Scalability and Portability of a Belief Network-based Dialog Model for Different Application Domains

Carmen Wai
The Chinese University of Hong Kong
Shatin, N.T., Hong Kong
SAR, China
Tel: +852 2609 8327
cmwai@se.cuhk.edu.hk

Helen M. Meng
The Chinese University of Hong Kong
Shatin, N.T., Hong Kong
SAR, China
Tel: +852 2609 8327
hmmeng@se.cuhk.edu.hk

Roberto Pieraccini
SpeechWorks International Ltd
17 State Street
New York, NY 1004
Tel: +1.212.425.7200
roberto.pieraccini@speechworks.com

ABSTRACT

This paper describes the scalability and portability of a Belief Network (BN)-based mixed initiative dialog model across application domains. The Belief Networks (BNs) are used to automatically govern the transitions between a system-initiative and a user-initiative dialog model, in order to produce mixed-initiative interactions. We have migrated our dialog model from a simpler domain of foreign exchange to a more complex domain of air travel information service. The adapted processes include: (i) automatic selection of specified concepts in the user's query, for the purpose of informational goal inference; (ii) automatic detection of missing / spurious concepts based on backward inference using the BN. We have also enhanced our dialog model with the capability of discourse context inheritance. To ease portability across domains, which often implies the lack of training data for the new domain, we have developed a set of principles for hand-assigning BN probabilities, based on the "degree of belief" in the relationships between concepts and goals. Application of our model to the ATIS data gave promising results.

1. INTRODUCTION

Spoken dialog systems demonstrate a high degree of usability in many restricted domains, and dialog modeling in such systems plays an important role in assisting users to achieve their goals. The system-initiative dialog model assumes complete control in guiding the user through an interaction towards task completion. This model often attains high task completion rates, but the user is bound by many constraints throughout the interaction. Conversely, the user-initiative model offers maximum flexibility to the user in determining the preferred course of interaction. However this model often has lower task completion rates relative to the system-initiative model, especially when the user's request falls beyond the system's competence level. To strike a balance between these two 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.

We strive to reduce handcrafting in the design of mixed-initiative dialogs. We propose to use Belief Networks (BN) to automatically govern the transitions between a system-initiative and a user-initiative dialog model, in order to produce mixed-initiative interactions. Previous work includes the use of semantic interpretation rules for natural language understanding, where the rules are learnt by decision trees known as Semantic Classification Trees (SCTs) [6]. Moreover, there is also previous effort that explores the use of machine learning techniques to automatically determine the optimal dialog strategy. A dialog system can be described as a sequential decision process that has states and actions. An optimal strategy can be obtained by reinforcement learning [7, 8]. While the system is interacting with users, it can explore the state space and thus learn different actions.

Our BN framework was previously used for natural language understanding [1,2]. We have extended this model for dialog modeling, and demonstrated feasibility in the CU FOREX (foreign exchange) [3,4] system, whose domain has low complexity. This work explores the scalability and portability of our BN-based dialog model to a more complex application. We have chosen the ATIS (Air Travel Information Service) domain due to data availability.[1]

2. BELIEF NETWORKS FOR MIXED-INITIATIVE DIALOG MODELING – THE CU FOREX DOAMIN

We have devised an approach that utilizes BNs for mixed-initiative dialog modeling, and demonstrated its feasibility in the CU FOREX domain. Details can be found in [4]. We provide a brief description here for the sake of continuity.

CU FOREX is a bilingual (English and Cantonese) conversational hotline that supports inquiries regarding foreign exchange. The domain is relatively simple, and can be characterized by two query types (or informational goals – Exchange Rate or Interest Rate); and five domain-specific concepts (a CURRENCY pair, TIME DURATION, EXCHANGE RATE and INTEREST RATE). Our approach involves two processes:

2.1 Informational Goal Inference

A BN is trained for each informational goal. Each BN receives as input the concepts that are related to its corresponding goal. In CU FOREX, there are two BNs, each with five input concepts. The pre-defined BN topology shown in Figure 1 (without dotted arrow) incorporates the simplifying assumption that all concepts are dependent only on the goal, but are independent of one another. This topology can be enhanced by

