IMPROVEMENTS ON A BELIEF NETWORK FRAMEWORK FOR NATURAL LANGUAGE UNDERSTANDING OF DOMAIN-SPECIFIC CHINESE QUERIES

Bonnie MOK and *Helen M. MENG*

Human-Computer Communications Laboratory,
Department of Systems Engineering and Engineering Management,
The Chinese University of Hong Kong,
Shatin, N.T., Hong Kong SAR
{oymok, hmmeng}@se.cuhk.edu.hk

ABSTRACT

This paper extends our work on natural language understanding (NLU) using Belief Networks, as proposed in [1]. We have previously devised a method for identifying the user's communicative goal(s) out of a finite set of domain-specific goals. The problem was formulated as making *N* binary decisions, each performed by a Belief Network (BN). This formulation allows for the identification of queries with multiple goals, as well as queries with out-of-domain (OOD) goals. Our current work presents two extensions: (i) migrating our investigation from English to Chinese; and (ii) exploring the alternate formulation of goal identification as making one *N*-ary decision by a *single* BN. Experiments with the AITS (Air Travel Information System) corpus showed that the *N*-ary formulation improved over the *N* binary formulation in terms of single/multiple goal identification accuracies and OOD rejection.

1. INTRODUCTION

This paper extends our previous work [1] on the use of Belief Networks for natural language understanding (NLU). NLU is a key technology in human-computer conversational systems [3,4]. These systems often need to handle information-seeking queries from the user regarding a restricted domain. A given communicative goal may be expressed in a variety of ways by the user. NLU in a domain-specific application requires identification of the user's communicative goal(s) out of a set of finite possibilities characteristic of the domain.

We use BNs with a pre-defined structure (as shown in Figure 1) for communicative goal(s) identification. We first parse the user's query into a sequence of semantic concepts. According to these concepts, we identify the underlying goal(s) by probabilistic inference. The BN structure captures the dependencies between the communicative goal and the relevant semantic concepts in a query. The naive Bayes' configuration in Figure 1 assumes that the concepts are independent of one another.

We previously formulated the goal identification in terms of making *N* binary decisions, each performed by a BN. Given a user's input query, each BN makes a binary decision regarding the presence or absence of its corresponding goal. The decisions are independent of each other, and we noticed a large number of sentences wrongly identified with multiple goals instead of a single goal. In this paper, we investigate the use of an alternative formulation in terms of one *N*-ary decision with the same predefined BN topology. This formulation can also identify single goal, multiple goals and out-of-domain (OOD) goals but with a *single* BN only.

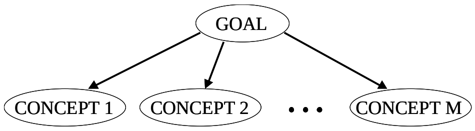

Figure 1. The pre-defined structure of our Belief Network. The arrows of the acyclic graph are drawn from cause to effect. This naive Bayes' topology assumes that the concepts are independent of one another.

2. TASK DOMAIN

We have chosen to work in the air travel domain due the availability of the ATIS (Air Travel Information System) corpora [2]. We have manually translated the Class A sentences of the ATIS-3 corpora, query by query from English to Chinese. The Chinese translation is expressed in spoken Cantonese style. Our corpora consist of a training set (1564 sentences), test set 1993 (448 sentences) and test set 1994 (444 sentences).

As seen from our corpora, there are 32 communicative goals in the ATIS domain. Of these, 11 goals cover over 95% of the training set. Hence we focus on the identification of this set of 11 goals. The remaining goals are treated as "out-of-domain" (OOD). There are 43 training utterances with more than one communicative goal. Table 1 shows two example sentences.

Single goal example	
Original query:	"flights on friday from newark to tampa"
Translated query:	"星期五由紐華克去坦帕既班機 "
Single goal:	flight.flight_id
Multiple goal example	
Original query:	"give me the least expensive first class round trip ticket on u s air from cleveland to miami"
Translated query:	"我想要美國航空由克里夫蘭去邁密最平既頭等來回機位"
Multiple goals:	flight.flight_id, fare.fare_id

Table 1. Single goal and multiple goal examples of translated Chinese sentences from the ATIS-3 Class A training corpus.

