGHT_NUMBER⟩

| *Concept $C_j$ (subset)* | $P^*(C_j)$ | Binary Decision for $C_j$ | Actual Occurrence for $C_j$ |
|---|---|---|---|
| aircraft | 1.000 | present | present |
| airline_name | 0.000 | absent | absent |
| **city_origin** | **1.000** | **present** | **present** |
| **city_destination** | **1.000** | **present** | **present** |
| **time_value** | **0.201** | **absent** | **present** |
| flight_number | 0.000 | absent | absent |

TABLE XIII

THIS QUERY EXAMPLE ILLUSTRATES OUR INITIAL SCHEME FOR CONTEXT INHERITANCE, IN WHICH THE CURRENT QUERY INHERITS ALL THE CONCEPTS FROM THE PREVIOUS QUERY IN THE SAME DIALOG SESSION. THIS SCHEME IS FOUND TO BE OVERLY AGGRESSIVE, AS THE EXTRA CONCEPTS INHERITED AFFECTED GOAL IDENTIFICATION PERFORMANCE

| | | |
|---|---|---|
| **System** | : | what kind of flight information are you interested in? |
| **User** | : | *find me a flight from cincinnati to westchester county arriving next saturday before six p m* (Class A) |
| **Concepts** | : | <FLIGHT> <FROM> <CITY_ORIGIN> <TO> <CITY_DESTINATION> <DAY_NAME> <TIME_VALUE> |
| **Goal Inferred** | : | Flight_ID (Correct) |
| **User** | : | *tell me the airports in new york city area* (Class D) |
| **Concepts** | : | <AIRPORT> <CITY_NAME> (from current query) |
| **Goal Inferred** | : | <FLIGHT> <FROM> <CITY_ORGIN> <TO> <CITY_DESTINATION> <DAY_NAME> <TIME_VALUE> (inherited from the previous query) Flight_ID (Wrong – the correct goal should be Airport_Code, the additional concepts inherited from the previous query results in the wrong identification of the information goal.) |

### C. Context Inheritance

The ATIS corpus contains both Class A and Class D queries. While the semantics of the Class A queries are self-contained, those of the Class D queries are context-dependent. Interpretation of the Class D queries requires referencing discourse context from previous dialog turns. Consequently, we have enhanced our BN-based dialog model with the capability of context inheritance for handling ATIS queries.

TABLE XIV

CONTEXT INHERITANCE SCHEME FOR THE BN-BASED DIALOG MODEL. MISSING CONCEPTS AND OPTIONAL CONCEPTS (I.E., CLASS_NAME) ARE DETERMINED BY A SIMPLIFIED BACKWARD INFERENCE PROCEDURE THAT INSTANTIATES ONLY THE GOAL NODE, TOGETHER WITH THE DOUBLE THRESHOLD SCHEME AS MENTIONED IN SECTION V-B1. THE MISSING CONCEPTS ARE INHERITED FROM DISCOURSE CONCEPTS

| | | |
|---|---|---|
| **System** | : | what kind of flight information are you interested in? |
| **User** | : | *please list all the flights from Chicago to Kansas city on June seventeenth.* (Class A) |
| **Goal Inferred** | : | Flight_ID (Correct) |
| **User** | : | *for this flight how much would a first class fare cost* (Class D) |
| **Goal Inferred** | : | Fare_ID (Correct) |
| **Missing Concepts** | : | <CITY_ORIGIN> and <CITY_DESTINATION> these are missing concepts detected by the double threshold scheme during backward inference (see results shaded below). The missing concepts are inherited automatically from discourse. |

**Results from Backward Inference**

| *Concept $C_j$ (subset)* | $P^*(C_j)$ | Binary Decision for $C_j$ | Actual Occurrence for $C_j$ |
|---|---|---|---|
| AIRPORT_NAME | 0.0000 | absent | absent |
| **CITY_ORIGIN** | **0.9629** | **present** | **absent** |
| **CITY_DESTINATION** | **0.9629** | **present** | **absent** |
| CLASS_NAME | 0.2716 | optional | present |
| FARE | 0.8765 | present | present |

In our initial approach to context inheritance, the current query inherits all the semantic concepts from the previous query (of the same dialog session) prior to goal inference. However, this scheme was found to be too aggressive, and the extra concepts affected the goal identification performance (see Table XIII).

By investigating the incorrect inheritance cases in the training sentences (Class A and Class D), we refined our context inheritance scheme by taking three remedial measures.

Context inheritance is invoked only to fill up the semantic slots for missing concepts detected during backward inference. An example is shown in Table XIV.

If a query is classified as OOD, it may be an indication that we are handling a Class D query. In this case, all concepts are inherited from the previous query and goal inference is invoked again after concept inheritance. An example is shown in Table XV.