---

[1] The ATIS data can be licensed from the Linguistic Data Consortium (www.ldc.upenn.edu).

learning the inter-concept dependencies from training data according to the Minimum Description Length (MDL) principle [2]. The resultant topology is illustrated in Figure 1.
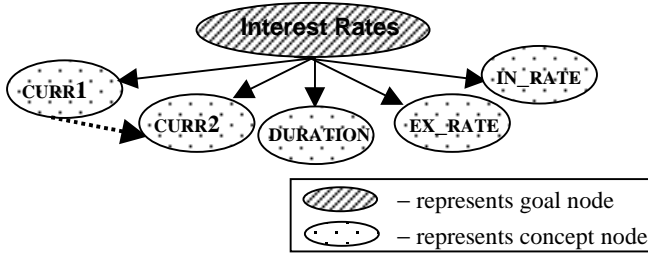

Figure 1. The predefined topology of our BNs is enhanced by the linkage (dotted arrow) learnt to capture dependencies among concepts. The arrows of the acyclic graph are drawn from cause to effect.

Given an input query, each trained BN will make a binary decision (using pre-set threshold of 0.5)[2] regarding the presence or absence of its corresponding informational goal, based on the presence or absence of its input concepts in the query. The decisions across all BNs are combined to identify the informational goal of the input query. We labeled the query to a goal if the corresponding BN votes positive with the maximum *aposteriori* probability. Alternatively, we may label the query with all goals for which the BNs vote positive. Should all BNs vote negative, the query is rejected as out-of-domain (OOD).

2.2 Detection of Missing / Spurious Concepts

Automatic detection of missing or spurious concepts is achieved by *backward inference* in the BN. Given an identified goal from the previous process, the goal node of the corresponding BN is instantiated (i.e. $P(G_i)$ set to *1*), and backward inference updates the probability of each concept $P(C_i)$. Comparison between $P(C_i)$ and a pre-set threshold θ *(=0.5)* determines whether the concept should be present or absent; and further comparison with the actual occurrence(s) determines whether the concept is missing or spurious. In this way, domain-specific constraints for database access is captured and enforced in the BN, i.e. an Exchange Rate inquiry requires a currency pair, and an Interest Rate inquiry requires specifications of the currency and the duration. A missing concept will cause the dialog model to automatically trigger a system prompt. A spurious concept will cause automatically trigger a request for clarification.

Table 1 provides an illustrative example from the CU FOREX domain. The first process infers that the query "Can I have the interest rate of the yen?" has the informational goal of Interest Rate. The second process of backward inference indicates that the concept <DURATION> should be present, but is absent from the query. Hence <DURATION> is a missing concept and the dialog model prompts for the information.

Table 1. This table steps through our dialog modeling process. The input query is "Can I have the interest rate of the yen". Process 1 (informational goal inference) identifies that this is an interest rate inquiry. Process 2 performs backward inference to compute the concept probabilities. Thresholding with $\theta=0.5$ indicates whether the concept should be present or absent. Comparison between this binary decision and the actual occurrence detects that the concept <DURATION> is missing. Hence the dialog model prompts for the missing information.

Query: *Can I have the interest rate of the yen?*			
Process 1: Informational Goal Inference			
BN for Interest Rate $P(Goal = Interest Rate \mid Query) = 0.801 \rightarrow$ goal present			
BN for Exchange Rate $P(Goal = Exchange Rate \mid Query) = 0.156 \rightarrow$ goal absent			
Hence, inferred goal is Interest Rate.			
Process 2: Detection of Missing / Spurious Concepts			
Concept C_i	$P(C_i)$	Binary Decision for C_i	Actual Occurrence of C_i
CURRENCY1	0.91	present	present
CURRENCY2	0.058	absent	absent
DURATION	**0.77**	**present**	**absent**
EXCHANGE_RATE	0.011	absent	absent
INTEREST RATE	0.867	present	present
Response: *How long would you like to deposit?*			

3. MIGRATION TO THE ATIS DOMAIN

Our experiments are based on the training and test sets of the Air Travel Information Service (ATIS) domain. ATIS is a common task in the ARPA (Advanced Research Projects Agency) Speech and Language Program in the US. We used the 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, 773 (1993 test), 732 (1994 test) transcribed utterances respectively. Each utterance is accompanied with its corresponding SQL query for retrieving the relevant information.

