Improvement on Belief Network Framework for Natural Language Understanding

莫靄欣
MOK, Oi Yan

A Thesis Submitted in Partial Fulfilment
of the Requirements for the Degree of
Master of Philosophy
in
Systems Engineering and Engineering Management

August 2003

Abstract

This thesis extends the investigation into the Belief Network (BN) framework for natural language understanding (NLU) as proposed in [31]. A method was derived 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 BN. We aim to improve the goal identification performance and reduce the amount of computation in training and testing. We explore an alternate formulation as making one N-ary decision by a single BN. This formulation captures the interdependency among the goals. In order to identify multiple goals in a single BN, we propose two goal identification strategies: multiple selection strategy and maximum selection strategy. We evaluate the goal identification performance by accuracy measure, macro- and micro-averaging. Experiments with the ATIS (Air Travel Information Service) corpus showed that the one N-ary formulation improved over the N binary formulation in terms of overall goal identification, out-of-domain rejection and multiple goal identification. A considerable amount of computation was reduced as we migrate from the N binary formulation to the one N-ary formulation. We also test the language portability of the BN framework on Cantonese Chinese. The test used the one N-ary formulation with the maximum selection strategy and the results were encouraging.

摘要

本論文主要是伸延對信念網絡 (Belief Network) 架構在自然語言理解上的研究 [31]。此方法是爲了從有限特定領域的目標裡識別出用户的交流目標。這個問題是以多個二元 (N binary) 決策的信念網絡的公式來表示。我們的目的是改善目標識別的表現及減少在訓練和測試上的計算。我們探究另一種一個多元 (one N-ary) 決策的信念網絡的公式。這公式捕取不同目標之間的互相依賴。爲了從一個信念網絡中識別多種目標，我們提議了兩種識別目標的策略：多種選擇策略及最大選擇策略。我們以準確度量度、宏觀及微觀平均來評估目標識別的表現。在航空資訊 (ATIS) 領域語料庫中得出的實驗結果證明一個多元的公式在全部的目標識別，在領域之外目標的拒絕及多種目標識別上都較多個二元公式有所改進。並且當我們從多個二元的公式轉移到一個多元公式時，可以減省可觀數量的計算。我們還測試信念網絡架構在廣東話式中文上的語言可移植性。在使用附有最大選擇策略的一個多元的公式的測試中可得出鼓勵性的結果。

Acknowledgments

First of all, I would like to thank my supervisor, Professor Helen Meng, for the precious advices and endless supports to my research. I am grateful for being her master student in the past two years. She sacrificed her valuable time for guiding me through the research road. She frankly pointed out my weaknesses and helped me to improve myself. I believe that what I learnt from her is useful for my whole life.

I wish to express my gratitude to Professor Lide Wu and Professor Xuanjing Huang from Fudan University, for travelling to Hong Kong and providing precious suggestions. I would also like to thank my internal thesis committee, Professor Chun Hung Cheng and Professor Wai Lam from the Chinese University of Hong Kong, for their time and valuable comments.

I want to take this chance to say thanks to my friends from HCCL. Special thanks to Tiffany, Yuk and Ada for their encouragements and accompanying me during the hard time. Thanks to Carmen and Ah Fan for teaching me BN when I was a beginner. Thanks to Kin, Homa, Tony, Michael Lo and May for playing shuttlecock with me. Thanks to Ida, Ma Bin, Lo sir, Winnie, Michael Lau, Xu Kui, Ka Fai and Simon for their friendship and laughers. I also want to thank all 219 fellows for providing funny games and tasty snacks. Big thanks to Angie and Polly for the sharing and supports.

My heartfelt thanks to my family for their love, caring and providing a sweet home for me to rest when I am tired. I also want to thank my old friend Karen Sin for her cheering. Thanks to Minnie, Madeline, Ming, Joey and Joyce for bringing me confidence when I was frustrated.

Finally, I would like to thank dear God for giving me strength and leading me to go on. Thanks my cell group members – Lisa, Dymo, Karen Yip, Beatrice, Aarus, Janis and Fu – for the caring and prayers.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Overview

In this information era, computers have already permeated our lives. The development of the Internet and mobile communication technologies is rapid. People can interact with computers to access information and mail with friends at anytime and anywhere. Many computer applications are also developed to assist office operations, business developments and scientific research. In order to achieve an efficient user service, having an intelligent and effective communication between human and computers becomes a key issue. Spoken language is one of the most natural and intuitive ways for human to communicate with computers. Users do not need to learn any complicated usage instructions. Furthermore, the use of spoken language allows users to interact with computers in an eye-busy or hand-busy environment.

Due to these advantages, the use of human-computer conversational systems has become more and more widespread in many applications. Figure 1.1 is a typical architecture of spoken dialog systems (SDSs) [42]. The main

components include a speech recognizer, a natural language understanding (NLU) module, a text-to-speech synthesizer and a dialogue manager. A network interface obtains the input data and passes the output data. An application backend contains the task-specific information for the NLU module and the dialog manager to use.

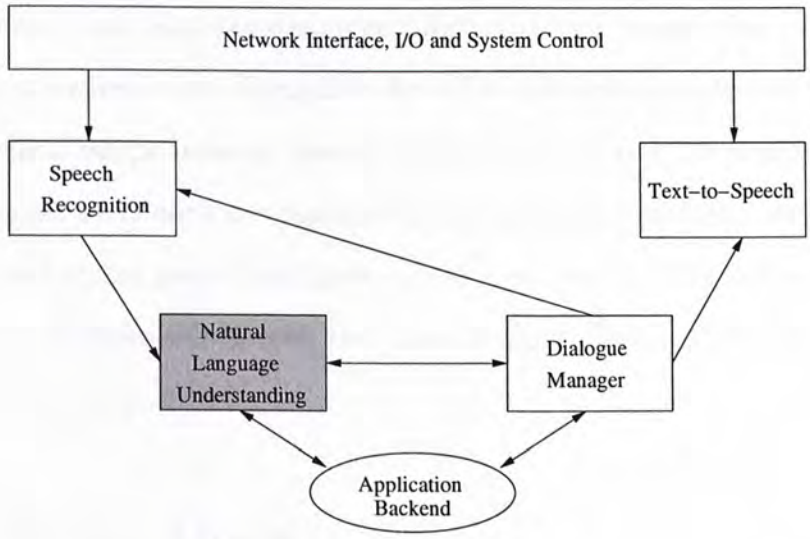

Figure 1.1: Architecture of spoken dialog systems, referenced from [42].

An NLU module plays an important role in SDSs. It receives a user's utterance from a speech recognizer and interprets the meaning. These systems often need to handle information-seeking queries from the user regarding a restricted domain. For example, an SDS may provide information about weather [46], traffic conditions [14] or air travel [41, 47]. Different users use different expressions to convey the same meaning. NLU in a domain-specific application requires identification of the user's communicative goal(s) out of a set of finite possibilities. Traditional approaches of NLU require grammar, which is created by domain experts, for parsing a user's utterance into seman-

tic concepts. Rules are applied to map the concepts to the communicative goal(s). However, grammatical coverage is a limitation. Manpower and time are also concerns.

Stochastic approaches were proposed to solve the above problems because they can automatically learn the semantic relationships from a large annotated training corpus. The use of Belief Networks (BNs) is a stochastic approach that incorporates uncertainty through probability theory and conditional dependence. Using BNs for NLU was first proposed in [31]. The causal relationships between the semantic concepts and the communicative goal of a user's sentence are captured in the network structure. We can identify the underlying goal of an input sentence by probabilistic inference. BNs can handle spontaneous speech and learn linguistic knowledge from training data automatically.

1.2 Thesis Goals

This thesis extends the investigation in the BN framework for NLU as proposed in [31]. A method was derived 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 BN. This formulation allows for the identification of queries with multiple goals, as well as queries with out-of-domain goals. However, the decisions are independent of each other. We noticed that a large number of sentences wrongly identified with multiple goals instead of a single goal. We aim to improve the goal identification accuracy by introducing interdependency among the goals. We will propose an alternative formulation that involves an one N-ary decision

using a *single* BN.

The NLU component in a human-computer conversational system should interpret a user's input quickly and avoid to keep the user waiting. Since we adopt a stochastic approach in our NLU framework, a large amount of computation is required for training and testing each BN. In the N binary formulation, one BN is built for each goal. We wish to minimize the amount of computation by reducing the number of BNs from N to one.

Since the BN framework automatically learns the linguistic knowledge from training data, it is portable to other languages. We aim to demonstrate the language portability of the BN in the one N-ary formulation by using a Cantonese Chinese corpus.

1.3 Thesis Outline

This thesis is organized as follows: Chapter 2 describes the background knowledge of the natural language understanding technology, our task domain and the Belief Network framework in the N binary formulation. Chapter 3 details the use of Belief Network for natural language understanding in the one N-ary formulation. Chapter 4 introduces the evaluation metrics and presents the comparative evaluation of the N binary and one N-ary formulations. Chapter 5 demonstrates that the Belief Network framework in the one N-ary formulation is portable to Chinese. Conclusions and future work are provided in Chapter 6.

Chapter 2

Background

This chapter presents the background knowledge relating to the natural language understanding, the related work of the Belief Network framework and our task domain. Natural language understanding is an important technology in human-computer conversational systems. The natural language understanding component is responsible for interpreting the meaning of the input text and returning a corresponding semantic representation. Various applications and approaches have been developed for it and will be introduced in Section 2.1. We will introduce the previous work on Belief Network framework for natural language understanding in Section 2.2 and our task domain, ATIS (Air Travel Information Service), in Section 2.3.

2.1 Natural Language Understanding

Natural Language Understanding (NLU) is a key technology in Spoken Dialog Systems (SDSs). It allows computers to communicate in a natural and intuitive way with users. These systems save the users' time and effort in

learning special usage instructions and thus reach the goal of universal usability. SDSs are often needed to handle information-seeking queries from the users regarding a restricted domain. An NLU component in a domain-specific application identifies the user's communicative goal(s) out of a set of finite possibilities characteristic of the domain. However, users can express a communicative goal in a variety of ways. Ambiguity of words or sentence structures, ellipsis, idioms and metaphor also make NLU difficult. Moreover, disfluencies (e.g. hesitations, false starts, repeated words and repairs) are common in spontaneous speech.

Different domain-specific SDSs are developed and relied on an NLU component to provide the meaning representation of a given query. Prominent examples include air travel information systems PEGASUS [47] and MERCURY [41], train information systems RAILTEL [4] and TABA [3], city guides MATCH [21] and VOYAGER [14], and automatic telephone switchboard and directory information system PADIS [24]. The languages concerned include English and multiple European languages. Some systems were developed for Chinese, such as the foreign exchange inquiry system CUFOREX [32]. There are also commercial organizations, like Nuance [12] and SpeechWorks [18], which provide speech-activated solutions for different industries, such as banking and travel planning.

Different approaches for NLU have been proposed. Each approach has its own advantages and disadvantages which make it to be adoptable in different conditions. Rule-based approach is a traditional NLU approach. The rules are strict in characterizing the users' speech. Other approaches to NLU are data-oriented, such as the phrase-spotting and stochastic approaches. Differ-

ent approaches can be mixed to model a NLU problem, in order to take the advantage of the relative strengths of each approach. In this thesis, we adopt a stochastic approach for NLU because it provides a best guess of uncertainty and offers the robustness. We choose a Belief Network model because it captures the causal relationship between the communicative goal(s) and the concepts[1] in a user's sentence. Furthermore, a BN model gives a concise specification of joint probability distribution. The network is tractable during reasoning. The details will be in the Section 2.2.

2.1.1 Rule-based Approaches

The major work of the rule-based approaches is hand-designing the grammar rules, which define the semantic and syntactic structures allowable in the task. Context-free grammars (CFGs) are widely used because the formalism is powerful enough to describe most of the structures in natural language. Each rule consists of a non-terminal on the left and a sequence of terminals and/or non-terminals on the right. The development of effective grammar rules usually requires linguistic experts to design the syntactic and semantic patterns of the users' input in a domain. After that, a parser applies the grammar rules to analyze the syntactic and semantic patterns of a user's sentence. The NLU component of the MASK [26] system is an example of the rule-based approach. During its development phase, the major work was to define the concepts that are meaningful for the railway travel information task and their appropriate keywords.

A critical factor in the rule-based approaches is grammar coverage. If a

[1] A concept is the smallest unit of meaning that is relevant to a specific task [27].

user says something that has not been defined in the grammar, the sentence cannot be interpreted. Rule-based models are usually applied to domain-specific applications. When we change the application's domain or extend the application scenarios, rules often have to be revised or rewritten [27]. Extensive amount of manpower is required to create, enhance and maintain the grammar rules [38].

2.1.2 Phrase-spotting Approaches

Phrase-spotting approaches are data-oriented. Some special syntactic or semantic phrases are frequently observed in the training data. The key phrases of interest have salient semantics. Similarity measures, like the Kullback-Leibler distance [2] and Mutual Information [11], are used for extracting candidate phrases automatically from the training corpus. A parser (phrase spotter) adopts a progressive search strategy to capture the key-phrases in a sentence, which are then analyzed and associated with a semantic representation for further interpretation [5]. The call routing system AT&T "*How May I Help you?*" [37] applied a phrase-spotting approach, in which grammar fragments have semantic associations with different call-types. A telephone dialog system for accessing e-mails [45] also applied the phrase-spotting technique.

Phrase-spotting approaches require a training set for capturing the specific phrases during the system development and do not have the capability in handling unseen data. The phrase-spotting technique is useful for dealing with ill-formed structures, such as hesitations, fillers, and out-of-vocabulary words [23]. However, it is hard to describe all possible keywords. Systems

using a phrase-spotting approach only work well for small applications with limited task complexities.