We apply pragmatic "refresh rules" to undo context inheritance for several query types [8]. These rules refer to the inferred goal of a query and determine the related concepts that should not be inherited from discourse. We have applied three such rules. First, Class D queries with inferred goal

[8]These heuristics are introduced after careful inspection of the ATIS training data. In a real ATIS-like application, we expect that similar heuristics will need to be defined by the system developer.

TABLE XV
CONTEXT INHERITANCE SCHEME FOR AN OUT-OF-DOMAIN QUERY. CONCEPTS FROM THE PREVIOUS QUERY ARE INHERITED AND GOAL INFERENCE IS INVOKED AGAIN AFTERWARDS

| | | |
|---|---|---|
| **System** | : | what kind of flight information are you interested in? |
| **User** | : | *i'd like to fly from miami to chicago on american airlines.* (Class A) |
| **Concepts** | : | <FLIGHT> <FROM> <CITY_ORIGIN> <TO> <CITY_DESTINATION> <PREP> <AIRLINE_NAME> |
| **Goal Inferred** | : | Flight_ID (Correct) |
| **User** | : | *which ones arrive around five p.m.?* (Class D) |
| **Concepts** | : | <TO> <TIME_VALUE> (from current query) |
| **Goal Inferred Concepts Inherited** | : | OOD <FLIGHT> <FROM> <CITY_ORIGIN> <TO> <CITY_DESTINATION> <PREP> <AIRLINE_NAME> (inherited from previous query since the current one is OOD) |
| **Goal Inferred** | : | Flight_ID (Correct – goal inference invoked again after concept inheritance for an OOD query) |

TABLE XVI
GUIDELINES FOR ASSIGNING VALUES TO $P(C_j = 1|G_i = 1)$

| Condition | Probability of $P(C_j=1\|G_i=1)$ |
|---|---|
| *1.* $C_j$ must occur given $G_I$ | 0.95 – 0.99 |
| *2.* $C_j$ often occurs given $G_i$ | 0.7 – 0.8 |
| *3.* $C_j$ may occur given $G_I$ | 0.4 – 0.6 |
| *4.* $C_j$ seldom occurs given $G_i$ | 0.2 – 0.3 |
| *5.* $C_i$ never occurs given $G_i$ | 0.01 – 0.1 |

`Airline_Code` will disinherit the concepts ⟨AIRLINE_NAME⟩ and ⟨AIRLINE_CODE⟩ from the previous discourse (since the user is clearly asking about the airline of a flight). Second, Class D queries with inferred goal `Airline_Name` does not inherit discourse since the user is simply asking about an airline name, e.g., *"what does y x mean?"* Third, Class D queries with inferred goals `Flight_ID` or `Fare_ID` disinherit the concept ⟨FARE_CODE⟩.

## V. PORTABILITY OF THE BN-BASED FRAMEWORK

In addition to scalability, this work also includes a preliminary examination of the portability of the BN-based framework across different application domains. Migration to a new application often implies lack of domain-specific data to train the BN probabilities. Under such circumstances, the BN probabilities can be hand-assigned to reflect the "degree of belief" of the knowledge domain expert. The trained and the hand-assigned models have similar complexities as they both attempt to capture the probability distribution of ATIS. However, the hand-assigned model requires human knowledge in order to decide the BN probabilities. We have designed some general principles for probability assignment, as will be presented in Sections V-A and B. As explained previously, since the ATIS corpus provides reference semantic frames but not mixed-initiative interactions, our evaluation for portability focuses on the ability of the ported BNs to correctly identify goals and concepts in the user's query. We assume that BN probabilities that achieve good performance in goal and concept identification should have captured domain-specific constraints well. These constraints can, in turn, be used for automatic detection of missing or spurious concepts in order to drive mixed-initiative dialog modeling, as we illustrated previously with the CU FOREX system.

### A. BN Design and Probability Assignment

Under the condition that there is little or no training data, we do not have a statistical basis for selecting the relevant concepts for each BN [18]. An alternative method that does not rely on statistics is to use human judgment to identify the concepts that are directly relevant to each goal. Doing so for all the 11 goals in ATIS extracts a set of 23 concepts in total. Among these, 13 are semantic concepts that correspond to SQL attributes for database access, e.g., ⟨AIRPORT_NAME⟩, ⟨AIRLINE_NAME⟩, ⟨TRANSPORT_TYPE⟩; and the remaining ones are concepts based on keywords, e.g., ⟨AIRCRAFT⟩, ⟨FARE⟩ and ⟨FROM⟩. For the sake of simplicity, we assumed independence among concepts in the BN (i.e., adopt the simple topology), and develop 11 BNs with 23 concepts each. We then hand-assigned the four probabilities for each BN, namely $P(C_j = 1|G_i = 1)$, $P(C_j = 0|G_i = 1)$, $P(C_j = 1|G_i = 0)$, $P(C_j = 0|G_i = 0)$. In the following we describe the general principles for assigning $P(C_j = 1|G_i = 1)$ and $P(C_j = 1|G_i = 0)$ The remaining probabilities can be derived since $P(Cj = 0|Gi = 1) = 1 - p(Cj = 1|G_I = 1)$; and $P(Cj = 0|Gi = 0) = 1 - P(C_j = 1|G_i = 0)$.