We derive the informational goal for each utterance from the main attribute label of its SQL query. 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 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, where both syntactic (e.g. <PREPOSITION> <SUPERLATIVE>) and semantic concepts (e.g. <DAY_NAME>, <FLIGHT_NUMBER>) are present. Hence, ATIS presents increased domain complexity, which is characterized by 11 query types and total 60 domain-specific concepts.

---

[2] We choose threshold at 0.5 since $P(G=1/\mathbf{C})+P(G=0/\mathbf{C})=1$

4. SCALABILITY OF A BN-BASED DIALOG MODEL

4.1 Informational Goal Inference

There is a total of 60 hand-designed[3] semantic concepts in the ATIS domain. In order to constrain computation time for goal inference, we have limited the number of semantic concepts (N) that are indicative of each goal G_j. The parameter N (=20) has been selected using the Information Gain criterion to optimize on overall goal identification accuracy on the Class A training utterances [1].

We have also refined the pre-defined topology using Minimum Description Length (MDL) principle to model concept dependencies. Example of the BN is shown in Figure 2. Their inclusion brought performance improvements in goal identification [2].
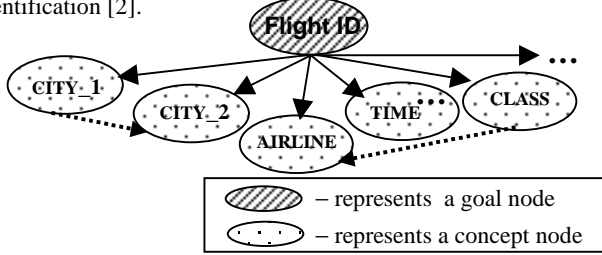

Figure 2. Topology of the BNs for the informational goal Flight_ID.

Consequently, each BN has a classification-based network topology – there are N (=20) input concept nodes (e.g. airline, flight_number, etc.) and a single output node. To avoid the use of sparsely trained BNs, we have developed 11 BNs to capture the domain-specific constraints for each informational goal using Class A training data. The remaining goals are then treated as out-of-domain.

A trained BN is then used to infer the presence / absence of its corresponding informational goal, based on the input concepts. According to the topology shown in Figure 1, the learnt network is divided into sub-networks: {Flight_ID, CITY_1, CITY_2}, {Flight_ID, AIRLINE, CLASS}, {Flight_ID, TIME}, etc. The updated joint probabilities are iteratively computed according to the Equation (1) by each sub-network, the aposteriori probability $P^*(G_i)$ is computed by the marginalization of the updated joint probability $P^*(G_i, C)$. $P^*(G_i)$ is then compared to a threshold (θ) to make the binary decision.

$$P^*(G_i, \bar{C}) = P(G_i | \bar{C})P^*(\bar{C}) \rightarrow P^*(G_i, \bar{C}) = \frac{P(G_i, \bar{C})}{P(\bar{C})}P^*(\bar{C}) \quad (1)$$

where $P^*(C)$ is instantiated according the presence or absence of the concepts; $P(G_i, C)$ is the joint probability obtained from training and $P^*(G_i, C)$ is the updated joint probability

The binary decisions across all BNs are combined to identify the informational goal of the input query. We may label the query to a goal if the corresponding BN votes positive with the highest *aposteriori* probability. Alternatively, we may label the query with all the goals for which the BNs votes positive. Should all BNs vote negative, the input query is rejected as out-of-domain (OOD).

<hr />

[3] We have included the concepts/attributes needed for database access, as well as others that play a syntactic role for natural language understanding.