2.1.3 Stochastic Approaches

The problems of reusability and portability in grammar rules and the difficulty in pre-defining all possible keywords for phrase-spotting motivate the investigating of the stochastic approaches. Stochastic approaches (also known as statistical or probabilistic approaches) can automatically learn the relationships between the semantic concepts and their corresponding words of expression from a large annotated training corpus. It is data-oriented and hence more portable across domains and languages. The linguistic knowledge is captured in terms of statistical parameters, which are used to find the most likely concept sequence of a given string during the testing phase. A stochastic approach is flexible and robust because it can handle spontaneous speech. However, the performance of a model depends on the volume and sparseness of corpus that we used in training. Manual annotation and data collection are time-consuming and costly procedures. For example, the ATIS corpus (for which the details will be introduced in Section 2.3) took over a year in creation [25]. A domain-specific corpus is usually used to train the parameters for a specific task domain. Human experts are required to provide subjective probabilities when there is an insufficient or sparse training data [6]. Common stochastic models are probabilistic context-free grammars, connectionist models and Hidden Markov Models.

2.1.3.1 Probabilistic Context-Free Grammars (PCFGs)

Probabilistic context-free grammars (PCFGs) extend context-free grammars (CFGs) with probabilities. The assignment of a probability to each rule is based on the frequency of the rule applied to the training corpus. The most suitable parse tree is selected by maximizing aposterior probabilities of the trees. However, considerable amount of search time may be required. Some algorithms, like the N-best parsing algorithm, only explore the N most promising parse trees instead of all possible hypotheses. Efficiency can be highly increased but accuracy may be partially sacrificed. PCFGs solve the problem of grammar ambiguity in CFGs. Example applications include the restaurant guide, the Berkeley Restaurant Project (BeRP) [22], and a boat traffic information system, WAXHOLM [7]. TINA [40], a natural language system developed in MIT, also uses PCFGs for sentences parsing.

2.1.3.2 Connectionist Models

Connectionist models are artificial neural networks (ANN) which consist of layers of interconnected processing units [16, 19]. These units operate in parallel with weighted connections in order to store linguistic knowledge. The weights are learned from training data. The BASURDE Spanish dialogue system [8, 39] is an example of the use of multilayer perceptrons (a type of ANN) for natural language understanding. The system is applied to the railway information inquiry task with a fixed-size lexicon as input units. Each output unit corresponds to a dialogue act label which represents the intention of a user utterance in a restricted domain. However, the complex architecture in a connectionist model makes the representation and computation difficult.

When the size of a neural network is huge, the training is too slow to be tolerable [43]. Therefore, it is not a popular stochastic technique.

2.1.3.3 Hidden Markov Models (HMMs)

The Hidden Markov Model (HMM) is a popular stochastic model for semantic decoding. Some research prototype systems are modelled with this technique, such as AT&T-CHRONUS [34] and LIMSI-CNRS [33] for the ATIS task in English and French respectively. An HMM consists of sets of states, observations and acceptable transitions among states [28]. During training, the statistic parameters are estimated from the words in an input query (observations) and the corresponding semantic concepts (hidden states). In a testing phase, the most likely word string W and concept string C are decoded for a given acoustic string A according to:

$$P(\widetilde{W}, \widetilde{C}|A) = \max_{W \times C} P(W, C|A) \tag{2.1}$$

2.2 Belief Network Framework – the *N* Binary Formulation

2.2.1 Introduction of Belief Network

A Belief Network[2] (BN) is a probability reasoning tool [10, 20]. BN is an expressive graphical representation of causal relationships among the parameters in a domain. It combines the prior knowledge with the current observation. The notion of conditional independence in a BN simplifies knowledge

[2] Also known as Bayesian network, probabilistic network, causal network, causal graph or knowledge map.

acquisition and computation in reasoning [1]. A BN is a directed acyclic graph (DAG), where the nodes are the random variables and the arrows are the causal links (as shown in Figure 2.1). Every arrow points from cause (parent) to effect (child). A child node can also be a parent node, such as node B in Figure 2.1. Each variable represents an event with a finite set of mutually exclusive states. There is a conditional probability table $P(X|Y_1, Y_2, \ldots, Y_n)$ for each variable X with parents $Y_1, Y_2, \ldots, Y_n$. The conditional probability table shows the conditional probabilities of X being in a particular state given the states of its parents. In the case of a root node (without parents), its conditional probability table only gives a prior probability $P(X)$. For the example in Figure 2.1, node E has a conditional probability table $P(E|B, C)$, while the table of node A is reduced to $P(A)$. The BN structure, conditional and prior probabilities should be specified at the development stage. After that, when evidence / observation comes in, the BN performs belief updating by changing the conditional probabilities of the nodes.

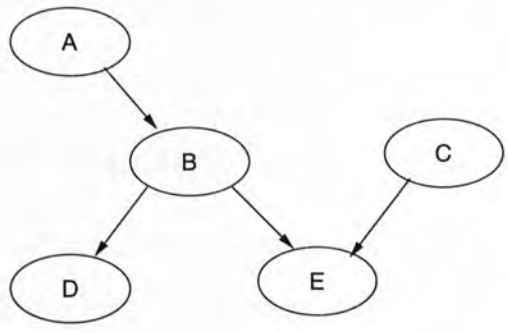

Figure 2.1: An simple example of Belief Network.

The use of Belief Networks in natural language understanding has been studied in [31]. The problem was formulated as making N binary decisions. The details will be introduced in the following subsections.

2.2.2 The N Binary Formulation

A method was derived for identifying the user's communicative goals out of a finite set of domain-specific goals (N) using Belief Networks [31]. It formulated the goal identification problem in term of making N binary decisions, each performed by a BN. The work was based on the ATIS domain, in which 11 goals were chosen as in-domain ($N = 11$). The details will be introduced in Section 2.3.

The objective of this method is to classify queries as single goal, multiple goals or out-of-domain (OOD) goal. The first step is to parse an input query into a sequence of semantic concepts, which is the input to the BNs. A BN applies Bayesian inference and outputs an aposterior probability for the query to represent the likelihood of the corresponding goal. Then, each BN makes a binary decision regarding the presence or absence of its goal by comparing the aposterior probability against a tuned threshold. The decisions are independent of each other. A query is rejected as OOD if all BNs vote negative.

2.2.3 Semantic Tagging

Semantic tagging is a process to parse an input query into a sequence of semantic concepts using hand-designed grammar rules. The sequence of semantic concepts form an input to the BNs for further goal inference. There are 60 semantic concepts defined for the ATIS domain, based on the attribute labels in the SQL expressions associated with the ATIS queries. The grammar rules are listed in Appendix C. Example in Table 2.1 shows an ATIS query with its parsed semantic tags and the annotated goal. Spontaneous

Query:	*"what flights are available from denver to baltimore first class on united airlines arriving may seventh before noon"*
Semantic tags:	<WHAT> <FLIGHT> <CHUNK> <FROM> <CITY_ORIGIN> <TO> <CITY_DESTINATION> <CLASS_NAME> <PREPOSITION> <AIRLINE_NAME> <TO> <MONTH> <DAY> <PRE_TIME> <PERIOD>
Goal:	flight.flight_id

Table 2.1: An ATIS query with its corresponding semantic tags and communicative goal.

speech effect, ill-formed and irrelevant expression are tagged into <CHUNK> and finally ignored in goal inference.

2.2.4 Belief Networks Development

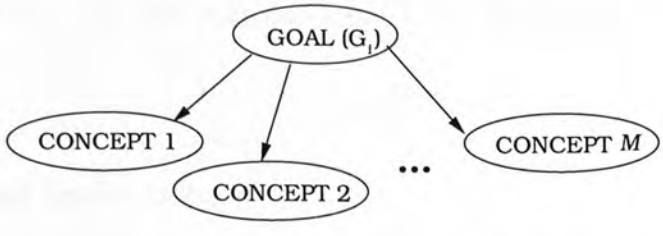

Figure 2.2: The naive Bayes' structure of a BN. The goal node outputs a binary state to indicate the presence or absence of the corresponding goal in a given query.

A BN in naive Bayes' topology (as shown in Figure 2.2) is used for the communicative goal(s) identification. The arrows are drawn from cause to effect. This structure captures the causal relationships between the com-

municative goal and the relevant semantic concepts in a query. The BN structure assumes that the concepts are independent of one another. Each concept has a binary state to indicate its presence or absence, based on the observation in a query. The goal node also has a binary state to show the presence or absence of the corresponding goal in a given query.

A BN is developed for each communicative goal from the training data. Each BN has M semantic concepts that is the most indicative to the corresponding goal. The dependency between a goal and a concept is measured by Information Gain (IG). For a given goal G_i $(i = 1, 2 \ldots N)$, we selected M concepts $\{C_1, C_2 \ldots C_M\}$ that have the highest IG in relation with G_i (Equation 2.2).

$$IG(C_k, G_i) = \sum_{c=0,1} \sum_{g=0,1} P(C_k = c, G_i = g) \log \frac{P(C_k = c, G_i = g)}{P(C_k = c)P(G_i = g)} \quad (2.2)$$

Each variable in a BN has a conditional probability table, i.e. $P(G_i)$ and $P(C_k|G_i)$. At the development stage, the statistical values are obtained by tallying the counts from the training data. They will be used for the Bayesian inference.

2.2.5 Goal Inference

After the development of N BNs, we parse an input query into a sequence of concepts by semantic tagging. According to the occurrences of the concepts, each BN applies Bayesian inference (Equation 2.3) and outputs an aposterior probability $P(G_i = 1|\vec{C})$ which is a confidence level of the goal G_i present in $\vec{C}$.

The aposterior probability is then compared with a threshold (denoted as θ_{f_i} for $i = 1, 2 \ldots N$) in order to make the binary decision. The thresholds

are tuned with the training data by optimizing the F-measure (Equation 2.4) 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 2.4 adopts $\beta = 1$ which treats precision and recall with equal importance.

$$P(G_i = 1|\vec{C}) = \frac{P(G_i = 1)\prod_{k=1}^{M} P(C_k = c_k|G_i = 1)}{\sum_{g=0,1}[P(G_i = g)\prod_{k=1}^{M} P(C_k = c_k|G_i = g)]} \tag{2.3}$$

$$F = \frac{(1+\beta^2)RP}{\beta R + P} \tag{2.4}$$

The binary decisions across the N BNs are united to identify the communicative goal(s) of a query. If all BNs vote negative, the framework treats the input query as OOD. If only a single BN votes positive for its corresponding goal, the framework labels the input query with the goal. If multiple BNs vote positive for their corresponding goals, the query is labeled with multiple goals.

2.2.6 Potential Problems

This approach formulated the goal identification problem as making N binary decisions. The decisions are independent of one another. We noticed a large number of sentences wrongly identified with multiple goals instead of a single goal. Furthermore, the computation in training and testing increases with the number of BNs. Hence, we have investigated the use of an alterna-

tive formulation in terms of one N-ary decision which will be introduced in Chapter 3.

2.3 The ATIS Domain

We have chosen to work in the ATIS (Air Travel Information Service) domain [17, 36]. ATIS is a common research domain, for which corpora were collected under the sponsorship of the ARPA (Advanced Research Projects Agency) spoken language systems technology development program. The Multi-Site ATIS Data Collection Working (MADCOW) group monitored the collection of data at five sites in the United States. The ATIS database is based on data obtained from the Official Airline Guide (OAG), which is organized under a relational schema. It contains information about flights, fares, airlines, airports, ground transportation and numerous others for 46 cities and 52 airports in the United States and Canada.

	Training	**1993 Test**	**1994 Test**
# Transcribed Queries	1564	448	444

Table 2.2: Distribution of the ATIS-3 Class A sentences.

We conducted our experiments on ATIS-3 Class A sentences, which are context-independent and hence can be understood unambiguously without dialog context. There are 1564, 448 (1993 test) and 444 (1994 test) transcribed queries in the disjoint training and test sets respectively (see Table 2.2). The corpora include a SQL expression for each query that can retrieve the reference answer from the OAG database. An example of a Class A query

with the corresponding simplified SQL and communicative goal is shown in Table 2.3.

Query: "*show me all flights from new york to milwaukee on northwest airlines departing at seven twenty a m*"
Simplified SQL: SELECT flight_id FROM flight WHERE airline_name = "*northwest airlines*" AND origin = "*new york*" AND destination = "*milwaukee*" AND departure_time = "*seven twenty am*"
Communicative Goal: flight.flight_id

Table 2.3: An ATIS-3 Class A sentence with the corresponding SQL query and communicative goal.

The main attribute labels of the SQL queries indicate the interested communicative goals. There are 32 communicative goals derived from the training set for the ATIS domain [31]. For example, the communicative goal of the SQL query in the Table 2.3 is flight.flight_id (flight identification). Among these 32 goals, 11 goals cover over 95% of the training set, 93% of the 1993 test set, and 92% of the 1994 test set. Hence, we only focus on the investigation of this set of 11 goals. The remaining goals are treated as out-of-domain (OOD). The communicative goals in the ATIS domain are listed in Appendix A. The distribution of the communicative goals in the training and test sets

are shown in Appendix B. Furthermore, we found 36 training queries with more than one communicative goals. We can classify the ATIS queries into three types: single goal, multiple goal and OOD. Examples are shown in Table 2.4.