*1) Probability Assignment for* $P(C_j{=}1|G_i{=}1)$*:* Table XVI displays the guidelines by which we assign values to the probabilities $P(C_j = 1|G_i = 1)$. The assignment is based on human judgment of the possible occurrence frequency of a concept $C_j$ in queries of goal $G_i$. As mentioned previously, these probabilities may also be derived from the business logic expressed by the database schema (i.e., which attributes are important for what goal), though this is currently regarded to fall beyond the scope of this paper.

Case 1) **$C_j$ must occur given $G_i$**

If we identify a concept $C_j$ to be mandatory for a query of goal $G_i$, we will hand-assign a high probability roughly from 0.95 to 0.99 for $P(C_j = 1|G_i = 1)$. For example, this applies to the ⟨FARE⟩ (corresponding to the words *fare*, *price*, etc.) which must occur in a Fare_ID query. (e.g., "*what is the first class* ***fare*** *from detroit to las vegas*" and "*show me the first class and coach* ***price***").

TABLE XVII
GUIDELINES FOR ASSIGNING VALUES TO $P(C_j = 1|G_i = 0)$

| Condition | Probability of $P(C_j=1\|G_i=0)$ |
|---|---|
| *1.* $C_j$ always occurs for goals other than $G_i$ | 0.7 – 0.9 |
| *2.* $C_j$ sometimes occurs for goals other than $G_i$ | 0.2 – 0.5 |
| *3.* $C_j$ seldom occurs for goals other than $G_i$ | – 0.1 |

Case 2) $\boldsymbol{C_j}$ **often occurs given** $\boldsymbol{G_i}$
If the concept often occurs with the $G_i$ query (e.g., a Fare_ID query often occurs ⟨CITY_ORIGIN⟩ and ⟨CITY_DESTINATION⟩, we lower the assigned values of $P(C_j = 1|G_i = 1)$ to the range 0.7 to 0.8.

Case 3) $\boldsymbol{C_j}$ **may occur given** $\boldsymbol{G_i}$
This applies to the concepts that act as additional constraints for database access, e.g., ⟨TIME_VALUE⟩, ⟨DAY_NAME⟩, ⟨PERIOD⟩. The assigned values for $P(C_j = 1|G_i = 1)$ range between 0.4 and 0.6.

Case 4) $\boldsymbol{C_j}$ **seldom occurs given** $\boldsymbol{G_i}$
This is the case where the occurrence of concept $C_j$ in queries with goal $G_i$ is infrequent. For example, the concept ⟨STOPS⟩ which specify a nonstop flight is not usually associated with Fare_ID queries and the values assigned to $P(C_j = 1|G_i = 1)$ range from 0.2 to 0.3.

Case 5) $\boldsymbol{C_j}$ **never occurs given** $\boldsymbol{G_i}$
If the presence of concept $C_j$ usually implies absence of goal $G_i$, then the probability of $P(C_j = 1|G_i = 1)$ is set to low values between 0.01 and 0.1. For example, the presence of ⟨FLIGHT_NUMBER⟩ usually implies that the query's goal is not flight identification (Flight_ID).

*2) Probability Assignment for $P(C_j{=}1|G_i{=}0)$:* Table XVII displays the guidelines by which we assign values to the probabilities $P(C_j = 1|G_i = 0)$. The assignment is based on human judgment of the possible occurrence frequency of a concept $C_j$ in queries of goals other than $G_i$.

Case 1) $\boldsymbol{C_j}$ **always occurs for goals other than** $\boldsymbol{G_i}$
Consider the relationship between the concept ⟨CITY_ORIGIN⟩ and the goal Aircraft_Code. Since ⟨CITY_ORIGIN⟩ always occurs in *other* informational goals, (e.g., Flight_ID, Fare_ID, etc.), we assign $p\langle C\langle\text{CITY\_ORIGIN}\rangle = 1|G\langle\text{Aircraft\_Code}\rangle = 0)$ in the range from 0.7 to 0.9.

Case 2) $\boldsymbol{C_j}$ **sometimes occurs for goals other than** $\boldsymbol{G_i}$
Consider the relationship between the concept ⟨CLASS⟩ and the goal Aircraft_Code. ⟨CLASS⟩ may occur in the informational goals other than Aircraft_Code to act as an additional constraint for database access. We assign $p\langle C\langle\text{CLASS}\rangle = 1|G\langle\text{Aircraft\_Code}\rangle = 0)$ in the range from 0.2 to 0.5.