4.2 Detection of Missing / Spurious Concepts

Having inferred the informational goal of the query, the corresponding node (goal node) is instantiated, and we perform backward inference to test the networks' confidence in each input concept. In this way, we can test for cases of *spurious* and *missing* concepts, and generate the appropriate systems response.

When the goal node is instantiated for backward inference, the joint probability of $P(C, G_i)$ will be updated for each sub-network by Equation 2:

$$P^*(\bar{C}, G_i) = P(\bar{C} | G_i)P^*(G_i) \quad (2)$$

where $P^*(G)$ is updated and instantiated to 1, $P(C|G_i)$ is the conditional probability obtained from training data and $P^*(C, G_i)$ is the updated joint probability

By marginalization, we can get $P(C_j)$. We have pre-set threshold 0.5 for the CU FOREX domain to determine whether the concept should be present or absent. However, when the dialog modeling using single threshold scheme is applied to the ATIS domain, we often obtained *several* missing / 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

Our BN for AIRCRAFT_CODE performed backward inference and the results in Table 2 using single threshold scheme indicated that the concepts <ORIGIN> and <DESTINATION> are missing, while <FLIGHT_NUMBER> is spurious. One reason is because in the training data, most queries with the goal Aircraft_Code provided the city pair instead of the flight number, but *both* serve equally well as an additional specification for database access. If our dialog model followed through with these detected missing and spurious concepts, it would prompt the user for the city of origin, then the city of destination; and then clarify that the flight number is spurious. In order to avoid such redundancies, we defined two thresholds for backward inferencing, as follows:

$$P(C_j) \begin{cases} >= \theta_{upper} \rightarrow C_j \text{ should be } present \text{ in the given } Gi \text{ query} \\ < \theta_{upper} \text{ and } >= \theta_{lower} \rightarrow C_j \text{ is } optional \text{ in the given } Gi \text{ query} \\ < \theta_{lower} \rightarrow C_j \text{ should be } absent \text{ in the given } Gi \text{ query} \end{cases}$$

Hence concepts whose probabilities (from backward inference) scores between θ_{upper} and θ_{lower} will not take effect in response generation (i.e. prompting / clarification). Concepts whose scores exceed θ_{upper}, and also correspond to an SQL attribute will be prompted if missing; and concepts whose scores scant θ_{lower}, and correspond to an SQL attribute will be clarified if spurious. By minimizing number of dialog turns interacting with the users in the training data, we have empirically adopted 0.7 and 0.2 for θ_{upper} and θ_{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.

Table 2. Aposteriori probabilities obtained from backward inferencing using *0.5* as threshold for the query "*What type of aircraft is used in american airlines flight number seventeen twenty three?*"

Concept$_j$ (C$_j$) (Part of concepts)	P(C$_j$)	Binary Decision For C$_i$	Actual Occurrence for C$_i$
AIRCRAFT	1.000	present	present
CITY_NAME1	**0.645**	**present**	**absent**
CITY_NAME2	**0.615**	**present**	**absent**
DAY_NAME	0.077	absent	absent
FLIGHT_NUMBER	**0.420**	**absent**	**present**

4.3 Context Inheritance

We attempt to test our framework using ATIS-3 Class A and D queries. As the Class D queries involve referencing discourse context derived from previous dialog turns, we have enhanced our BN-based dialog model with the capability of context inheritance. Since the additional concepts may affect our goal inference, we choose to invoke goal inference *again* (after context inheritance) *only if* query was previously (prior to context inheritance) classified as OOD. Otherwise, the original inferred goal of the query is maintained. This is illustrated in Table 3. Context inheritance serves to fill in the concepts detected missing from the original query. This is illustrated in Tables 4 and 5.

Table 3. Examples of ATIS dialogs produced by the BN-based dialog model. It indicates that the OOD query is inferred again as Flight_ID query after the inheritance of discourse context.