Single goal example

Query:	*"flights on friday from newark to tampa"*
Goal:	flight.flight_id

Multiple goal example

Query:	*"give me the least expensive first class round trip ticket on u s air from cleveland to miami"*
Goals:	flight.flight_id, fare.fare_id

Out-of-domain (OOD) example

Query:	*"how many first class flights does united have leaving from all cities today"*
Goal:	count_flight (OOD, count_flight is not selected as in-domain)

Table 2.4: Examples of single goal, multiple goal and OOD queries in the ATIS domain.

2.4 Chapter Summary

In this chapter, we have covered the background information of this thesis. We presented the common approaches on NLU. After that, we described the previous approach of using BNs on NLU. The problem was formulated as making N binary decisions. We also introduced the ATIS domain, which

is our research domain. In this thesis, we adopt the BN framework due to its flexibility and robustness. We would like to improve the use of the BN framework for NLU.

Chapter 3

Belief Network Framework – the One N-ary Formulation

In this chapter, we propose an alternative formulation for natural language understanding (NLU) using the Belief Network (BN) framework. We formulate the communicative goal identification problem as making one N-ary decision. This extends the previous work and resolves the potential problems of independent decisions and massive computation in the N binary formulation. We employ the same pre-defined BN topology. In the one N-ary formulation, the goal node contains N states, one for each goal class. Each goal class represents an in-domain or out-of-domain (OOD) goal. This formulation captures the interdependency among the communicative goals as $\sum_g P(G = g|\vec{C}) = 1$ for $g \in \{g_1, g_2 \ldots g_N\}$. We have two goal identification strategies: (i) multiple aposterior strategy and (ii) maximum aposterior strategy. Both can identify single goal, multiple goals and OOD goals with a *single* BN only.

3.1 The One N-ary Formulation

We identify the communicative goal(s) of a given query out of a finite set of domain-specific goals (N) by making an one N-ary decision in a *single* BN. The BN is our stochastic tool for learning the causal relationships between the goal and the semantic concepts from the annotated training data. We work on the ATIS domain which contains single goal, multiple goal and out-of-domain (OOD) queries. We design two goal identification strategies, which extend the capability of identifying multiple goals in the single N-ary decision approach. The numbers of states (N) in the goal variable are different in these strategies:

(1) Multiple selection strategy concentrates on the 11 selected in-domain goals and add an extra goal for OOD queries ($N = 12$). The single BN makes a N-ary decision regarding the occurrence of each goal by comparing the aposterior probabilities with a relative threshold. Multiple goals are classified when there are more than one aposterior probabilities above the threshold.

(2) Maximum selection strategy selects the goal with the highest aposterior probability in the BN. We define a new class for each possible combination of the *in-domain multiple goals*. There are varied multiple goal combinations with mixed in-domain goal and OOD goal in the training and test sets. In order to prevent sparse data problems, we only extend new classes for the in-domain multiple goals. After examining the training data, we extend four classes of goals in total (see Table 3.1). They also cover all in-domain multiple goal combinations in the

test sets. Together with the 11 selected goals and the OOD goal, the goal variable has 16 states ($N = 16$) under this strategy. Multiple goals are classified when a corresponding goal class achieves the maximum aposterior probability.

The general steps in the NLU framework are similar to the N binary formulation. However, the calculation of each process in the one N-ary formulation is different as the representation of the goal node is changed, from binary states to N states. To identify the appropriate goal(s) of a given query, we first apply semantic tagging to parse the query into a sequence of semantic concepts. These concepts form the input to our single BN and initiate the BN probabilistic inference. According to the aposterior probabilities and our goal identification strategies, we assign the goal(s) to the input query.

3.2 Belief Network Development

We adopt a pre-defined BN topology with a naive Bayes' structure (see Figure 3.1), which is the same as we used in the N binary formulation. Each concept has a binary state (presence or absence) based on its occurrence in a query. The goal node has N states to represent the occurrence of the N goals, instead of the absence or presence of a particular goal in the N binary formulation. The single BN directly outputs the inferred goal(s) of a given query and captures the interdependency among the communicative goals.

We develop a *single* BN for *all* communicative goals using our training data. We use Information Gain (IG) to measure the dependency between

each concept and the presence of *all* N goals ($g \in \{g_1, g_2 \dots g_N\}$). In comparison, the N binary formulation concerns the absence or presence of a goal ($g = 0, 1$), see Equation 2.2. We select M concepts $\{C_1, C_2 \dots C_M\}$ that have the highest IG (Equation 3.1) as an input to the BN. The number of input concepts (M) is selected by optimizing with the overall goal identification performance.

$$IG(C_k, G) = \sum_{c=0,1} \sum_{g \in \{g_1, g_2 \dots g_N\}} P(C_k = c, G = g) \log \frac{P(C_k = c, G = g)}{P(C_k = c)P(G = g)} \tag{3.1}$$

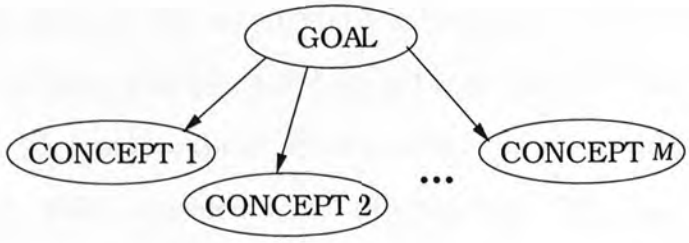

Figure 3.1: The Belief Network structure is the same as the one in the N binary formulation (Figure 2.2) but the goal node directly outputs the inferred goal(s) of a given query.

3.3 Goal Inference

Given a sequence of semantic concepts, we perform Bayesian inference (Equation 3.2). A set of aposterior probabilities, $P(G = g|\vec{C})$ where $g \in \{g_1, g_2 \dots g_N\}$, are produced together from a *single* BN and show the likelihood of each goal g_i present in a given query $\vec{C}$. In comparison, each BN in the N binary

formulation outputs an aposterior probability $P(G_i = 1|\vec{C})$ which is a confidence level of the goal G_i present in $\vec{C}$. The N-ary decision regarding the existence of each goal is made by a goal identification strategy. We have two goal identification strategies: multiple selection strategy and maximum selection strategy.

$$P(G = g_i|\vec{C}) = \frac{P(G = g_i)\prod_{k=1}^{M} P(C_k = c_k|G = g_i)}{\sum_{g\in\{g_1,g_2\ldots g_N\}}[P(G = g)\prod_{k=1}^{M} P(C_k = c_k|G = g)]} \quad (3.2)$$

3.3.1 Multiple Selection Strategy

Multiple selection strategy uses a relative threshold, $\theta \times \max P(G = g|\vec{C})$, to infer multiple outputs. The parameter θ is between 0 and 1. The θ is tuned based on the training data by optimizing with the multiple goal identification performance. We evaluate the multiple goal identification performance based on F-measure, which considers recall and precision. We adopt $\beta = 1$ in F-measure to combine recall and precision with equal importance. The input query is classified as the goal(s) $\hat{g}$ which has an aposterior probability above the relative threshold (Equation 3.3).

$$\hat{g} = \{g \in \{g_1, g_2 \ldots g_{12}\}|P(G = g|\vec{C}) \geq (\theta \times \max P(G = g|\vec{C}))\} \quad (3.3)$$

Figure 3.2 is a schematics which shows how the relative threshold captures multiple goals. The relative threshold is defined as a certain percentage ($\theta \in [0, 1]$) of the maximum aposterior probability ($\max P(G = g|\vec{C})$). Hence, it is flexible and changes according to the confidence level of the most likely goal in the given query. The θ controls the capability in identifying

multiple goals since it decides the coverage of the gray area in Figure 3.2. A BN votes positive for its goal if the output aposterior probability is higher than the threshold. There can be more than one goal. This strategy is explicit and it can identify unseen multiple goal combinations, which may exist in real situation. However, some queries will be wrongly identified with multiple goals due to this flexibility. In the example on Figure 3.2, the query contains g_1 and g_3. Under this strategy, even each goal class represents a single goal or an OOD goal, multiple goals can be inferred as well.

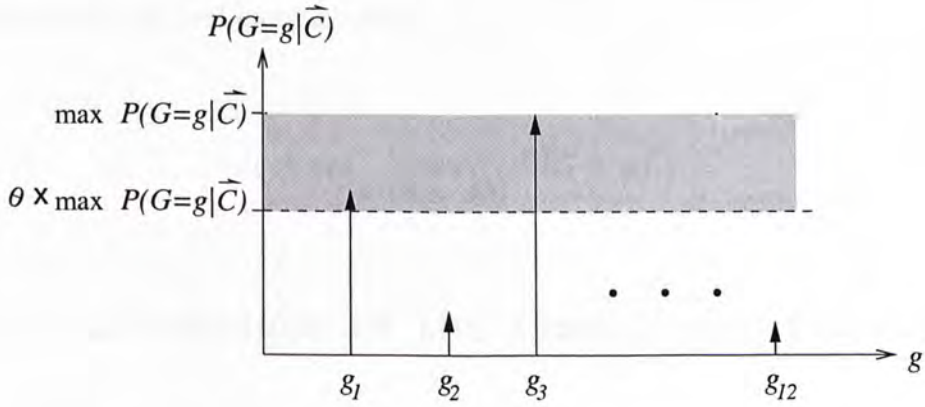

Figure 3.2: A schematics illustrates how the relative threshold ($\theta \times \max P(G = g|\vec{C})$) captures the multiple goals (g_1 and g_3).

3.3.2 Maximum Selection Strategy

Maximum selection strategy classifies a given query into the goal(s) $\hat{g}$ with the highest aposterior probability (Equation 3.4). Since we have multiple goal queries in the ATIS domain, we extend the goal classes by defining a new class for each possible combination of *in-domain multiple goals* . There

are varied multiple goal combinations with mixed in-domain goal and OOD goal in the training and test sets. We do not define new goal classes for such queries to prevent sparse data problems. We examined the *training set* and found that there are four combinations of goals. These four combinations also cover all the multiple goal cases in the *test sets*. Therefore, we extended four goal classes and hence $N = 16$. The combinations of goals are shown on Table 3.1 with example queries. We can identify multiple in-domain goals when $\hat{g}$ corresponds to an extended goal class. This strategy has extra knowledge / constraints to help multiple goal identification. However, it cannot identify unseen multiple goal combinations.

$$\hat{g} = \arg \max_{g \in \{g_1, g_2 \cdots g_{16}\}} P(G = g|\vec{C}) \tag{3.4}$$

3.4 Advantages of the One N-ary Formulation

The one N-ary formulation makes the goal identification decision in a single BN, where $\sum_g P(G = g|\vec{C}) = 1$ for $g \in \{g_1, g_2 \ldots g_N\}$. It captures the interdependency among the communicative goals. Hence, the existence of one goal affects that of other goals. It prevents a single goal query to be wrongly identified with multiple goals, which is common in the N binary formulation. This feature should improve the goal identification performance.

Since we adopt a stochastic approach in our NLU framework, we need to estimate the probabilities by tallying the counts from the training set. We apply Bayesian inference on the probabilities during parameters selection.

Multiple Goal Class 1:	
Query:	"*i need to find a plane from boston to san francisco on friday*"
Goals:	aircraft.aircraft_code, flight.flight_id
Multiple Goal Class 2:	
Query:	"*what's the airport at orlando*"
Goals:	airport.airport_code, airport.airport_name
Multiple Goal Class 3:	
Query:	"*explain the fare code q*"
Goals:	class_of_service.class_description, fare_basis.fare_basis_code
Multiple Goal Class 4:	
Query:	"*give me the least expensive first class round trip ticket on u s air from cleveland to miami*"
Goals:	flight.flight_id, fare.fare_id

Table 3.1: The four possible combinations of multiple goals and the corresponding example queries in the ATIS domain.

These processes require certain amount of additive and multiplicative operations. The amount of computation is in relation to the number of BNs, goals and concepts involved. We develop a single BN only in the one N-ary formulation. Hence the computation in the training stage can be highly reduced. These advantages will be proven in the next chapter with a number of experiments.

3.5 Chapter Summary

This chapter describes how to make one N-ary decision in a *single* Belief Network, in order to identify the communicative goal(s) of an information-seeking query. The naive BN structure captures the dependencies between the communicative goals and the semantic concepts. The semantic information is stored as statistical parameters, which are used for Bayesian inference. We propose two goal identification strategies: multiple selection strategy and maximum selection strategy. By using these goal identification strategies, the single BN can identify single goal, multiple goal as well as OOD queries. The one N-ary formulation has the capability of capturing the interdependency among communicative goals and such relationships should enhance the goal identification performance.

Chapter 4

Evaluation on the N Binary and the One N-ary Formulations

In the previous chapters, we have presented the use of Belief Network (BN) for natural language understanding (NLU) in the N binary and the one N-ary formulations. In this chapter, we conduct experiments using the ATIS corpora and compare the NLU performance between the two formulations. We have three goal identifiers in total: (i) a suite of BNs modeled under the N binary formulation, (ii) a single BN modeled under the one N-ary formulation with *multiple* selection strategy and (iii) a single BN modeled under the one N-ary formulation with *maximum* selection strategy. We introduce three evaluation methods – *accuracy measure*, *macro-averaging* and *micro-averaging* – for measuring the goal identification performance. Each evaluation method analyzes the goal identification performance from a differ-

ent angle. Our experiments compare the two formulations on the (i) overall goal identification performance, (ii) out-of-domain rejection, (iii) multiple goal identification and (iv) computation.