TABLE XVIII
PERFORMANCE ACCURACY ON GOAL IDENTIFICATION BY BNS WITH HANDCRAFTED PROBABILITIES COMPARED TO BNS WITH PROBABILITIES TRAINED WITH THE ATJS-3 TRAINING CORPUS. RESULTS ARE BASED ON THE ATJS-3 1993 TEST SET, INCLUDING BOTH CLASS A AND CLASS D QUERIES

| Class | Goal identification accuracy for BNs with handcrafted probabilities | Goal identification accuracy for BNs with trained probabilities |
|---|---|---|
| A (448 queries) | 90.2% | 91.7% |
| D (325 queries) | 68.3% | 74.8% |
| A+D | 80.9% | 84.6% |

TABLE XIX
SENTENCE ERROR RATES OF THE BNS WITH HANDCRAFTED PROBABILITIES COMPARED TO BNS WITH PROBABILITIES TRAINED WITH THE ATIS-3 TRAINING CORPUS. RESULTS ARE BASED ON THE ATIS-3 1993 TEST SET, INCLUDING BOTH CLASS A AND CLASS D QUERIES

| Class | Sentence error rates for BNs with handcrafted probabilities | Sentence error rates for BNs with trained probabilities |
|---|---|---|
| A (448 queries) | 12.1% | 9.2% |
| D (325 queries) | 40.9% | 33.9% |
| A+D | 24.2% | 19.5% |

Case 3) $\boldsymbol{C_j}$ **Seldom Occurs For Goals Other Than** $\boldsymbol{G_i}$
This applies to concepts that are strongly dependent on a specific goal and hence seldom appear with other goals. For example, the concept ⟨TRANSPORT⟩ usually accompanies the goal Ground_Transport only. Hence $p(C\langle\text{TRANSPORT}\rangle = 1|G\langle\text{Ground\_Transport}\rangle = 0)$ is set close to 0.

### *B. Goal Identification Performance With Hand-Assigned Probabilities*

For this preliminary study, we have only conducted experiments with the ATIS-3 1993 test set (Class A and D queries included). BNs with hand-assigned probabilities which assume independent concepts achieved a goal identification accuracy of 80.9%. This compares with 84.6% for BNs with trained topologies based on the ATIS-3 training set. The availability of training data for the BNs enhances performance in goal identification. Queries whose goals are not covered by our 11 BNs are treated as OOD, and are considered to be identified correctly if they are classified as such. Performance accuracy on goal identification is shown in Table XVIII.

Another performance indicator is sentence error rate. A sentence is considered correct only if the inferred goal and extracted concepts in the generated semantic frame agrees with those in the reference semantic frame (derived from the SQL in the ATIS corpora). The sentence error rates are shown in Table XIX. These results lie within the range reported by the ATIS evaluation sites (see Table XX) [29].

TABLE XX
BENCHMARK RESULTS FROM THE 10 ATIS EVALUATION SITES [28], BASED ON THE ATIS-3 1993 TEST SET

| Class | Sentence error rates |
|---|---|
| A (448 queries) | 6.0 – 28.6% |
| D (325 queries) | 13.8 – 63.1% |
| A+D | 9.3 – 43.1% |

## VI. CONCLUSION

This paper describes our first attempt in using Belief Networks (BN) for mixed-initiative dialog modeling. The BN was previously used in natural language understanding to infer the informational goal(s) of the user's query based on the semantic concepts in the query. The BN topology may assume concept independence (for the "simple" topology), or it may model inter-concept dependencies by learning inter-node linkages according to the MDL principle (for the "trained" topology). We extended this framework with the technique of backward inference, i.e., the inferred goal node is instantiated, and probabilities propagate back to the concept nodes within the BN to validate whether a concept should be present or absent. Such validation is based on domain-specific constraints captured in the BN probabilities. Comparison between the validation results and the actual occurrences of the concepts enable the framework to detect missing and spurious concepts automatically. This is used to drive the mixed-initiative dialog model, i.e., the spoken dialog system will prompt for missing concepts and clarify for spurious concepts.

We have demonstrated the feasibility of this approach in the foreign exchange domain, based on the CU FOREX system. This domain is simple, and consists only of two informational goals and five domain-specific concepts. The original CU FOREX system implements two types of interactions. The directed dialog (DD) interaction is system-initiative, and constrains the user to input a single information attribute per dialog turn. The natural language shortcut (NLS) is user-initiative, and allows the user to input multiple attributes per dialog turn but fails if the database retrieval constraints are not met. Our BN-based dialog model aims to transition freely in between DD and NLS. It automatically detects missing and spurious concepts, and prompts for the former while clarifies for the latter. Evaluation based on 550 user calls shows that the BN-based dialog model can handle all the cases of failure for either DD or NLS.