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 query)
System	*Goal Inference: Flight_ID (Concepts pass the domain constraints)*
User	**Which ones arrive around five p.m.? (Class D)**
System	*Goal Inference: Flight_ID. (System first infers this query as OOD, but it retrieves the concepts from the discourse context and infers again to get Flight _ ID.)*

Table 4. Examples of ATIS dialogs produced by the BN-based dialog model with the capability of inheritance for the missing 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 query)
System	*Goal Inference: Flight_ID (Concepts pass the domain constraints)*
User	**For this flight how much would a first class fare cost. (Class D)**
System	*Goal Inference: Fare_ID. (The missing concepts <CITY_NAME1> <CITY_NAME2> are automatically retrieved from the discourse context.)*

Table 5. Aposteriori probabilities obtained from backward inferencing using double threshold scheme for the Class D query "*For this flight how much would a first class fare cost.*" in Table 4. It indicates that the cities of origin and destination are missing.

Concept$_j$ (C$_j$) (Part of concepts)	P(C$_j$)	Decision for C$_i$	Actual Occurrence for C$_i$
AIRPORT_NAME	0.0000	absent	absent
CITY_NAME1	**0.9629**	**present**	**absent**
CITY_NAME2	**0.9629**	**present**	**absent**
CLASS_NAME	0.2716	optional	present
FARE	0.8765	present	present

We inherit discourse context for all the Class D queries. Based on the training data, we have designed a few context refresh rules to "undo" context inheritance for several query types. For example, if the goal of the Class D query is <Airline_Code>, it is obviously asking about an airline, hence the concept <AIRLINE_NAME> will not be inherited.

5. PORTABILITY OF A BN-BASED DIALOG MODEL

In addition to scalability, this work conducts a preliminary examination of the portability of our BN-based dialog models across different application domains. Migration to a new application often implies the lack of domain-specific data to train our BN probabilities. At this stage, BN probabilities can be hand-assigned to reflect the "degree of belief" of the knowledge domain expert.

5.1 General Principles for Probability Assignment

For each informational goal, we have to identify the concepts that are related to the goal. For example, the informational goal Ground_Transportation is usually associated with the key concepts of <AIRPORT_NAME> <CITY_NAME> and <TRANSPORT_TYPE>. After the identification of all concepts for the 11 goals, 23 key concepts (more details below) are extracted from the total 60 concepts. Each of the 11 handcrafted BNs hence receives as input of the identical set of 23 concepts.

13 semantic concepts (out of 23) (e.g. <CITY_NAME>, <AIRPORT_NAME>, <AIRLINE_NAME>) correspond to the SQL attributes for database access, while the reminding 10 correspond to syntactic/semantic concepts (e.g. <AIRCRAFT>, <FARE>, <FROM>). For the sake of simplicity, we assumed independence among concepts in the BN (pre-defined topology), and we then hand-assigned the four probabilities for each of the 11 BNs, namely $P(C_j=1/G_i=1)$, $P(C_j=0/G_i=1)$, $P(C_j=1/G_i=0)$, $P(C_j=0/G_i=0)$. We avoid assigning the probabilities of 1 or 0 since they are not supportive of probabilistic inference. 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 $P(C_j=0/G_i=1)$ and $P(C_j=0/G_i=0)$ can be derived by the complement of the former two probabilities.

5.1.1 Probability Assignment for $P(C_j=1/G_i=1)$

We assign the probabilities of $P(C_j=1/G_i=1)$ based on the occurrence of the concept C_j with the corresponding G_i query as shown in Table 6.

Case 1. C_j must occur given G_i

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

Case 2. C_j often occurs given G_i

If the concept often occurs with the G_i query, then we will lower the probabilities of $P(C_j=1/G_i=1)$ to the range of 0.7-0.8. For example, the Fare_ID query often comes with the concepts of <CITY_ORIGIN> and <CITY_DESTINATION>.

Case 3. C_j may occur given G_i

This applies to the concepts that act as additional constraints for database access. Examples are <TIME_VALUE>, <DAY_NAME>, <PERIOD> specified in the user query.

Case 4. C_j seldom occurs given G_i

The occurrence of this kind of concepts in the user query is infrequent. Example includes the concept <STOPS> which specify the nonstop flight for the Fare_ID query.