4.1 Evaluation Metrics

We have three different evaluation metrics for measuring the goal identification performance. The first one is the *accuracy measure* which is based on the number of errors in the inferred goals of each query. However, this measure overlooks the correctly identified goal(s). *Macro-* and *micro-averaging* are the evaluation techniques commonly used in text categorization [15, 44], which measure the category assignments in terms of recall and precision. Macro-averaging is a per-goal average which assigns equal weight to every goal, regardless of its frequency. Micro-averaging is a per-query average which gives an equal weight to every query. The two averaging techniques bias the results differently. Macro-averaging is influenced by the rare goals while micro-averaging is influenced by the most frequent goals. The details will be presented in subsections 4.1.2 and 4.1.3. We will use all these evaluation metrics in order to achieve a thorough understanding on each goal identifier's performance.

4.1.1 Accuracy Measure

The accuracy measure is an alignment measure in relation to the number of insertion (INS), deletion (DEL) and substitution (SUB) errors [29, 35]. Each sentence in the training and test sets associates with its reference goal(s). To score the goal identification performance, we align the hypothesized goal(s) with the reference goal(s) and identify the errors. The definitions of the errors and the example queries are shown in Table 4.1. The goal identification accuracy is computed in Equation 4.1. To obtain the accuracy, we tally the errors and the reference goals in the training or test sets. The accuracy is negative if the number of errors is larger than the number of reference goals.

$$accuracy = (1 - \frac{\#\text{INS} + \#\text{DEL} + \#\text{SUB}}{\#reference\_goals}) \times 100\% \tag{4.1}$$

4.1.2 Macro-Averaging

Macro-averaging evaluates the NLU goal identification performance as a categorization problem, which classifies a query with respect to a finite set of goals, $g \in \{g_1, g_2 \ldots g_N\}$. Each goal g is associated with a 2×2 contingency table as shown in Table 4.2 to denote the number of queries in each situation. Since our experiments are based on the ATIS domain, we have 12 per-goal contingency tables in total ($N = 12$), to represent the 11 in-domain goals and the OOD goal.

Deletion error (DEL)	
Definition:	There is a missing reference goal.
Query:	*"show me the cheapest first class round trip from new york to miami"*
Reference goals:	fare.fare_id, flight.flight_id
Inferred goal:	fare.fare_id (flight.flight_id is missing)
Insertion error (INS)	
Definition:	There is an additional inferred goal.
Query:	*"give me the fares for round trip flights from cleveland to miami next wednesday"*
Reference goal:	fare.fare_id
Inferred goals:	fare.fare_id, flight.flight_id (additional)
Substitution error (SUB)	
Definition:	There is an incorrect inferred goal.
Query:	*"i need the fares on flights from washington to toronto on a saturday"*
Reference goal:	fare.fare_id
Inferred goal:	flight.flight_id (incorrect)

Table 4.1: The definitions and examples of deletion, insertion and substitution errors.

	Reference = Yes	**Reference = No**
Inferred = Yes	a_g	b_g
Inferred = No	c_g	d_g

Table 4.2: A contingency table of a goal g, for $g \in \{g_1, g_2 \dots g_N\}$.

where

- a_g is the number of queries correctly inferred as goal g;
- b_g is the number of queries incorrectly inferred as goal g;
- c_g is the number of queries incorrectly rejected from goal g;
- d_g is the number of queries correctly rejected from goal g.

Macro-averaging computes recall (Equation 4.2) and precision (Equation 4.3) for *every* goal based on the corresponding per-goal contingency table. Then, we average the performance scores over the number of goals (Equation 4.4, 4.5). F-measure with $\beta = 1$ is used to combine the macro-recall and macro-precision into a single measure. The F-value is our final score for the performance. Since every goal has the same weight in the F-value regardless of its frequency, macro-averaging tends to over-emphasize the performance on the rare goals.

$$r(g) = \frac{a_g}{a_g + c_g} \tag{4.2}$$

$$p(g) = \frac{a_g}{a_g + b_g} \tag{4.3}$$

$$recall_{Macro} = \frac{\sum_{g \in \{g_1, g_2 \cdots g_N\}} r(g)}{N} \tag{4.4}$$

$$precision_{Macro} = \frac{\sum_{g \in \{g_1, g_2 \cdots g_N\}} p(g)}{N} \tag{4.5}$$

4.1.3 Micro-Averaging

Micro-averaging calculates only one value of recall and precision to evaluate the overall goal classification. A *global* contingency table is built by adding the corresponding cells in the per-goal contingency tables. The micro-recall (Equation 4.6) and micro-precision (Equation 4.7) are then computed over all decisions. Likewise, we adopt $\beta = 1$ in the F-measure to integrate recall and precision, and obtain a F-value. Since every individual query has an equal weight on the F-value, micro-averaging tends to over-emphasize the performance on the most frequent goals.

$$recall_{Micro} = \frac{\sum_{g \in \{g_1, g_2 \cdots g_N\}} a_g}{\sum_{g \in \{g_1, g_2 \cdots g_N\}} a_g + \sum_{g \in \{g_1, g_2 \cdots g_N\}} c_g} \tag{4.6}$$

$$precision_{Micro} = \frac{\sum_{g \in \{g_1, g_2 \cdots g_N\}} a_g}{\sum_{g \in \{g_1, g_2 \cdots g_N\}} a_g + \sum_{g \in \{g_1, g_2 \cdots g_N\}} b_g} \tag{4.7}$$

4.2 Experiments

Our experiments are conducted with the ATIS-3 Class A sentences in the training set, test set 1993 and test set 1994. We compare the goal identification performance among three goal identifiers: (i) a suite of BNs modeled

under the N binary formulation, (ii) a single BN modeled under the one N-ary formulation with *multiple* selection strategy and (iii) a single BN modeled under the one N-ary formulation with *maximum* selection strategy. Given a query mixed with in-domain and out-of-domain (OOD) goals, the single BN modeled under the one N-ary formulation with *multiple* selection strategy can identify both of them. However, the other two goal identifiers can only identify the in-domain goal. Therefore, we divide the multiple goal queries into two types:

- multiple in-domain goal,
- in-domain goal mixed with OOD

Including the case where only a single goal exists and the case of OOD goal, we have four types of query in all. The numbers of goals for each query type in test set 1993 and 1994 are shown in Table 4.3 and Table 4.4 respectively. The numbers on the fourth row are different because only the one N-ary formulation with multiple selection strategy can identify the in-domain and OOD goals together. In order to achieve a fair comparison among different goal identifiers, we do not use this type of queries in our experiments. We compare the goal identifiers in terms of (i) overall goal identification performance, (ii) out-of-domain rejection, (iii) multiple goal identification and (iv) computation. Before the goal identification process, we set the parameters for the Belief Network dimensions and the thresholds, which are described in subsections 4.2.1 and 4.2.2 respectively.

Formulation (strategy)	N **binary**	**One N-ary (multiple)**	**One N-ary (maximum)**
Single goal (# queries: 395)	395	395	395
Multiple in-domain goal (# queries: 8)	16	16	16
In-domain goal + OOD (# queries: 10)	10	20	10
OOD (# queries: 35)	35	35	35

Table 4.3: The number of goals for the four types of query in the *test set 1993*. The numbers on the fourth row are different because only the one N-ary formulation with multiple selection strategy can identify in-domain and OOD goals together.

Formulation (strategy)	N **binary**	**One N-ary (multiple)**	**One N-ary (maximum)**
Single goal (# queries: 399)	399	399	399
Multiple in-domain goal (# queries: 6)	12	12	12
In-domain goal + OOD (# queries: 2)	2	4	2
OOD (# queries: 37)	37	37	37

Table 4.4: The number of goals for the four types of query in the *test set 1994*. The numbers on the fourth row are different because only the one N-ary formulation with multiple selection strategy can identify in-domain and OOD goals together.

4.2.1 Network Dimensions

Our experiments determined the numbers of concept nodes (M) in the BNs based on the training data. We set a value of M for each goal identifier. We varied the number of input concepts from 10 to the full set of 60 concepts. We chose the value for M which gave the best goal identification performance or obtained less than 0.001 marginal improvement. We used micro-averaging to evaluate the goal identification performance, instead of the other two evaluation metrics, because of its simplicity in calculation.

For the N binary formulation, each BN has M concept nodes that map to the concepts with the highest values of Information Gain relating to the BN's goal. We tuned a single value of M for all BNs to keep the formulation simple. Figure 4.1 shows that an appropriate value to use for M is 50. For the one N-ary formulation, we defined two goal identification strategies and applied each of them to build a single BN. The results show that the F-value is optimal at 60 concepts (see Figure 4.2) using multiple selection strategy. Figure 4.3 shows the trend of the F-values becomes stable beyond 55 concepts using the maximum selection strategy, as the marginal improvement was less than 0.001. The single BN with maximum selection strategy has five concepts fewer than that with multiple selection strategy. These five concepts do not appear in the training set or only have few occurrences in the most frequent goals. The goal inference concerns the existence of each goal class and the single BN with maximum selection strategy contains more goals classes. Therefore, these concepts are less important in the goal identification using maximum selection strategy. Hence we developed the single BNs with $M = 60$ and $M = 55$ corresponding to the strategies. There are more

concepts involved in an one N-ary formulated BN because it integrates all the goals in a single BN. The selected concepts for each BN in both formulations are listed in Appendix D.

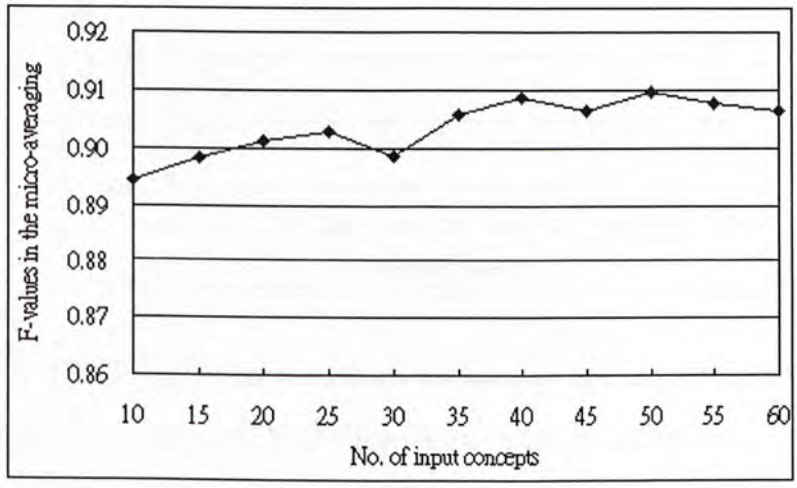

Figure 4.1: The F-values in the micro-averaging vary with the number of the input concepts in the N binary formulation. The graph suggests that we should use 50 concepts in each BN.

4.2.2 Thresholds

The N binary formulation and the one N-ary formulation with multiple selection strategy require thresholds for the goal identification. We selected the threshold values based on the training data. In the N binary formulation, each BN makes a binary decision regarding the absence or presence of the corresponding goal in a given query by comparing the aposterior probability with its threshold (θ_{f_i}). Therefore, we tuned 11 thresholds, one for each BN. The single BN makes one N-ary decision with multiple selection

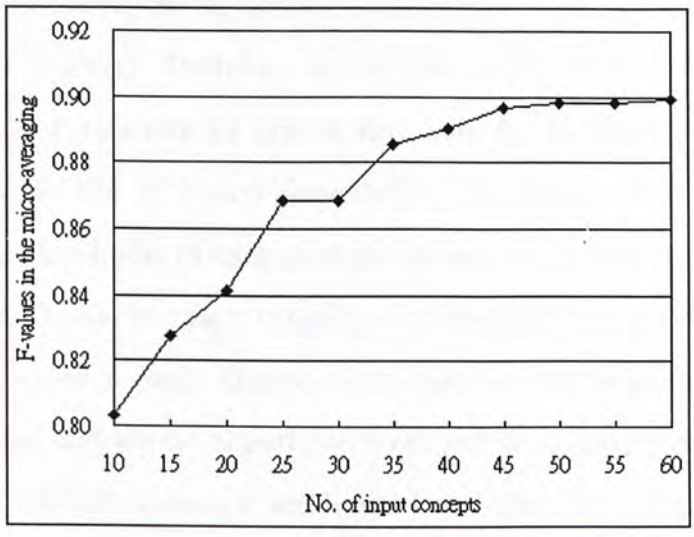

Figure 4.2: The F-values in the micro-averaging vary with different network dimensionalities in the one N-ary formulation using *multiple* selection strategy. The graph suggests that we should use $M = 60$ in the single BN.

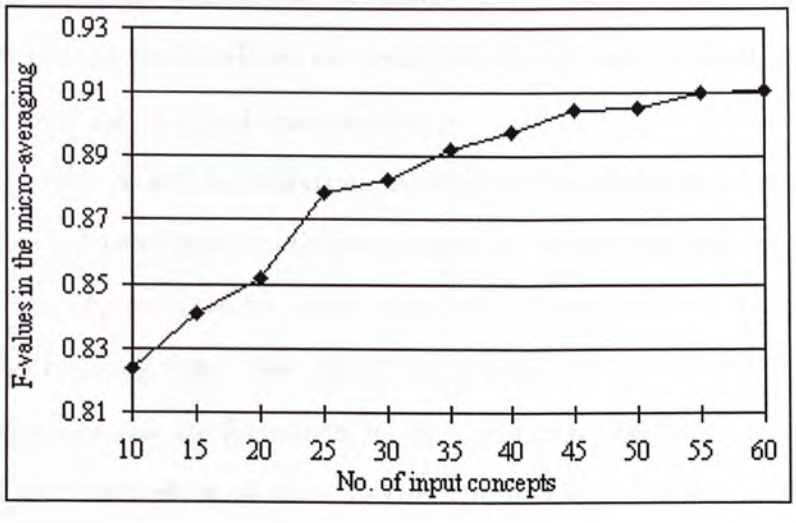

Figure 4.3: The F-values in the micro-averaging vary with different network dimensionalities in the one N-ary formulation using *maximum* selection strategy. The graph suggests that we should use $M = 55$ in the single BN.

strategy by comparing the aposterior probabilities with a relative threshold ($\theta \times \max P(G = g|\vec{C})$). Therefore, we set the value of θ.