We proceeded to investigate the scalability of the BN-based framework from the CU FOREX domain to the ATIS (air travel) domain. We developed a framework for ATIS that is characterized by eleven information goals and sixty concepts; hence the domain has a significantly greater complexity. We assessed the capability of the BN-based framework in the tasks of the identifying the information goals and extracting the concepts from the user's queries, as well as in the task of detecting missing and spurious concepts. The original detection scheme gave rise to many more missing/spurious concepts, which may lead to redundancies in the dialog model. As a remedial measure, we propose the use of a double threshold scheme for missing/spurious concept detection. Analysis showed that a possible cause for dialog redundancies is the existence of "equivalence classes" in concept sets that provide sufficient constraints for database access. For example, the concept set (CITY_ORIGIN, CITY_DESTINATION) and the set (AIRLINE_NAME, FLIGHT_NUMBER) both provide sufficient constraints for retrieving an Aircraft_Code. However, occurrences of (AIRLINE_NAME, FLIGHT_NUMBER) in Aircraft_Code queries are few in the training corpus, hence the trained BN topology did not capture the equivalence class. We propose to hand-insert inter-concept linkages in the BN topology prior to training their probabilities. We showed that using hand-refined BNs could help in eliminating the dialog redundancies described above, and we can replace the double threshold detection scheme with a single-threshold scheme, which avoids setting *ad hoc* threshold values. We have also defined a set of "optional" concepts according to training data statistics. If a spurious concept is detected during backward inference, but if the concept also belongs to our optional set, it will not invoke clarification prompts in the dialog model. In order to handle the Class D (context-dependent) ATIS queries, we have endowed the BN framework with the capability of context inheritance. The current user query will only inherit missing concepts from discourse. However, if the current query is deemed out-of-domain (OOD) during goal inference, it will inherit all concepts from discourse and then invoke goal inference again. The performance of the BN-based framework in goal identification and sentence error rate falls within the range reported in the 10 ATIS evaluation sites.

In addition, we have investigated the portability of the BN-based framework across application domains. Porting to a new domain often implies lack of training data. Hence we have developed a set of principles for developing BNs and hand-assigning BN probabilities based on the "degree of belief" in the relationships between domain-specific concepts and goals. The BNs with hand-assigned probabilities gave respectable performance in goal identification and sentence error rates, but this performance can be significantly improved with the availability of training data. In the future, we plan to explore the use of BN as a general framework for mixed-initiative dialog modeling, and incorporate dialog strategies that aim to maximize an overall evaluation criterion for the dialog's usability.

## APPENDIX

This appendix elaborates on the process of probability propagation in a Belief Network with trained topology, using Fig. 2 as an example. We begin by a set of joint probabilities of the two cliques $(G, C_1, C_2)$ and $(G, C_3)$ as shown in Table XXI.

Suppose the input user query contains the concepts $C_1$ and $C_2$ but not $C_3$. Hence the evidence for the BN include $C_1 = 1$, $C_2 = 1$ and $C_3 = 0$. We proceed to infer the presence or absence of the goal $G$ based on this evidence.

Step 1) From the evidence $C_1 = 1$ we derive $P^*(C_1 = 1) = 1$. The asterisk $(*)$ associated with a probability indicates that it has been updated by instantiation of some node(s).

TABLE XXI
THE JOINT PROBABILITIES FOR THE TWO CLIQUES. THE VECTOR $(\alpha, \beta)$ IN THE TABLE CORRESPONDS TO $(C_2 = 1, C_2 = 0)$

| | $P(G, C_1, C_2)$ | |
|---|---|---|
| | $G=1$ | $G=0$ |
| $C_1=1$ | (0.32,0.08) | (0.03,0.12) |
| $C_1=0$ | (0.05,0.05) | (0.035,0.315) |
| | $P(G, C_2)$ | |
| | $G=1$ | $G=0$ |
| $C_3=1$ | 0.275 | 0.2 |
| $C_3=0$ | 0.225 | 0.3 |

TABLE XXII
THE UPDATED JOINT PROBABILITY $P^*(C_1, C_2, G)$ WHICH REDUCES TO $P^*(C_2, G)$ BASED ON THE INPUT EVIDENCE $P^*(C_1 = 1) = 1$

| | $P^*(C_1, C_2, G)$ | |
|---|---|---|
| | $G=1$ | $G=0$ |
| $C_1=1$ | (0.32,0.08)×1/0.55 | (0.03,0.12)×1/0.55 |
| $C_1=0$ | (0.05,0.05)×0 | (0.035,0.315)×0 |
| | ⇓ | |
| | $P^*(C_1, G)$ | |
| | $G=1$ | $G=0$ |
| $C_2=1$ | 0.5818 | 0.0545 |
| $C_2=0$ | 0.1455 | 0.2182 |

Since $C_1$ belongs to the clique $(G, C_1, C_2)$, we update its joint probability according to (A.1)

$$P^*(C_1, C_2, G) = P(C_1, C_2, G)\,\frac{P^*(C_1)}{P(C_1)}. \tag{A.1}$$

We substitute into (A.1) the values $P^*(C_1 = 1) = 1$ and $P(C_1 = 1) = 0.55$ which can be obtained from training data. The updated joint probabilities $P^*(C_1, C_2, G)$ are as shown in Table XXII and also reduced to $P^*(C_2, G)$. From this we can marginalize to get $P^*(C_2) = (0.6363, 0.3637)$.