3. WORD TOKENIZATION AND PARSING

The Chinese language has no delimiter for word boundaries. We tokenize each Chinese query into words by a maximum matching algorithm. Then, the words are parsed into semantic concepts using hand-designed grammar rules. The sequence of semantic concepts form the input to our BN(s). We have 65 semantic tags for the Chinese ATIS. In comparison, English ATIS has 60 semantic tags. Table 2 is an example to show the processes of word tokenization and parsing.

Original query	話俾我知內陸航空星期日晚九點三十四分後所有由丹佛起飛既班機
Word tokenization	話 / 俾 / 我 / 知 / 內陸航空 /星期日 / 晚 / 九 / 點 / 三 / 十 / 四 / 分 / 後 / 所有 / 由 / 丹佛 / 飛 / 既 / 班機
Semantic concepts	<query> <airline_name> <day_name> <period> <digit> <time_unit> <digit> <time_unit> <pre> <all> <from> <city_name> <to> <既> <flight>
Goal	flight.flight_id

Table 2. An example illustrating the processes of word tokenization and parsing.

4. PREVIOUS FORMULATION (*N* BINARY DECISIONS)

We previously formulated the goal identification problem in terms of making N binary decisions. We developed 11 BNs, one for each selected goal ($N = 11$).

4.1 BN Development

We developed one BN for each of the communicative goal from the training data. Each BN has its own set of semantic concepts that is the most indicative to the corresponding goal. We measure the dependency between a goal and a concept by Information Gain (IG). For a given goal G_i ($i = 1, 2 \ldots 11$), we selected M concepts $\{C_1,C_2\ldots C_M\}$ that have the highest IG in relation with G_i (Equation 1).

$$IG(C_k,G_i)=\sum_{c=0,1}\sum_{g=0,1}P(C_k=c,G_i=g)\log\frac{P(C_k=c,G_i=g)}{P(C_k=c)P(G_i=g)}\ \ldots(1)$$

4.2 Goal Inference

The sequence of semantic concepts present in the user's query form the input to the BN. The BN applies Bayesian inference (Equation 2) and outputs $P(G_i|\mathbf{C})$. This probability is compared with a threshold in order to make the binary decision. We tuned the threshold with the training data by optimizing the F-measure (Equation 3) in goal identification. Precision (P) is the percentage of queries with correct inference out of all queries classified to have the goal G_i. Recall (R) is the percentage of queries correctly inferred with G_i out of all G_i queries. Equation 3 adopts $\beta = 1$ which treats precision and recall with equal importance.

$$P(G_i=1|\vec{C})=\frac{P(G_i=1)\prod_{k=1}^{M}P(C_k=c_k|G_i=1)}{\sum_{g=0,1}[P(G_i=g)\prod_{k=1}^{M}P(C_k=c_k|G_i=g)]}\ \ldots(2)$$

$$F=\frac{(1+\beta^2)RP}{\beta^2R+P}\ \ldots(3)$$

If all the 11 BNs vote *negative* for their corresponding goals, our framework treats the input query as OOD. If only a single BN votes *positive* for its corresponding goal, our framework labels the input query with the goal. If multiple BNs vote *positive* for their corresponding goals, our framework labels the query with multiple goals.

5. CURRENT FORMULATION (ONE *N*-ARY DECISION)

As mentioned earlier, we noticed that the N binary formulation tends to wrongly label single-goal queries with multiple goals. Hence our current work proposes an alternate formulation based on a single N-ary decision. We focus on the same 11 goals and add an extra goal for out-of-domain (OOD) queries. Hence N=12 and $\Sigma_i P(G=g_i|\mathbf{C})= 1$ for $i = 1, 2\ldots12$.

5.1 BN Development

We use IG to measure the dependency between each concept and all the twelve goals. We select the M concepts $\{C_1,C_2\ldots C_M\}$ based on the training data that have the highest IG (Equation 4) as input to the single BN.

$$IG(C_k,G)=\sum_{c=0,1}\sum_{i=1,2\ldots12}P(C_k=c,G=g_i)\log\frac{P(C_k=c,G=g_i)}{P(C_k=c)P(G=g_i)}\ \ldots(4)$$

5.2 Goal Inference

Given a sequence of parsed semantic concepts as input, we perform Bayesian inference (Equation 5).