We applied F-measure to tune a threshold θ_{f_i} for each BN representing a goal G_i in the N binary formulation, as mentioned in Section 2.2. The resulting thresholds of each goal are shown on Table 4.5 with example queries. The thresholds vary considerably due to the sentences structure of the corresponding goal. Queries with communicative goals such as airline.airline_name and airport.airport_name are generally simple and short sentences. The relevant concepts are limited and they have high conditional probabilities, $P(C_k = 1|G_i = 1)$, collected from the training set. As a result, the queries with these goals have high aposterior probabilities, $P(G_i = 1|\vec{C})$, and hence we use high thresholds for classification. On the contrary, long and complex sentences involve a wide range of concepts in different expressions. Therefore, the conditional probabilities are comparatively smaller and smaller aposterior probabilities are resulted. In this case, a smaller threshold should be used for the goal classification.

For the one N-ary formulation, a relative threshold ($\theta \times \max P(G = g|\vec{C})$) is needed to capture multiple goals when using the multiple selection strategy. Hence, we have to select an appropriate value for the parameter θ using the training data. We varied the θ from 0 to 1 and chose the value which optimizes the performance in the multiple goal identification. We evaluated the multiple goal identification performance based on F-measure (with $\beta = 1$), which combines recall and precision. The results are shown on Figure 4.4, which suggests that 0.3 is a suitable value.

We used the multiple goal identification performance for the parameter

Goal (Threshold):	aircraft.aircraft_code (0.78)
Example Query:	*"show me the aircraft that canadian airlines uses"*
Goal (Threshold):	airline.airline_code (0.59)
Example Query:	*"which airlines go from san francisco to washington by way of indianapolis"*
Goal (Threshold):	airline.airline_name (0.99)
Example Query:	*"what is h p"*
Goal (Threshold):	airport.airport_code (0.97)
Example Query:	*"what airport is at tampa"*
Goal (Threshold):	airport.airport_name (0.99)
Example Query:	*"what is y y z"*
Goal (Threshold):	class_of_service.class_description (0.99)
Example Query:	*"what does y mean"*
Goal (Threshold):	fare.fare_id (0.40)
Example Query:	*"how much does a first class round trip ticket from cleveland to miami on u s air cost"*
Goal (Threshold):	fare_basis.fare_basis_code (0.99)
Example Query:	*"what does fare code q oh mean"*
Goal (Threshold):	flight.flight_id (0.26)
Example Query:	*"show me the continental flights with meals which depart seattle on sunday for chicago"*
Goal (Threshold):	flight.flight_number (0.18)
Example Query:	*"what are the flight numbers of the flights which go from san francisco to washington via indianapolis"*
Goal (Threshold):	ground_service.city_code (0.99)
Example Query:	*"tell me about ground transportation at toronto"*

Table 4.5: A threshold is tuned for each BN representing a goal in the N binary formulation. An example query is listed with each goal to show the threshold value varies with the sentence structure.

selection, instead of the single goal identification performance, because there are few multiple goal queries (38 out of 1564 queries) in the training set. The F-values in the single goal identification tends to increase when the θ varies from 0 to 1 (i.e. the ability of capturing multiple goals decreases). It is because as the value of θ increases, there are less single goal queries wrongly identified with multiple goals and hence increases the single goal identification performance. However, the BN loses the ability in capturing multiple goals.

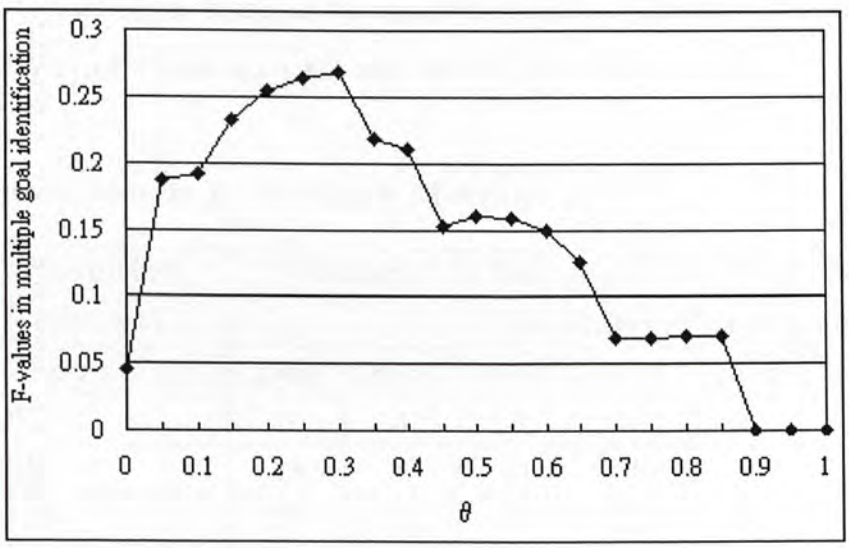

Figure 4.4: The F-values in multiple goal identification vary with the θ in the one N-ary formulation using multiple selection strategy. The graph suggests that $\theta = 0.3$ is a suitable value.

4.2.3 Overall Goal Identification

We measured the overall goal identification performance with the accuracy measure, macro- and micro-averaging. The problem is devised as categoriz-

ing a query into goal(s). We compared the performance of the N binary formulation and the one N-ary formulation with the multiple and maximum selection strategies using test set 1993 and test set 1994. In this part of work, we expanded our test sets by counting queries with multiple in-domain goals multiple times, which is the same as the number of the in-domain goals. For example, if a test query has two in-domain goals, we treat it as two single goal queries in the evaluations. The figures have been shown on Table 4.3 and Table 4.4 already. We do not evaluate the queries with mixed in-domain and OOD goals (row four), as we mentioned before. Therefore, we have 446 and 448 queries in test set 1993 and test set 1994 respectively.

Evaluation Metric 1: Accuracy Measure

Formulation (strategy)	**N binary**		**One N-ary (multiple)**		**One N-ary (maximum)**	
Test set	**1993**	**1994**	**1993**	**1994**	**1993**	**1994**
# DEL	3	4	3	4	3	2
# INS	43	27	41	26	8	6
# SUB	37	47	23	31	28	39
Total # errors	83	78	67	61	39	47
Goal identification accuracies	81.4% ($\frac{363}{446}$)	82.6% ($\frac{370}{448}$)	85.0% ($\frac{379}{446}$)	86.4% ($\frac{387}{448}$)	91.3% ($\frac{407}{446}$)	89.5% ($\frac{401}{448}$)

Table 4.6: Comparing the goal identification accuracies of the N binary formulation and the one N-ary formulation with the multiple and maximum selection strategies. The comparison is based on the numbers of deletion (DEL), insertion (INS) and substitution (SUB) errors produced in test set 1993 and 1994.

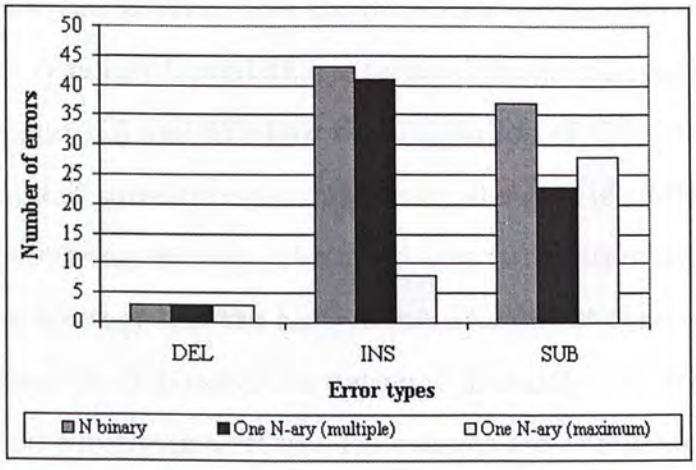

Figure 4.5: Comparing the numbers of deletion (DEL), insertion (INS) and substitution (SUB) errors among the goal identifiers in the N binary formulation and the one N-ary formulation with the multiple and maximum selection strategies using test set 1993.

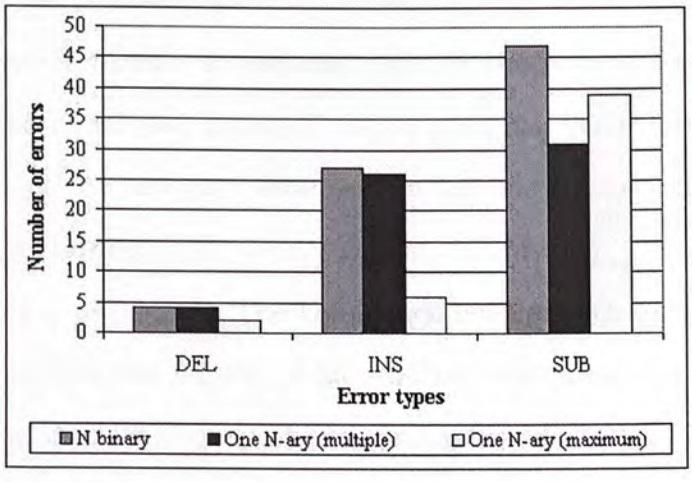

Figure 4.6: Comparing the numbers of deletion (DEL), insertion (INS) and substitution (SUB) errors among the goal identifiers in the N binary formulation and the one N-ary formulation with the multiple and maximum selection strategies using test set 1994.

The results in Table 4.6 show that the one N-ary formulation gave improvements over the N binary formulation in terms of the overall goal identification accuracies. Figure 4.5 and 4.6 show the comparison of the numbers of deletion, insertion and substitution errors among the goal identifiers using test set 1993 and 1994 respectively. The one N-ary formulation using the maximum selection strategy had the highest goal identification accuracies, up to 9.9% higher than the N binary formulation. It is mainly due to the reduction of insertion and substitution errors. The reasons are as follows:

1. Reduction of insertion errors.

   The number of insertion errors is reduced up to 35 and 21 when we migrated from the N binary formulation to the one N-ary formulation using test set 1993 and 1994 respectively. It is because the goal identifier in the N binary formulation wrongly identified many *single* goal queries with *multiple in-domain* goals. In the N binary formulation, a query can be labeled as one of the 11 goals and the decisions are independent of one another. However, in the one N-ary formulation, the goal probabilities $P(G = g|\vec{C})$ are dependent as $\sum_g P(G = g|\vec{C}) = 1$ for $g \in \{g_1, g_2 \ldots g_N\}$. The confidence level of each goal is compared among themselves for the most suitable classification(s). When the correct goal has a high aposterior probability, the other goals will have small probabilities in order to maintain the sum of all probabilities equal to one. Therefore, the interdependency among the goals prevents multiple in-domain goals identified for a single goal query.

   In the one N-ary formulation, the reduction of insertion errors using the *multiple* selection strategy is less than that using the *maximum*

selection strategy. It is because the relative threshold ($\theta \times \max P(G = g|\vec{C})$) in the *multiple* selection strategy is too low for some single goal queries and we have too few multiple goal queries in the training set (38 out of 1564) for tuning the value of θ. The interdependency among the goals in a single BN tends to increase the aposterior probability of one goal and lower the aposterior probabilities of other goals. Therefore, when we tune a θ in the *multiple* selection strategy, the θ has to be small (i.e. $\theta \times \max P(G = g|\vec{C})$ is low) in order to capture the multiple goals. However, using a small relative threshold, some single goal queries were wrongly identified with multiple goals. In comparison, the single BN with *maximum* selection strategy does not need a threshold because it has extra constraints (extended goal classes) to help multiple goal identification. However, the extended goal classes can increase the chance of confusion among a greater number of goal classes.

The effect of the interdependency among the goals is illustrated by an example in Table 4.7. The N binary formulation wrongly inferred the query with the goal flight.flight_id (G_9) because the sentence contains certain semantic tags, like <FLIGHT> and <CITY_NAME>, which are indicative of that goal. Hence, the aposterior probability of the goal flight.flight_id was increased to 0.435. As the confidence levels of the goals airline.airline_code (G_2) and flight.flight_id (G_9) are larger than the corresponding thresholds, both goals were inferred. Figure 4.7 shows the aposterior probabilities of each BN in the N binary formulation for the example in Table 4.7.