Step 2) From the evidence $C_2 = 1$ we derive $P^*(C_2 = 1) = 1$. The joint probability $P^{**}(C_2, G)$ is updated again as shown in (A.2):

$$P^{**}(C_2, G) = P^*(C_2, G)\frac{P^{**}(C_2)}{P^*(C_2)}. \tag{A.2}$$

We substitute into (A.2) the values $P^{**}(C_2 = 1) = 1$ (according to the evidence from the input query) and $P^*(C_2 = 1) = 0.6363$ (according to the results from Step 1). The updated joint probabilities $P^{**}(C_2, G)$ are as shown in Table XXIII and by marginalization we obtain $P^*(G) = (0.9143, 0.0857)$.

Step 3) Since $G$ is the separator between the two cliques, we can then update the joint probability $P^*(C_3, G)$ in the clique $(G, C_3)$ based on the results $P^*(G)$ from Step 2. Details are shown in (A.3) and Table XXIV. We used $P(G) = 0.5$. Note also from the results in Table XXIV that we can marginalize to get $P^*(C_3) = (0.5372, 0.4628)$.

$$P^*(C_3, G) = P(C_3, G)\frac{P^*(G)}{P(G)} \tag{A.3}$$

Step 4) From the evidence $C_3 = 0$, we derive $P^{**}(C_3 = 1) = 0$ This can be used together with the resulting $P^*(C_3)$ from Step 3 to further update the joint probability $P^{**}(C_3, G)$ in the current clique. Details are shown in (A.4) and Table XXV.

$$P^{**}(C_3, G) = P^*(C_3, G)\frac{P^{**}(C_3)}{P^*(C_3)} \tag{A.4}$$

TABLE XXIII
THE UPDATED JOINT PROBABILITY $P^{**}(C_2, G)$ BASED ON THE INPUT EVIDENCE $P^*(C_2 = 1) = 1$

| | $P^{**}(C_2, G)$ | |
|---|---|---|
| | $G=1$ | $G=0$ |
| $C_2=1$ | 0.5818×1/0.6363 | 0.0545×1/0.6363 |
| $C_2=0$ | 0.1455×0 | 0.2182×0 |
| | ⇓ | |
| | $P^{**}(C_2, G)$ | |
| | $G=1$ | $G=0$ |
| $C_2=1$ | 0.9143 | 0.0857 |
| $C_2=0$ | 0 | 0 |

TABLE XXIV
THE UPDATED JOINT PROBABILITY $P^*(C_3, G)$ BASED ON $P^*(G)$ FROM THE SEPARATOR NODE

| | $P^*(C_3, G)$ | |
|---|---|---|
| | $G=1$ | $G=0$ |
| $C_3=1$ | 0.275×0.9143/0.5 | 0.2×0.0875/0.5 |
| $C_3=0$ | 0.225×0.9143/0.5 | 0.3×0.0875/0.5 |
| | ⇓ | |
| | $P^*(C_3, G)$ | |
| | $G=1$ | $G=0$ |
| $C_3=1$ | 0.5029 | 0.0343 |
| $C_3=0$ | 0.4114 | 0.0514 |

TABLE XXV
THE UPDATED JOINT PROBABILITY $P^*(C_3, G)$ BASED ON THE EVIDENCE OF THE INPUT QUERY $P^{**}(C_3 = 1) = 0$

| | $P^{**}(C_3, G)$ | |
|---|---|---|
| | $G=1$ | $G=0$ |
| $C_3=1$ | 0.5029×0 | 0.0343×0 |
| $C_3=0$ | 0.4114×1/0.4628 | 0.0514×1/0.4628 |
| | ⇓ | |
| | $P^{**}(C_3, G)$ | |
| | $G=1$ | $G=0$ |
| $C_3=1$ | 0 | 0 |
| $C_3=0$ | 0.889 | 0.1111 |

TABLE XXVI
THE UPDATED JOINT PROBABILITY $P^*(C_1, C_2, G)$ BASED ON THE EVIDENCE OF GOAL INFERENCE $P^*(G = 1) = 1$

| | ***P*(C₁, C₂,G)*** | |
|---|---|---|
| | *G=1* | *G=0* |
| *C₁=1* | (0.32, 0.08)×1/0.5 | (0.03, 0.12)×0 |
| *C₁=0* | (0.05, 0.05)×1/0.5 | (0.035, 0.315)×0 |

⇓

| | ***P*(C₁, C₂,G)*** | |
|---|---|---|
| | *G=1* | *G=0* |
| *C₁=1* | (0.64, 0.16) | (0, 0) |
| *C₁=0* | (0.1, 0.1) | (0, 0) |