$$P(G=g_i|\vec{C})=\frac{P(G=g_i)\prod_{k=1}^{M}P(C_k=c_k|G=g_i)}{\sum_{j=1,2..12}[P(G=g_j)\prod_{k=1}^{M}P(C_k=c_k|G=g_j)]}\ \ldots(5)$$

We select a single threshold (θ) based on the training data to compare with the output $P(G=g_i|\mathbf{C})$ for goal identification. Recall g_{12} corresponds to OOD. The decision-making process is as follows:

$$\begin{cases}\text{if}\quad g_{12}=\arg\max_{i=1,2\ldots12}P(G=g_i|\vec{C}) & \rightarrow \text{Label the query as OOD}\\ \text{otherwise,}\ P(G=g_i|\vec{C})\geq\theta\ ,\ i=1,2\ldots11 & \rightarrow \text{Label the query's goal to be } g_i\end{cases}$$

(there can be more than one goals)

6. EVALUATION

The goal identification accuracy is measured in relation to the number of errors in the inferred goals. There are three types of errors. Examples are shown in Table 3.

- Deletion error (DEL) – missing a reference goal
- Insertion error (INS) – inserting an additional inferred goal
- Substitution error (SUB) – incorrectly inferred goal

Deletion error (DEL)	
Query:	我想要由納什維爾飛去西雅圖收費最平既來回航機
Reference goals:	fare.fare_id, flight.flight_id
Inferred goal:	fare.fare_id *(flight.flight_id is missing)*
Insertion error (INS)	
Query:	話我知由夏洛特去拉斯維加斯最平既價錢
Reference goal:	fare.fare_id
Inferred goals:	fare.fare_id, flight.flight_id *(additional)*
Substitution error (SUB)	
Query:	我要星期六由多倫多去華盛頓既班機既價錢
Reference goal:	fare.fare_id
Inferred goal:	flight.flight_id *(incorrect)*

Table 3. Examples of errors in BN inference.

The goal identification accuracy is computed as shown in Equation 6.

$$accuracy = (1 - \frac{\#\ \text{errors}}{\#\ \text{goals in the training/test set}})*100\% \qquad \ldots(6)$$

7. EXPERIMENTS

As mentioned in section 2, our experiments are conducted with ATIS-3 Class A sentences in the training set, test set 1993 and test set 1994. Before the goal identification process, we set the parameters for the BN dimensions and the thresholds.

7.1 Network Dimensions

We conducted experiments based on the training data to determine the number of concept nodes, *M*, to be used in the BNs. We varied the number of input concepts from 10 to the full set of 65 concepts and chose the value for *M* which gives the best goal identification training accuracy.

For the *N* binary formulation, each BN has *M* concept nodes that map to the concepts with the highest values of IG relating to the BN's goal. Figure 2 shows that an appropriate value to use for *M* is 40.

For the one *N*-ary formulation, training accuracies tend to converge beyond 45 concepts, hence we developed a single BN with *M* = 45 (see Figure 3).

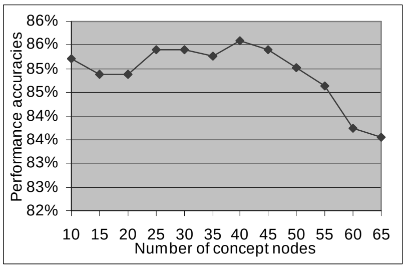

Figure 2. Goal identification training accuracies for the *N* binary formulation. The graph suggests that we use *M* = 40 concepts in each BN.

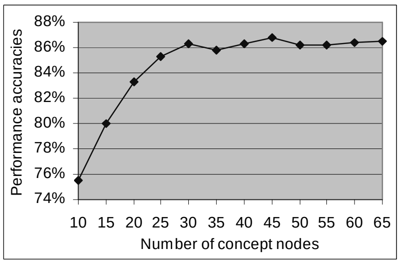

Figure 3. Goal identification training accuracies for the one *N*-ary formulation. The graph suggests that we use *M* = 45 concepts in the single BN.

7.2 Thresholds

For the *N* binary formulation, as mentioned in section 4.2, we applied Equation 3 to tune the threshold in each BN using the training data.