Case 5. C_j never occurs given G_i

This kind of concepts usually provides negative evidence for goal inference. Examples include the concept <FLIGHT_NUMBER> in the Flight_ID query. The presence of <FLIGHT_NUMBER> in the input query implies that the goal Flight_ID is unlikely, because the aposteriori probability for the BN Flight_ID is lowered.

Table 6. Conditions for assigning the probabilities $P(C_j=1/G_i=1)$.

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 occur given G_i	0.7 – 0.8
3. C_j may occur given G_I	0.4 – 0.6
4. C_j seldom occur given G_i	0.2 – 0.3
5. C_j must not occur given G_i	0.01 – 0.1

5.1.2 Probability Assignment for $P(C_j=1/G_i=0)$

For assignment the probabilities of $P(C_j=1/G_i=0)$ for BN_i, we have to consider the occurrence of the concepts for goals other than G_i, i.e. for goal G_m (where m ranges between 1 and 11 but is not equal to i). The scheme for assigning $P(C_j=1/G_i=0)$, i.e. probability of concept C_j being present while goal G_i is absent, is shown in Table 7.

Case 1. C_j always occurs for goals other than G_i

Consider the relationship between the concept <CITY> and the goal Aircraft_Code. Since <CITY> always occur for *other* informational goals, (e.g. Flight_ID, Fare_ID, etc.), we assign $P(C<CITY>=1/G<Aircraft\_Code>=0)$ in the range of 0.7-0.9.

Case 2. C_j sometimes occurs for goals other than G_i

Consider the relationship between the concept <CLASS> and the goal Aircraft_Code. Since <CLASS> sometimes occurs in the informational goals other than Aircraft_Code, and acts as the additional constraints for database access, we assign $P(C<CLASS>=1|G<Aircraft\_Code>=0)$ in the range of 0.2-0.5.

Case 3. C_j seldom occurs for goals other than G_i

This applies to the concepts that are strongly dependent on a specific goal and hence seldom appear for other goals. For example, the concept <TRANSPORTATION> usually accompanies the goal Ground_Transportation only. Hence $P(C<TRANSPORTATION>=1/G<Ground\_Transportation>=0)$ is set closed to 0.

Table 7 Conditions for assigning the probabilities $P(C_j=1/G_i=0)$

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	0.01 – 0.1

5.2 Evaluation

BNs with hand-assigned probabilities achieved a goal identification accuracy of 80.9% for the ATIS-3 1993 test set (Class A and D sentences included). This compares to 84.6% when they have been automatically trained on the training data. 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 there are classified as such.

We have compared the handcrafted probabilities with the trained probabilities based on natural language understanding, where the evaluation metric is the 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 goal identification accuracies and the sentence error rates for the ATIS-3 1993 test set are summarized in Table 8. When we compare the our results with the NL understanding results from the 10 ATIS evaluation sites shown in Table 9, our performance falls within a reasonable range.

Table 8 Goal identification accuracies and the sentence error rates of Class A and D queries of ATIS test 93 data for the handcrafted probabilities and automatically trained probabilities respectively.

	Class	BNs (handcrafted probabilities)	BNs (trained probabilities)
Goal ID Accuracy	A (448)	90.18%	91.74%
	D (325)	68.31%	74.78%
	A+D	80.98%	84.61%
Sentence Error Rate	A (448)	12.05%	9.15%
	D (325)	40.92%	33.85%
	A+D	24.19%	19.53%

Table 9 Benchmark NL results from the 10 ATIS evaluation sites [6].

Class	Sentence Error Rate
A (448)	6.0 – 28.6%
D (325)	13.8 – 63.1%
A+D (773)	9.3 – 43.1%

We observed that our strategy for context inheritance may be too aggressive, which leads to concept insertion errors in the generated semantic frame. This is illustrated in the example in Table 10.

Table 10 The case frame for query 3 indicates our context inheritance strategy may be too aggressive which leads to a concept insertion error in the generated semantic frame.