However, statistics in the one N-ary formulation captures the

Query:	*"i would like to have the airline that flies between toronto, detroit and saint louis"*
Semantic tags:	<DUMMY> <CHUNK> <DUMMY> <AIRLINE> <CHUNK> <FLIGHT> <PREP> <CITY_NAME> <CITY_NAME_1> <CONNECTIVE> <CITY_NAME_2>
Reference goal:	airline.airline_code
N binary formulation	
Inferred goals:	airline.airline_code (G_2) (✓) ($P(G_2 = 1\|\vec{C}) = 0.968 > \theta_{f_2} = 0.59$) flight.flight_id ($G_9$) (INS) ($P(G_9 = 1\|\vec{C}) = 0.435 > \theta_{f_9} = 0.26$)
One N-ary formulation (multiple)	
Inferred goal:	airline.airline_code (g_2) (✓) ($P(G = g_2\|\vec{C}) = 0.992$)
One N-ary formulation (maximum)	
Inferred goal:	airline.airline_code (g_2) (✓) ($P(G = g_2\|\vec{C}) = 0.992$)

Table 4.7: An example illustrating a single goal query wrongly identified as multiple in-domain goals in the N binary formulation. Hence, an insertion error (INS) was produced. The one N-ary formulation labeled the query with the correct goal using either multiple or maximum selection strategies.

fact that the goals airline.airline_code ($P(G = g_2|\vec{C}) = 0.992$) and flight.flight_id ($P(G = g_9|\vec{C}) = 0.008$) are seldom together, and the input concepts are more likely to appear in airline.airline_code. In Figure 4.8 and 4.9, the graphs show the aposterior probabilities in the one N-ary formulation using the multiple and maximum selection strategies respectively for the example queries in Table 4.7. The goals with probabilities lower than 10^{-3} are not shown on the graphs. In Figure 4.8, the goal airline.airline_code (g_2) obtained the highest aposterior probability at 0.992 and the relative threshold became 0.298. The interdependency of the goals prevents an insertion error in the one N-ary formulation using multiple selection strategy. In Figure 4.9, the goal airline.airline_code (g_2) got the maximum aposterior probability. According to the maximum selection strategy, airline.airline_code was the only goal inferred for the example query and hence we obtained the correct result.

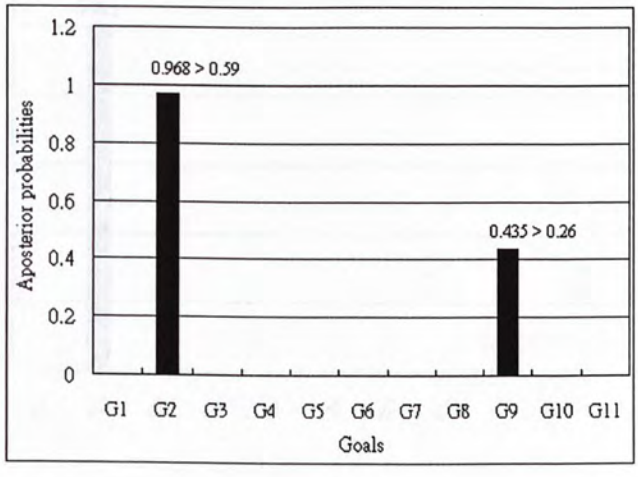

Figure 4.7: The graph shows the aposterior probabilities of each BN in the N binary formulation for the example in Table 4.7, except the goals with probabilities lower than 10^{-3}. Goals airline.airline_code (G_2) and flight.flight_id (G_9) voted positive as their probabilities is larger than the corresponding thresholds (labeled as $P(G_i = 1|\vec{C}) > \theta_{f_i}$ at the top of the bars) .

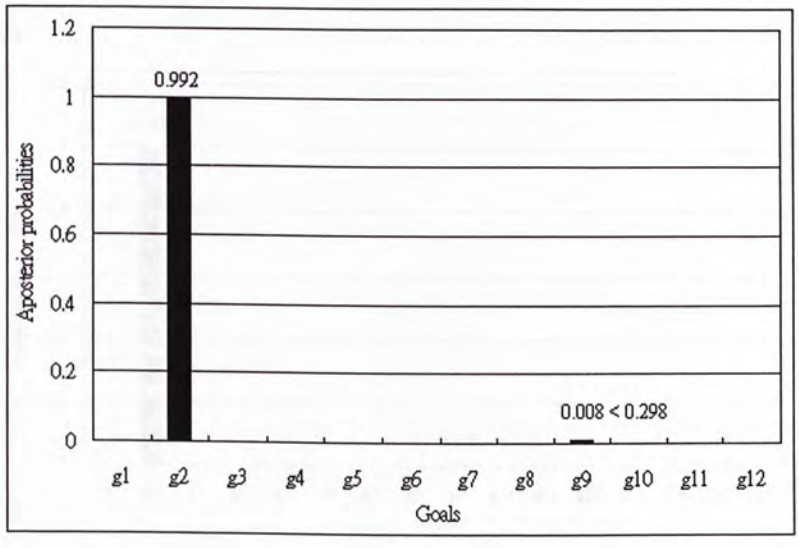

Figure 4.8: The graph shows the aposterior probabilities in the one N-ary formulation using the *multiple* selection strategy for the example in Table 4.7. The goals with probabilities lower than 10^{-3} are not shown on the graph. The interdependencies among the goals and the relative threshold ($\theta \times \max P(G = g|\vec{C}) = 0.298$) prevent an additional goal flight.flight_id (g_9) being inferred.

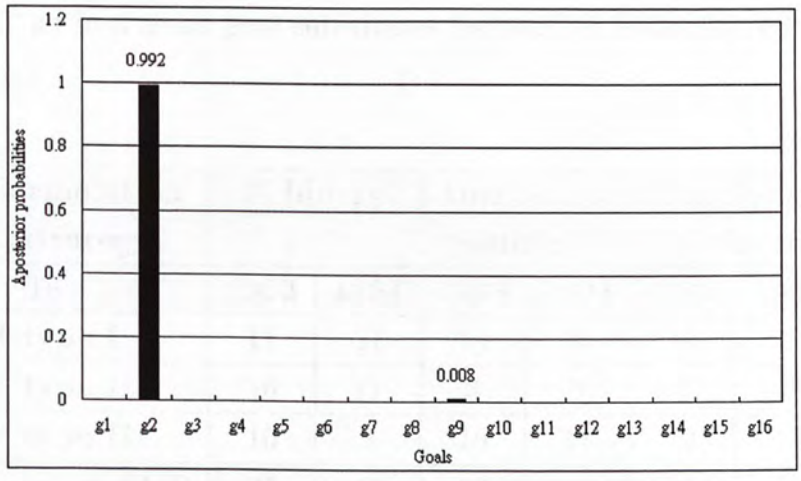

Figure 4.9: The graph shows the aposterior probabilities in the one N-ary formulation using the *maximum* selection strategy for the example in Table 4.7, except the goals with probabilities lower than 10^{-3}. The goal airline.airline_code (g_2) has the maximum probability and we labeled it as the query's goal.

2. Reduction of substitution errors.

   The number of substitution errors was high in the N binary formulation. We can divide the substitution errors into three types:

   (I) an in-domain goal substitutes for the OOD goal,

   (II) the OOD goal substitutes for an in-domain goal,

   (III) an in-domain goal substitutes for another in-domain goal.

Formulation (strategy)	N **binary**		**One N-ary (multiple)**		**One N-ary (maximum)**	
Test set	**1993**	**1994**	**1993**	**1994**	**1993**	**1994**
# type I	11	27	10	15	10	19
# type II	16	17	3	5	3	8
# type III	10	3	10	11	13	12
Total # SUB	37	47	23	31	28	39

Table 4.8: Distribution of the three types of substitution (SUB) errors – (I) an in-domain goal substitutes for the OOD goal, (II) the OOD goal substitutes for an in-domain goal and (III) an in-domain goal substitutes for another in-domain goal – in the N binary formulation and the one N-ary formulation using multiple and maximum selection strategies in test set 1993 and 1994.

Table 4.8 shows the distribution of the substitution errors in the N binary formulation and the one N-ary formulation using the multiple and maximum selection strategies in test set 1993 and 1994. We found that the N binary formulation generated more substitution errors related to the OOD goal (type I and II) than the one N-ary formulation

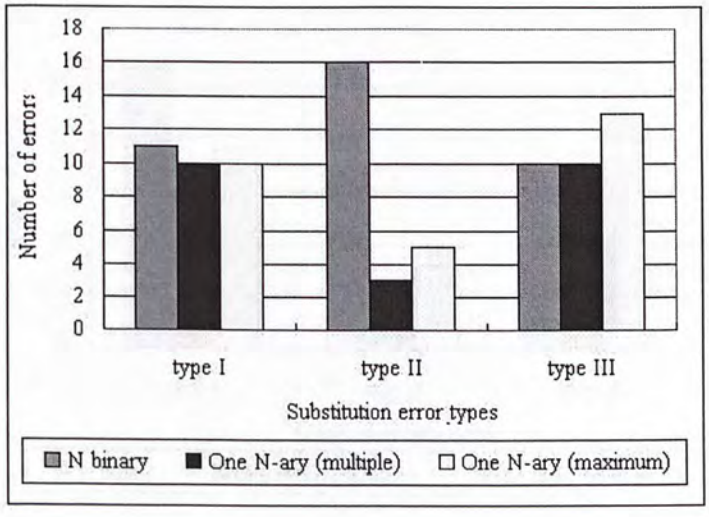

Figure 4.10: Comparing the numbers of substitution errors – (I) an in-domain goal substitutes for the OOD goal, (II) the OOD goal substitutes for an in-domain goal and (III) an in-domain goal substitutes for another in-domain goal – among the goal identifiers in the N binary formulation and the one N-ary formulation with the multiple and maximum selection strategies using test set 1993.

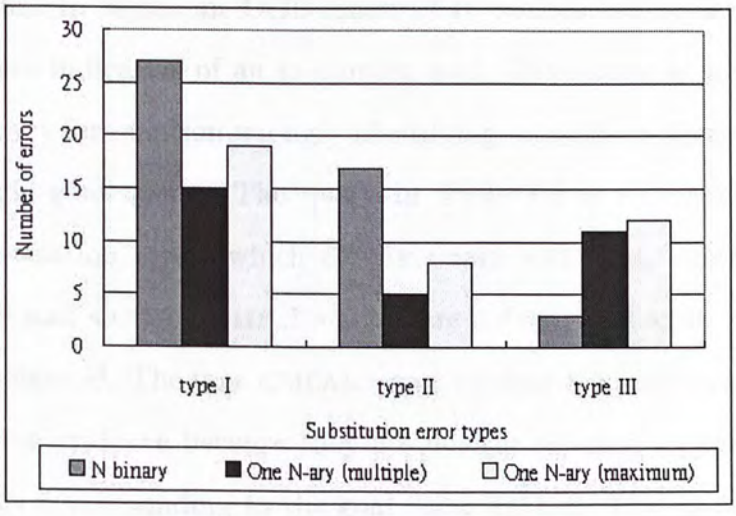

Figure 4.11: Comparing the numbers of substitution errors – (I) an in-domain goal substitutes for the OOD goal, (II) the OOD goal substitutes for an in-domain goal and (III) an in-domain goal substitutes for another in-domain goal – among the goal identifiers in the N binary formulation and the one N-ary formulation with the multiple and maximum selection strategies using test set 1994.

did. Figure 4.10 and 4.11 compare the numbers of the three types of substitution errors among the goal identifiers using test set 1993 and 1994 respectively.

Recall that the N binary formulation rejects a query as OOD when all BNs vote negative against it. We found that the suite of BNs fail to reject an OOD query if it contains some semantic tags that are indicative of an in-domain goal. The reason is similar to the N binary formulation wrongly identifying multiple in-domain goals for a single goal query. The query in Table 4.9 is an example of type I substitution error, which contains tags <FROM>, <CITY_NAME>, <TO> and <CITY_NAME_1>[1] that are indicative of an in-domain goal flight.flight_id. The tags <MEAL> and <AIRLINE_NAME> cannot be the negative evidence because they are not the selected input concepts in the BN corresponding to the goal flight.flight_id. The aposterior probability of the in-domain goal flight.flight_id was 0.33 and became larger than its corresponding threshold 0.26. The query was wrongly labeled with the in-domain goal and generated a substitution error using the N binary formulation. However, the one N-ary formulation trained a goal state to represent the OOD goal and built the interdependency among all goals. In the example, the OOD goal got the maximum aposterior probabilities at 0.71 in the one N-ary formulation using both goal identification strategies. It was directly labeled as the query's goal by the maximum selection strategy. As there was no other goal with a probability larger than the relative threshold 0.21 in the multiple selection

[1] The second city name in a query is tagged as <CITY_NAME_1>.

Query:	*"what meals are served on american flight eight eleven from tampa to milwaukee"*
Semantic tags:	<WHAT> <MEAL> <CHUNK> <SERVE> <PREP> <AIRLINE_NAME> <FLIGHT_NUMBER> <FROM> <CITY_NAME> <TO> <CITY_NAME_1>
Reference goal:	food_service.meal_code (OOD)
N binary formulation	
Inferred goal:	flight.flight_id (G_9) (SUB) $P(G_9 = 1\|\vec{C}) = 0.33 > \theta_{f_i} = 0.26$
One N-ary formulation (multiple)	
Inferred goal:	OOD (g_{12})(✓) $P(G = g_{12}\|\vec{C}) = 0.71$ (maximum) (no other aposterior probability is larger than the relative threshold, $0.71 \times 0.3 = 0.21$)
One N-ary formulation (maximum)	
Inferred goal:	OOD (g_{16}) (✓) $P(G = g_{16}\|\vec{C}) = 0.71$ (maximum)

Table 4.9: An OOD query wrongly labeled with an in-domain goal in the N binary formulation and generated a substitution (SUB) error. The one N-ary formulation rejected it successfully with the multiple and maximum selection strategies.