Step 5) Having incorporated all evidences from the input query in probability propagation, we can eventually marginalize $P^{**}(C_3, G)$ to obtain $P^{**}(G) = (0.8889, 0.1111)$. If we compare these values with the threshold $\theta = 0.5$, we will conclude that the corresponding goal $G$ is present.

$$P^*(C_1, C_2, G) = P(C_1, C_2, G) \frac{P^*(G)}{P(G)}. \tag{A.5}$$

Hence the updated values are $P^*(C_1 = 1) = 0.8$, $P^*(C_1 = 0) = 0.2$, $P^*(C_2 = 1) = 0.74$ and $P^*(C_2 = 0) = 0.26$ We can use some of these probabilities to infer "backward" regarding the existence of the concept $C_1$. Referring to Table X, the backward inference of concept involves use of the evidence ($C_1$ is present) from the user's query, i.e., $P^{**}(C_1 = 1) = 1$.

$$P^{**}(C_1, C_2, G) = P^*(C_1, C_2, G) \frac{P^{**}(C_1)}{P^*(C_1)}. \tag{A.6}$$

If we substitute the values from Table XXVI, $P^*(C_1 = 1) = 0.8$ and $P^{**}(C_1 = 1) = 1$ into (A.6) we obtain $P^*(C_2 = 1) = 0.8$ and $P^*(C_2 = 0) = 0.2$ These values can also be compared with a threshold to decide upon the presence or absence of the concept $C_2$.

## REFERENCES


[1] P. Price, "Evaluation of spoken language systems: The ATIS domain," in *Proc. DARPA Speech and Natural Language Workshop*, San Mateo, CA, 1990, pp. 91–95.

[2] M. Blomberg, R. Carlson, K. Elenius, B. Granstrom, J. Gustafson, S. Hunnicutt, R. Lindell, and L. Neovius, "An experimental dialog system: WAXHOLM," in *Proc. 3rd Eur. Conf. on Speech Communication and Technology*, 1993.

[3] D. Jurafsky, C. Wooters, G. Tajchman, J. Segal, A. Stolcke, E. Fosler, and N. Morgan, "The berkeley restaurant project," in *Proc. 2nd International Conf. on Spoken Language Processing*, Yokohama, Japan, 1994.

[4] H. Meng, S. Busayapongchai, J. Glass, D. Godduea, L. Hetherington, E. Hurley, C. Pao, Polifronri, S. Seneff, and V. Zue, "WHEELS: A conversational system in the automobile classifieds domain," in *Proc. 4th Int. Conf. Spoken Language Processing*, 1996.

[5] V. Zue, S. Seneff, J. Glass, L. Hetherington, E. Hurley, H. Meng, C. Pao, J. Polifroni, R. Schoming, and P. Schmid, "From interface to content: Translingual access and delivery of on-line information," in *Proc. 5th Eur. Conf. Speech Communication and Technology*, Rhodes, Greece, 1997.

[6] L. Lamel, S. Bennacef, J. Gauvain, H. Dartigues, and J. Temem, "User evaluation of the MASK kiosk," in *Proc. 4th Int. Conf. Spoken Language Processing*, Sydney, Australia, 1998.

[7] M. Walker, J. Fromer, and S. Narayanan, "Learning optimal dialogue strategies: A case study of a spoken dialogue agent for email," in *Proc. ACL/COLING 98*, 1998.

[8] J. Gustafson, L. Bell, J. Beskow, J. Boye, R. Carlson, J. Edlund, B. Granstrom, D. House, and M. Wiren, "AdApt—A multimodal conversational dialog system in an apartment domain," in *Proc. 5th Int. Conf. Spoken Language Processing*, Beijing, China, 2000.

[9] M. Walker, J. Aberdeen, J. Boland, E. Bratt, J. Garofolo, L. Hirschman, A. Le, S. Lee, S. Narayanan, K. Papineni, B. Pellom, J. Polifroni, A. Potamianos, P. Prabhu, A. Rudnicky, G. Sanders, S. Seneff, D. Stallard, and S. Whittaker, "DARPA communicator dialog travel planning systems: The June 2000 data collection," in *Proc. 7th Eur. Conf. Speech Communication and Technology*, June 2001.

[10] K. Kita, Y. Fukui, M. Nagata, and T. Morimoto, "Automatic acquisition of probabilistic dialogue models," in *Proc. 4th Int. Conf. Spoken Language Processing*, Sydney, Australia, 1998.

[11] E. Levin, R. Pieraccini, and W. Eckert, "A stochastic model of human-machine interaction for learning dialogue strategies," *IEEE Trans. Speech Audio Processing*, vol. 8, pp. 11–23, Jan. 2000.

[12] M. Walker, "An application of reinforcement learning to dialogue strategy selection in a spoken dialogue system for email," *J. Artif. Intell. Res.*, vol. 12, pp. 387–416, 2000.