Query 1:	List flights from oakland to salt lake city before six a m Thursday morning *(Our system generates a correct semantic frame.)*	
Query 2:	List delta flights before six a m (Class D) *(Our system generates a correct semantic frame.)*	
Query 3:	List all flights from twelve oh one a m until six a m (Class D) *(Our system detects missing concepts of <CITY_NAME>, which are inherited from discourse)*	
	Case Frame	**SQL Reference**
Goal:	Flight_ID	Flight_ID
Concepts:	CITY_NAME = oakland CITY_NAME = salt lake city DEPARTURE_TIME = twelve oh one a m until six a m AIRLINE_NAME = delta *(a concept insertion error)*	CITY_NAME = oakland CITY_NAME = salt lake city DEPARTURE_TIME = >=1 && <= 600

6. SUMMARY AND CONCLUSIONS

This paper describes the scalability and portability of the BN-based dialog model as we migrate from the foreign exchange domain (CU FOREX) to the relatively more complex air travel domain (ATIS). The complexity of an application domain is characterized by the number of in-domain informational goals and concepts. The presence / absence of concepts are used to infer the presence/absence of each goal, by means of the BN. When a large number of in-domain concepts are available, we used an information-theoretic criterion (Information Gain) to automatically select the small set of concepts most indicative of a goal, and do so for every in-domain goal. Automatic detection of missing / spurious concepts is achieved by backward inference using the BN corresponding to the inferred goal. This detection procedure drives our mixed-initiative dialog model – the system prompts the user for missing concepts, and asks for clarification if spurious concepts are detected. For the simpler CU FOREX domain, detection of missing / spurious concepts was based on a single probability threshold. However, scaling up to ATIS (which has many more concepts) shows that some concepts need to be present, others should be absent, but still others should be optional. Hence we need to use two levels of thresholding to decide if a concept should be present, optional or absent in the query. We have also enhanced our BN-based dialog model with the capability of context-inheritance, in order to handle the context-dependent user queries in the ATIS domain. Discourse context is inherited for the Class D queries, and we invoke goal inference again after context inheritance if a query was previously classified as OOD.

As regards portability, migration to a new application domain often implies the lack of domain-specific training data. Hence we have proposed a set of general principles for probability assignment to the BNs, as a reflection of our "degree of belief" in the relationships between concepts and goals. We compared the goal identification performance, as well as concept error rates between the use of hand-assigned probabilities, and the probabilities trained from the ATIS training set. Results show that the hand-assigned probabilities offer a decent starting performance to ease portability to a new domain. The system performance can be further improved if data is available to train the probabilities.

7. REFERENCES

[1] Meng, H., W. Lam and C. Wai, "To Believe is to Understand," Proceedings of Eurospeech, 1999.

[2] Meng, H., W. Lam and K. F. Low, "Learning Belief Networks for Language Understanding," Proceedings of ASRU, 1999.

[3] Meng, H., S. Lee and C. Wai, "CU FOREX: A Bilingual Spoken Dialog System for the Foreign Exchange Domain," Proceedings of ICASSP, 2000.

[4] Meng, H., C. Wai, R. Pieraccini, "The Use of Belief Networks for Mixed-Initiative Dialog Modeling," Proceeding of ICSLP, 2000.

[5] Kuhn, R., and R. De Mori, "The Application of Semantic Classification Trees for Natural Language Understanding," IEEE Trans. PAMI, Vol. 17, No. 5, pp. 449-460, May 1995.

[6] Pallet, D., J. Fiscus, W. Fisher, J. Garofolo, B. Lund, and M. Przybocki, "1993 Benchmark Tests for the ARPA Spoken Language Program," Proceedings of the Spoken Language Technology Workshop, 1994.

[7] Levin, E., Pieraccini, R., and Eckert, W., "A Stochastic Model of Human-Machine Interaction for Learning Dialogue Strategies", Speech and Audio Processing, IEEE Transactions, Vol 8, pp. 11-23, Jan 2000.

[8] Walker, M., Fromer, J., Narayanan, S., "Learning Optimal Dialogue Strategies: A Case Study of a Spoken Dialogue Agent for Email", in Proceedings of ACL/COLING 98 , 1998.