For the one *N*-ary formulation, a threshold is used for capturing multiple goals by comparing it with the output values of $P(G=g_i|\mathbf{C})$ using the training data. Since $\Sigma_i P(G=g_i|\mathbf{C})=1$ for $i = 1, 2\ldots12$, we vary the threshold values between 0 and 1 (as shown in Figure 4). As the threshold value increases, the overall goal identification accuracies (Equation 6) increase but the multiple goal identification performance (F-measure with $\beta = 1$) decreases. We measure the combined performance by taking the average of the both (Equation 7). We notice that the BN cannot capture any multiple goals when the threshold is over 0.5. The results suggest that 0.25 is a suitable threshold.

$$\text{combined performance} = (\text{overall goal identification accuracies} + \text{F-measure in multiple goal identification}) / 2 \qquad \ldots(7)$$

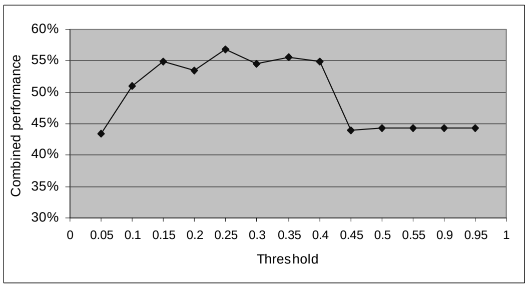

Figure 4. The performance varies with the threshold values in one *N*-ary decision formulation. The graph suggests that 0.25 is a suitable threshold.

7.3 Goal Identification

We compared the performance of the two formulations in terms of: (i) overall goal identification accuracies, (ii) OOD rejection and (iii) multiple-goal identification accuracies.

7.3.1 Overall Goal Identification Accuracies

In this part of work, we expanded our test sets by counting queries with multiple goals multiple times. For example, if a test query has two goals, we treat it as two single goal queries in the evaluation. Hence we count 456 queries in test set 1993 and 450 queries in test set 1994.

The results in Table 4 show that the one *N*-ary formulation gave improvements over the *N* binary formulation in terms of overall goal identification accuracies. This is mainly due to the reduction of insertion errors. In the *N* binary formulation, a query can be labeled as one of 11 goals and the decisions are independent of one another. However, in the one *N*-ary formulation, the goal probabilities $P(G_i|\mathbf{C})$ are dependent as $\Sigma_i P(G_i|\mathbf{C})= 1$ for $i = 1, 2\ldots12$. Thresholding can control (and reduce) the number of single-goal queries wrongly labeled with multiple goals. The effect is illustrated by an example in Table 5. The large number of insertion errors in the *N* binary formulation partially offsets (and reduces) the number of substitution and deletion errors.

Formulation	**One *N*-ary**		***N* binary**	
Test set	**1993**	**1994**	**1993**	**1994**
# deletions (DEL)	5	3	2	1
# insertions (INS)	36	31	86	52
# substitutions (SUB)	59	45	42	35
Total # errors	100	79	130	88
Goal identification accuracies	78.1% (356/456)	82.4% (371/450)	71.5% (326/456)	80.4% (362/450)

Table 4. Comparing the overall goal identification accuracies between the one *N*-ary formulation and the *N* binary formulation. Comparison is based on the number of deletion, insertion and substitution errors produced in test sets 1993 and 1994.

Original query:	“is there ground transportation available at the phoenix airport”
Translated query:	“費尼克斯機場有冇地面交通”
Reference goal:	ground_service.city_code
One *N*-ary formulation	
Inferred goal:	ground_service.city_code *(correct)*
***N* binary formulation**	
Inferred goal 1:	ground_service.city_code *(correct)*
Inferred goal 2:	airport.airport_code *(INS)*

Table 5. An example illustrating an insertion error introduced by the *N* binary formulation. The one *N*-ary formulation labeled the input Chinese query with the correct goal.

7.3.2 OOD Rejection

There are 35 and 37 OOD queries in test sets 1993 and 1994 respectively. We compared the two formulations in terms of appropriate OOD rejection. Results are shown in Table 6, based on precision, recall and F-measure (with $\beta = 1$). Table 7 shows an example of how the *N* binary formulation failed to reject a query which was correctly handled by the one *N*-ary formulation. There are examples of the reverse but better recall for the one N-ary formulation led to higher values for the F-measure overall in OOD rejection.