[13] M. Walker, D. Litman, C. Kamm, and A. Abella, "PARADISE: A framework for evaluating spoken dialog agents," in *Proc. 35th Annu. Meeting of the Association for Computational Linguistics (ACL)*, Madrid, Spain, 1997.

[14] S. Pulman, "Conversational games, belief revision and bayesian networks," in *Proc. 7th Computerational Linguistics in the Netherlands Meeting*, 1996.

[15] E. Horvitz, "Principles of mixed-initiative user interfaces," in *Proc. ACM SIGCHI Conf.*, 1999.

[16] E. Horvitz and T. Paek, "A computational architecture for conversation," in *Proc. 7th User Modeling Conf.*, 1999.

[17] T. Paek and E. Horvitz, "Conversation as action under uncertainty," in *Proc. Uncertainty in Artificial Intelligence Conference*, 2000.

[18] H. Meng, W. Lam, and C. Wai, "To believe is to understand," in *Proc. 6th Eur. Conf. Speech Communication and Technology*, 1999.

[19] R. Pieraccini and E. Levin, "Stochastic representation of semantic structure for speech understanding," *Speech Commun.*, vol. 11, pp. 283–288, 1992.

[20] S. Miller and R. Bobrow, "Statistical language processing using hidden understanding models," in *Proc. Human Language Technology Workshop*, 1994, pp. 278–282.

[21] R. Kuhn and R. De Mori, "The application of semantic classification trees for natural language understanding," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 17, pp. 449–460, May 1995.

[22] K. Arai, J. Wright, G. Riccardi, and A. Gorin, "Grammar fragment acquisition using syntactic and semantic clustering," in *Proc. 4th Int. Conf. Spoken Language Processing*, 1998.

[23] B. Carpenter and J. Chu-Carroll, "Natural language call routing: A robust, self-organizing approach," in *Proc. 4th Int. Conf. Spoken Language Processing*, 1998.

[24] S. M. Weiss and C. A. Kulikowski, *Computer Systems that Learn: Classification and Prediction Methods from Statistics, Neural Nets, Machine Learning and Expert Systems*. Boca Raton, FL: Morgan Kaufman, 1991.

[25] H. Meng, W. Lam, and K. F. Low, "Learning belief networks for language understanding," in *Proc. Automatic Speech Recognition and Understanding Workshop*, 1999.

[26] V. Jensen, *An Introduction to Bayesian Networks*. New York: UCL, 1996.

[27] H. Meng, S. Lee, and C. Wai, "CU FOREX: A bilingual spoken dialog system for the foreign exchange domain," Proc. Int. Conf. Acoustic, Speech, and Signal Processing, 2000.

[28] L. Hirschman, "MADCOW, Multi-Site data collection for a spoken language corpus," in *Proc. 5th DARPA Workshop on Speech and Natural Language*, Harriman, NY, Feb 1992.

[29] D. Pallet, J. Fiscus, W. Fisher, J. Garofolo, B. Lund, and M. Przybocki, "1993 benchmark tests for the ARPA spoken language program," in *Proc. Spoken Language Technology Workshop*, 1994.

**Helen M. Meng** (M'98) received the S.B., S.M., and Ph.D. degrees, all in electrical engineering, from the Massachusetts Institute of Technology, Cambridge.

She has been a Research Scientist at the MIT Spoken Language Systems Group, where she worked on multilingual conversational systems. She is currently an Associate Professor in the Department of Systems Engineering and Engineering Management, The Chinese University of Hong Kong, which she joined in 1998. She also established and directs the Human–Computer Communications Laboratory in the same department since 1999. Her research interest is in the area of human–computer interaction via multilingual spoken language systems, and related techniques in speech recognition, natural language understanding, discourse and dialog modeling language generation and speech synthesis. She is also working on translingual speech retrieval technologies. She serves on the editorial board of *Computer Speech and Language*.

Dr. Meng is a member of the IEEE Signal Processing Society Speech Technical Committee, Sigma Xi, and the International Speech Communication Association.

**Carmen Wai** received the B.Eng. and M.Phil. degrees in 1998 and 2001 respectively, in the Department of Systems Engineering and Engineering Management at the Chinese University of Hong Kong.

Her research interests are in natural language understanding and dialog modeling for the human–computer conversational system.

**Roberto Pieraccini** (M'90) graduated in electrical engineering from the Universita' degli Studi di Pisa, Pisa, Italy, in 1980.

From 1981 to 1990, he was a Researcher with CSELT, Torino, Italy. In June 1990, he joined AT&T Bell Laboratories, Murray Hill, NJ, as a Member of Technical Staff, and from 1995 to 1999 he was with AT&T Sahannon Laboratories, Florham Park, NJ. Since November 1999, he is leading the Natural Dialog Group at SpeechWorks International, New York, a R&D team for the advancement of the spoken dialog systems technology. During his carrier he conducted research on speech recognition, language modeling, spoken dialog systems and machine learning, and authored more than 100 papers on these subjects.