Query:	*"and now show me ground transportation that i could get in boston late night"*
Semantic tags:	<CONNECTIVE> <DAY_NAME> <DUMMY> <TRANSPORT> <CHUNK> <PREP> <CITY_NAME> <MODIFIER> <PERIOD>
Reference goal:	ground_service.city_code (G_{11})
N binary formulation	
Inferred goal:	OOD (SUB)
$P(G_{11} = 1\|\vec{C})$:	$0.83 < \theta_{f_{11}} = 0.99$
One N-ary formulation (multiple)	
Inferred goal:	ground_service.city_code (✓)
$P(G = g_{11}\|\vec{C})$:	0.99 (maximum)
One N-ary formulation (maximum)	
Inferred goal:	ground_service.city_code (✓)
$P(G = g_{11}\|\vec{C})$:	0.99 (maximum)

Table 4.10: An example query wrongly labeled with an OOD goal in the N binary formulation due to the high threshold of the goal ground_service.city_code. It generated a substitution (SUB) error. The one N-ary formulation correctly identified the in-domain goal with the multiple and maximum selection strategies.

strategy, the OOD goal was also correctly inferred.

Type II substitution error occurs when a single goal query is wrongly identified as OOD. The N binary formulation fails to identify an in-domain goal when the corresponding threshold is too high. The one N-ary formulation trained a goal state to represent the OOD goal. For the goal identification, one or none threshold is required when we use multiple and maximum selection strategy respectively. An example

in Table 4.10 shows that the threshold of the goal ground_service.city_code is too high for the query, and thus the N binary formulation failed in identifying the correct goal. The BN using the one N-ary formulation with multiple selection strategy could identify the correct goal because the goal ground_service.city_code got a high aposterior probability at 0.99 and it was impossible to have another goal inferred. The single BN using maximum selection strategy could also identify the correct goal because the goal ground_service.city_code got the maximum aposterior probability.

Type III substitution error occurs when an in-domain goal substitutes for another in-domain goal. We found that the number of type III substitution errors is larger in the one N-ary formulation using maximum selection strategy. This is because the other two goal identifiers had more insertion errors which covered some substitution errors. Table 4.11 is an example showing that the N binary formulation and the one N-ary formulation using the multiple selection strategy incorrectly inferred that a single goal query has multiple goals. As one of the multiple goals is the same as the reference goal, an insertion error was generated. However, the one N-ary formulation using the maximum selection strategy inferred a wrong single goal and hence a type III substitution error was generated.

Query:	*"give me the fares for round trip flights from cleveland to miami next wednesday"*
Semantic tags:	<CHUNK> <DUMMY> <FARE> <PREP> <ROUND_TRIP> <FLIGHT> <FROM> <CITY_NAME> <TO> <CITY_NAME_1> <MODIFIER> <DAY_NAME>
Reference goal:	fare.fare_id (G_7)
N binary formulation	
Inferred goals:	fare.fare_id (G_7) (✓) $P(G_7 = 1\|\vec{C}) = 0.796 > \theta_{f_7} = 0.40$ flight.flight_id (G_9) (INS) $P(G_9 = 1\|\vec{C}) = 0.999 > \theta_{f_9} = 0.26$
One N-ary formulation (multiple)	
Inferred goals:	fare.fare_id (g_7) (✓) $P(G = g_7\|\vec{C}) = 0.516$ (maximum)
	flight.flight_id (g_9) (INS) $P(G = g_9\|\vec{C}) = 0.484 > 0.3 \times 0.516 = 0.15$
One N-ary formulation (maximum)	
Inferred goal:	flight.flight_id (g_9) (SUB) $P(G = g_9\|\vec{C}) = 0.516$ (maximum)

Table 4.11: An example query shows that insertion errors can cover some substitution errors. The N binary formulation and the one N-ary formulation using the multiple selection strategy incorrectly inserted the goal flight.flight_id. The one N-ary formulation using the maximum selection strategy got a single incorrect goal only and generated a substitution (SUB) error.

Evaluation Metric 2: Macro-Averaging

Formulation (strategy)	N **binary**		**One N-ary (multiple)**		**One N-ary (maximum)**	
Test set	**1993**	**1994**	**1993**	**1994**	**1993**	**1994**
$recall_{Macro}$	0.89	0.67	0.92	0.84	0.90	0.75
$precision_{Macro}$	0.77	0.55	0.77	0.65	0.88	0.63
F-value	0.83	0.61	0.84	0.74	0.89	0.67

Table 4.12: Comparing the overall goal identification performance of N binary formulation and the one N-ary formulation with the multiple and maximum selection strategies using macro-averaging. The results show that the one N-ary formulation improved over the N binary formulation.

We compared the performance of the N binary formulation and the one N-ary formulation using the multiple and maximum selection strategies by macro-averaging. The recall and precision of each goal in the N binary formulation and the one N-ary formulation using test set 1993 and 1994 are listed in Appendix E. The overall macro-averaging results are tabulated in Table 4.12 which shows that the one N-ary formulation improved over the N binary formulation. Figure 4.12 and 4.13 compare the recalls, precisions and F-values among the goal identifiers using test set 1993 and 1994 respectively in macro-averaging. The improvement is up to 6% in test set 1993 and 13% in test set 1994.

The average of the goal classification performance is higher in the one N-ary formulation, regardless of the goal's frequency. It is because the inter-dependency among the goals is effective for selecting the most suitable goal(s) for a given query, as mentioned before. The large numbers of insertion and

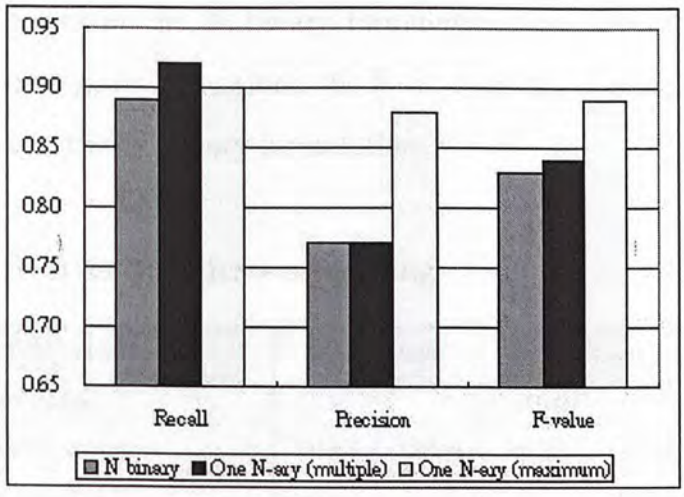

Figure 4.12: Comparing the recalls, precisions and F-values among the goal identifiers in the N binary formulation and the one N-ary formulation with the multiple and maximum selection strategies using test set 1993 in macro-averaging.

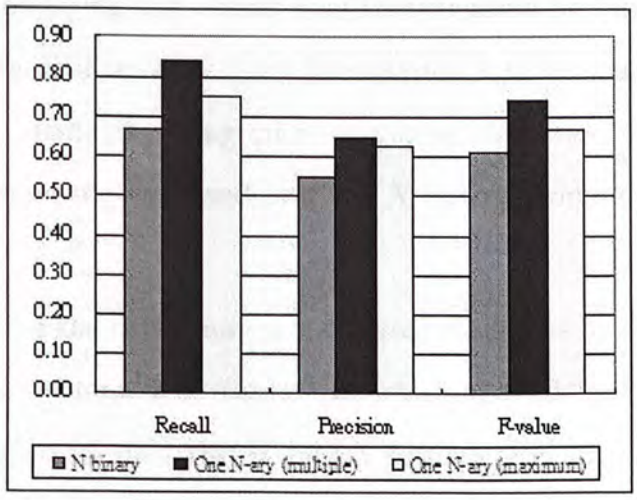

Figure 4.13: Comparing the recalls, precisions and F-values among the goal identifiers in the N binary formulation and the one N-ary formulation with the multiple and maximum selection strategies using test set 1994 in macro-averaging.

substitution errors in the N binary formulation also lower the recall and precision in some goals. Therefore, the lower $recall_{Macro}$ and $precision_{Macro}$ were obtained in the N binary formulation.

Evaluation Metric 3: Micro-Averaging

Formulation (strategy)	N **binary**		**One N-ary (multiple)**		**One N-ary (maximum)**	
Test set	**1993**	**1994**	**1993**	**1994**	**1993**	**1994**
# reference goals (A)	446	448	446	448	446	448
# inferred goals (B)	486	471	484	470	451	452
# correctly inferred goals (C)	406	397	420	413	415	407
$recall_{Micro}$ (C/A)	0.91	0.89	0.94	0.92	0.93	0.91
$precision_{Micro}$ (C/B)	0.84	0.84	0.87	0.88	0.92	0.90
F-value	0.87	0.86	0.90	0.90	0.93	0.90

Table 4.13: Comparing the overall goal identification performance of N binary formulation and the one N-ary formulation with the multiple and maximum selection strategies using micro-averaging. The results show that the one N-ary formulation improved over the N binary formulation.

We also compared the performance of the goal identifiers by micro-averaging. The results are tabulated in Table 4.13 which showed that the one N-ary formulation improved over the N binary formulation up to 6% in test set 1993 and 4% in test set 1994. The results are consistent with the accuracy measure and macro-averaging. Figure 4.14 and 4.15 compare the recalls, precisions and F-values among the goal identifiers using test set 1993 and 1994 respectively in micro-averaging. The improvement is also due to the reduction of insertion and substitution errors in one N-ary formulation. That

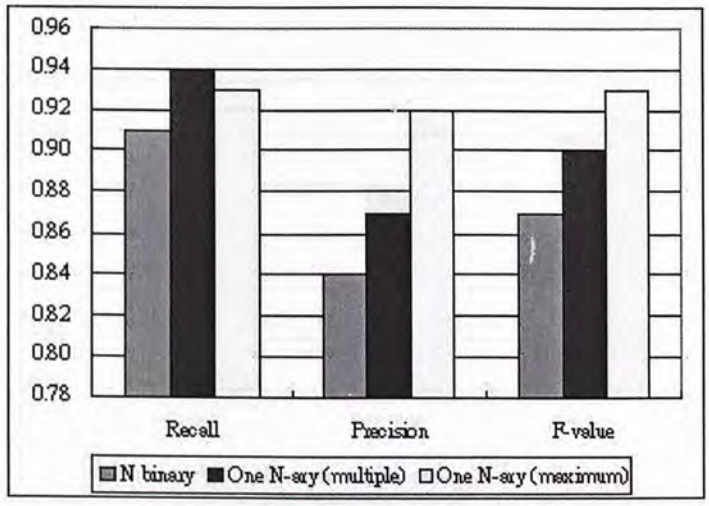

Figure 4.14: Comparing the recalls, precisions and F-values among the goal identifiers in the N binary formulation and the one N-ary formulation with the multiple and maximum selection strategies using test set 1993 in micro-averaging.

increases the numbers of correctly inferred goals (row five) and decreases the numbers of inferred goals (row four). Therefore, the $recall_{Micro}$ and $precision_{Micro}$ were higher in the one N-ary formulation and directly led to the higher F-values. The one N-ary formulation with maximum selection strategy had better performance than the multiple selection strategy because the imperfect relative threshold increased the insertion errors. We found that the F-values in micro-averaging were higher than those in the macro-averaging possibly because we selected the network dimensions (M) using micro-averaging. Moreover, the goal flight.flight_id have the largest numbers of queries in test set 1993 (301 out of 448) and test set 1994 (342 out of 444). The recalls and precisions of this goal were higher than 0.96 in both test sets. The good performance of this high frequency goal led to the higher F-values

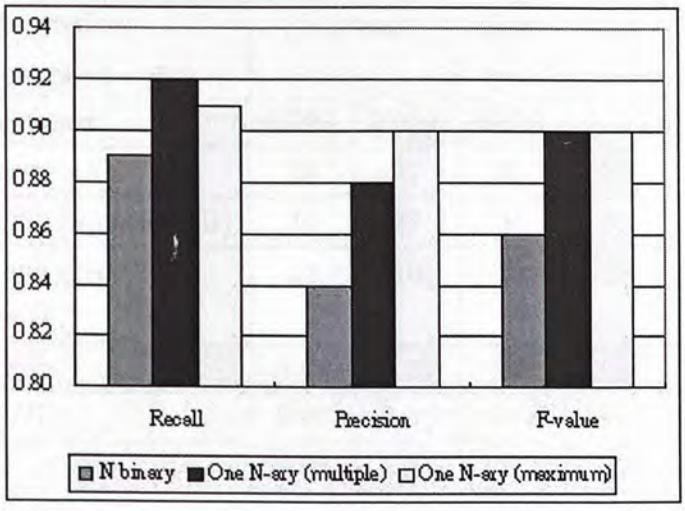

Figure 4.15: Comparing the recalls, precisions and F-values among the goal identifiers in the N binary formulation and the one N-ary formulation with the multiple and maximum selection strategies using test set 1994 in micro-averaging.

in micro-averaging.

4.2.4 Out-Of-Domain Rejection

We compared the formulations in terms of appropriate OOD rejection using the test sets. The results were analyzed in recall, precision and F-measure with $\beta = 1$. (see Table 4.14). Figure 4.16 and 4.17 compare the recalls, precisions and F-values among the goal identifiers using test set 1993 and 1994 respectively in OOD rejection. We can see that the one N-ary formulation improved over the N binary formulation up to 13% for test set 1993 and 34% for test set 1994. It is due to the reduction of substitution errors related to the OOD goal, as we mentioned in the previous subsection 4.2.3.

Formulation (strategy)	N **binary**		**One N-ary (multiple)**		**One N-ary (maximum)**	
Test set	**1993**	**1994**	**1993**	**1994**	**1993**	**1994**
# OOD queries (A)	35	37	35	37	35	37
# inferred OOD queries (B)	40	27	34	31	30	26
# correctly inferred OOD queries (C)	24	10	25	22	25	18
recall (C/A)	0.69	0.27	0.71	0.59	0.71	0.49
precision (C/B)	0.60	0.37	0.74	0.71	0.83	0.69
F-value	0.64	0.31	0.72	0.65	0.77	0.57

Table 4.14: Comparing the OOD rejection of the N binary formulation and the one N-ary formulation with the multiple and maximum aposterior strategies. The results suggest that one N-ary formulation improved over the N binary formulation.