Formulation	**One *N*-ary**		***N* binary**	
Test set	**1993**	**1994**	**1993**	**1994**
# OOD queries rejected	31	29	25	24
Recall	0.60 (21/35)	0.57 (21/37)	0.51 (18/35)	0.51 (19/37)
Precision	0.68 (21/31)	0.72 (21/29)	0.72 (18/25)	0.79 (19/24)
F-measure	0.64	0.64	0.60	0.62

Table 6. Experimental results comparing the one *N*-ary formulation and *N* binary formulation in OOD rejection.

Original query:	“what's the fare for a taxi to denver”
Translated query:	“係丹佛既地面交通車費要幾多”
Reference goal:	ground_service.ground_fare *(OOD)*
One *N*-ary formulation	
Inferred goal:	OOD
***N* binary formulation**	
Inferred goal:	fare.fare_id

Table 7. An example illustrating how the *N* binary formulation labeled an OOD query with in-domain goal. The one *N*-ary formulation correctly rejected the query and resulted in a better recall.

7.3.3 Multiple Goal Identification

There are 8 and 6 multiple goal queries in test sets 1993 and 1994 respectively. We analyzed the performance in multiple goal identification and results are tabulated in Table 8. Our measure is based on precision, recall and F-measure (with $\beta = 1$). While both formulations gave the same recall values, the one *N*-ary formulation gave better precision values. Table 9 is an example of how the *N* binary formulation wrongly identified a single goal query as multiple-goal query.

Formulation	**One *N*-ary**		***N* binary**	
Test set	**1993**	**1994**	**1993**	**1994**
# MG queries identified	39	34	86	54
Recall	0.38 (3/8)	0.50 (3/6)	0.38 (3/8)	0.50 (3/6)
Precision	0.077 (3/39)	0.088 (3/34)	0.035 (3/86)	0.056 (3/54)
F-measure	0.13	0.15	0.06	0.10

Table 8. Experimental results comparing the one *N*-ary formulation and *N* binary formulation in multiple goal (MG) identification.

Original query:	“which different airlines go from las vegas to new york city”
Translated query:	“有邊間航空公司既航機由拉斯維加斯飛去紐約市”
Reference goal:	airline.airline_code
One *N*-ary formulation	
Inferred goal:	airline.airline_code
***N* binary formulation**	
Inferred goals:	airline.airline_code, flight.flight_id

Table 9. An example illustrating how the *N* binary formulation labeled a single-goal query with multiple goals. Such an error was avoided by using the one *N*-ary formulation.

7.4 Computation

The one *N*-ary formulation requires a single BN while the *N* binary formulation requires 11 BNs. When we train BNs, we estimate the probabilities by tallying the counts from the training data. When we infer query’s goal(s), we perform Bayesian inference. If we compare the two formulations in terms of the number of additive and multiplicative operations, we found that the amount of computation is reduced by 93% during training and by 57% during testing as we migrate from the *N* binary formulation to the one *N*-ary formulation.

8. CONCLUSIONS

This paper extends our work on natural language understanding (NLU) using Belief Networks, as proposed in [1]. We have previously devised a method for identifying the user’s communicative goal(s) out of a finite set of domain-specific goals. The problem was formulated as making *N* binary decisions, each performed by a Belief Network (BN). This formulation allows for the identification of queries with multiple goals, as well as queries with out-of-domain (OOD) goals. Our current work is based on the ATIS-3 corpus translated from English to Chinese. We proposed a new formulation for goal identification that involves a single BN making an *N*-ary decision. The results show the one *N*-ary formulation bought improvements in (i) overall goal identification accuracies, (ii) out-of-domain rejection and (iii) multiple goal identification accuracies. The one *N*-ary formulation also reduced computation by 93% in training BN(s) and by 57% in goal inference during testing.

9. REFERENCES

[1] Meng, H., et al. "To Believe is to Understand," Proceedings of Eurospeech 1999.

[2] Linguistic Data Consortium http://www.ldc.upenn.edu

[3] Arai, K., et al., "Grammar Fragment Acquistion using Syntactic and Semantic Clustering," Proceedings of the ISCSLP 1998.

[4] Carpenter, B and J. Chu-Carroll, "Natural Language Call Rounting: A Robust, Self-Organizing Approach,” Proceedings of the ICSLP 1998.