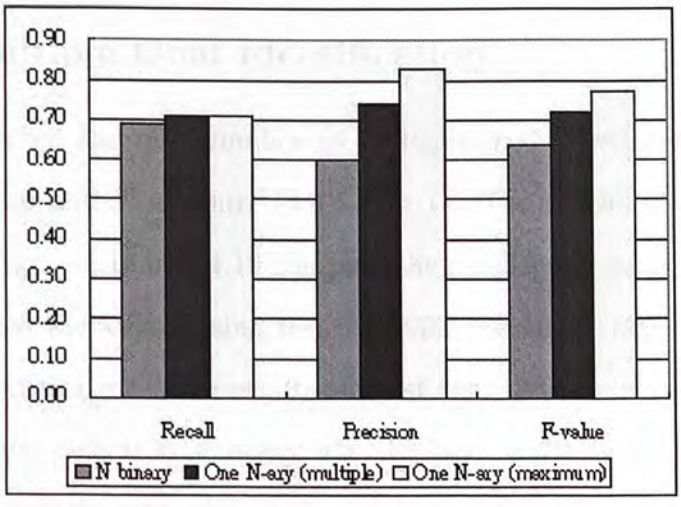

Figure 4.16: Comparing the recalls, precisions and F-values among the goal identifiers in the N binary formulation and the one N-ary formulation with the multiple and maximum selection strategies using test set 1993 in OOD rejection.

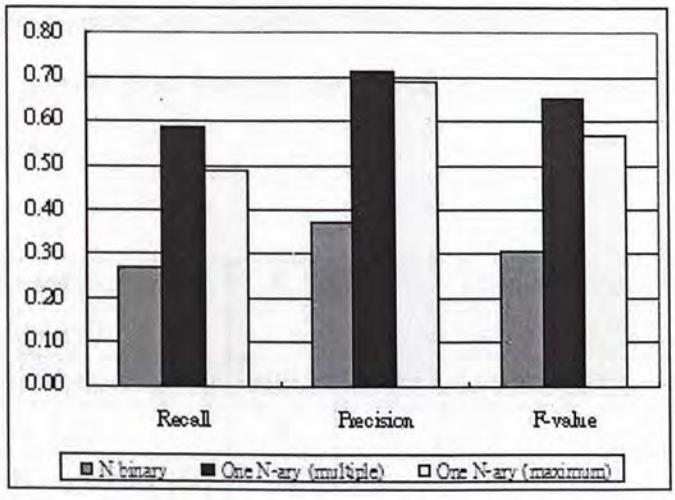

Figure 4.17: Comparing the recalls, precisions and F-values among the goal identifiers in the N binary formulation and the one N-ary formulation with the multiple and maximum selection strategies using test set 1994 in OOD rejection.

4.2.5 Multiple Goal Identification

We also analyzed the performance in multiple goal identification based on recall, precision and F-measure (with $\beta = 1$). The results are tabulated in Table 4.15. Figure 4.18 and 4.19 compare the recalls, precisions and F-values among the goal identifiers using test set 1993 and 1994 respectively in multiple goal identification. The results suggest that the one N-ary formulation using maximum selection strategy has the best multiple goal classification performance, which outperforms the N binary formulation up to 30% for test set 1993 and 39% for test set 1994. It is because the interdependency among the goals is effective in reducing insertion errors in the one N-ary formulation, as mentioned subsection 4.2.3. However, the θ in the relative threshold ($\theta \times \max P(G = g|\vec{C})$) of the multiple selection strategy was tuned

too low for some single goal queries. This increases the number of inferred multiple goal queries (row five) in the one N-ary formulation using multiple selection strategy.

Formulation (strategy)	N **binary**		One N-ary (multiple)		One N-ary (maximum)	
Test set	**1993**	**1994**	**1993**	**1994**	**1993**	**1994**
# MG queries (A)	8	6	8	6	8	6
# inferred MG queries (B)	48	29	46	28	13	10
# correctly inferred MG queries (C)	5	2	5	2	5	4
recall (C/A)	0.63	0.33	0.63	0.33	0.63	0.67
precision (C/B)	0.10	0.07	0.11	0.07	0.38	0.40
F-value	0.18	0.11	0.19	0.12	0.48	0.50

Table 4.15: Experimental results comparing the multiple goal (MG) identification of the N binary formulation and the one N-ary formulation with multiple and maximum aposterior strategies.

4.2.6 Computation

Since we adopted a stochastic approach for our NLU framework, computational costs are inevitable. When we train BNs, we estimate the probabilities by tallying the counts from the training data. When the BNs infer query's goal(s), they perform Bayesian inference. The one N-ary formulation requires a single BN while the N binary formulation requires 11 BNs. We compare the two formulations in terms of the number of additive and multiplicative

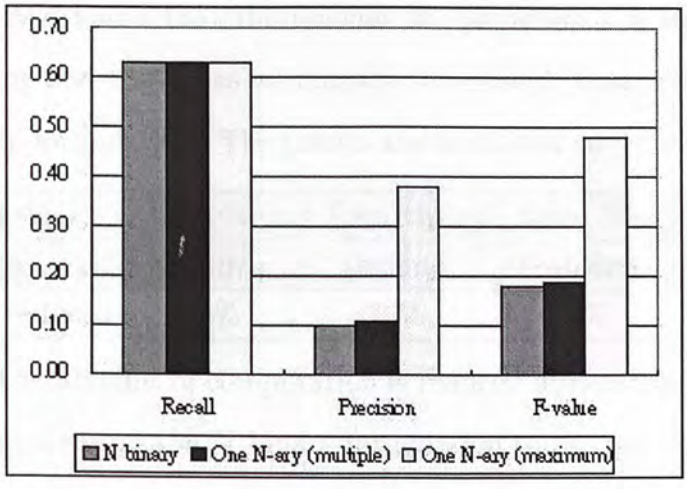

Figure 4.18: Comparing the recalls, precisions and F-values among the goal identifiers in the N binary formulation and the one N-ary formulation with the multiple and maximum selection strategies using test set 1993 in multiple goal identification.

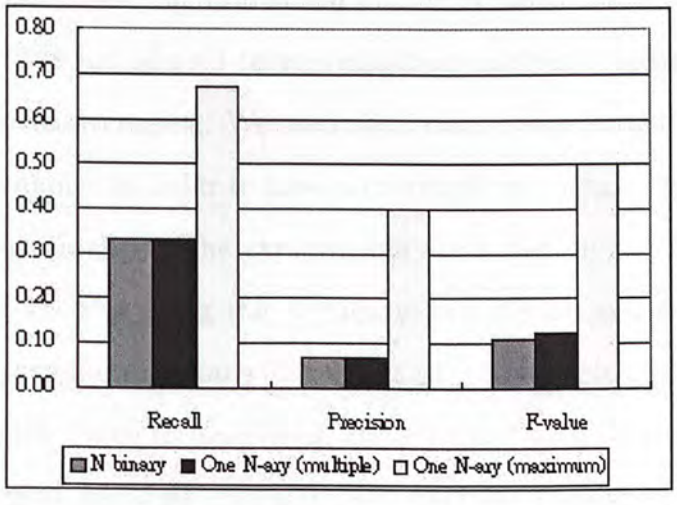

Figure 4.19: Comparing the recalls, precisions and F-values among the goal identifiers in the N binary formulation and the one N-ary formulation with the multiple and maximum selection strategies using test set 1994 in multiple goal identification.

operations[2]. We found that the amount of computation is highly reduced during training and testing as we migrate from the N binary formulation to the one N-ary formulation. The results are tabulated on Table 4.16.

Formulation	**One N-ary (multiple)**		**One N-ary (maximum)**	
Stage	**training**	**testing**	**training**	**testing**
Operations reduced	94%	70%	93%	47%

Table 4.16: The amount of computation is reduced during training and testing as we migrate from the N binary formulation to the one N-ary formulation.

4.3 Chapter Summary

This chapter describes our evaluation on the N binary and the one N-ary formulations. We introduced three evaluation metrics – accuracy measure, macro- and micro-averaging. We used all of them to evaluate the goal identification performance in order to have a thorough understanding on each goal identifier's performance. The experiments are based on the ATIS corpora. We developed 11 BNs using the N binary formulation and two single BNs for the one N-ary formulation with respect to the multiple and maximum selection strategies. Both formulations, accompanied with their strategies, can handle single goal, multiple goal and OOD queries. The experimental results showed that the one N-ary formulation improved over the N binary formulation in (i) overall goal identification performance, (ii) out-of-domain rejection

[2] The two formulations were implemented with different platforms. The N binary formulation was implemented with Hugin software while the one N-ary formulation was implemented with C program only. Therefore, we cannot directly compare the computation in term of operation time.

and (iii) multiple goal identification. This is mainly due to the interdependency among the goals in the one N-ary formulation as $\sum_g P(G = g|\vec{C}) = 1$ for $g \in \{g_1, g_2 \ldots g_N\}$. This feature reduces the number of insertion and substitution errors. The amount of computation is reduced over 90% in the training and up to 70% in the testing stage when we migrate from the N binary formulation to the one N-ary formulation. Our experiments also suggested that the one N-ary formulation have a better NLU performance in general when using the maximum selection strategy.

Chapter 5

Portability to Chinese

We have conducted experiments to compare the natural language understanding performance of the Belief Network framework in making N binary decisions and one N-ary decision, using the English ATIS corpora. We found that the one N-ary formulation using the maximum selection strategy has the best goal identification performance. In this chapter, we attempt to apply this formulation to Cantonese Chinese. The experiments are still based on the ATIS domain in order to demonstrate the language portability. We evaluate the performance in terms of (i) overall goal identification performance, (ii) out-of-domain rejection and (iii) multiple goal identification.

5.1 The Chinese ATIS Domain

We investigate the language portability of using Belief Network (BN) in natural language understanding (NLU). 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. Table 5.1

shows three examples of the translated Chinese queries.

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"
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阿密最平既頭等來回機位"
Goals:	flight.flight_id, fare.fare_id

Out-of-domain (OOD) example

Query:	"*how many first class flights does united have leaving from all cities today*"
Translated query:	"今日有幾多班聯合航空既頭等航機起飛"
Goal:	count_flight (OOD, count_flight is not selected as in-domain)

Table 5.1: Single goal, multiple goal and OOD examples of translated Cantonese Chinese sentences from the ATIS-3 Class A training corpus.

5.1.1 Word Tokenization and Parsing

The Chinese language has no explicit delimiter for word boundaries. Hence, the translated queries on Table 5.1 are in form of consecutive Chinese characters. We tokenize each Chinese query into words by a forward maximum-matching algorithm using a Cantonese lexicon, CULEX [13]. We extended the lexicon with the city names and airport names that we found in the

ATIS-3 training set. After that, the words are parsed into semantic concepts using hand-designed grammar rules (listed on Appendix C). The sequence of semantic concepts form the input to our BN. Table 5.2 is an example to show the processes of word tokenization and parsing.

We have 64 semantic concepts for the Chinese ATIS. In comparison, English ATIS has 60 semantic concepts. There are some semantic concepts defined for both English and Chinese ATIS, such as <CITY_NAME> and <AIRLINE_NAME>, in order to obtain the semantic information in common. However, some tags are designed for English or Chinese ATIS only. For example in Table 5.2, <FLIGHT_TYPE> (row five) is an unique tag for the Chinese query to capture "早機" (the flights in the morning) while English query uses <FLIGHT> and <PERIOD> in separate positions (row two).

5.2 Experiments

Our experiments are based on the Chinese ATIS-3 Class A sentences in the training set, test set 1993 and test set 1994. We prepared the corpora by word tokenization and parsing as mentioned earlier. The BN adopts the one N-ary formulation using the maximum selection strategy and applies Bayesian inference as it does in English. One N-ary decision is made by choosing the goal g, $g \in \{g_1, g_2 \ldots g_{16}\}$, with the maximum aposterior probability, $P(G = g|\vec{C})$. We first decide the parameter for the network dimension. After that, we use the trained BN for the goal identification. We compare the NLU performance with the same formulated BN in English in terms of (i) overall goal identification performance, (ii) out-of-domain rejection and (iii) multiple goal identification.

Original query:	"*give me the meal flights departing early saturday morning from chicago to seattle nonstop*"
Semantic concepts: (English)	<CHUNK> <DUMMY> <MEAL> <FLIGHT> <FROM> <MODIFIER> <DAY_NAME> <PERIOD> <FROM> <CITY_NAME> <TO> <CITY_NAME_1> <STOPS>
Translated query:	"我要星期六芝加哥直飛西雅圖有飛機餐既早機"
Word tokenization:	我 / 要 / 星期六 / 芝加哥 / 直飛 / 西雅圖 / 有 / 飛機餐 / 既 / 早機
Semantic concepts: (Chinese)	<QUERY> <DAY_NAME> <CITY_NAME> <STOPS> <CITY_NAME_1> <CHUNK> <MEAL> <CHUNK> <FLIGHT_TYPE>
Goal:	flight.flight_id

Table 5.2: An example illustrating the processes of word tokenization and parsing.

5.2.1 Network Dimension

To determine the number of input concepts (M) which has the highest Information Gain with the single BN, we varied M from 10 to the full set of 64. We evaluated the goal identification performance for each value of M by micro-averaging. The experiments were conducted with the training data. We selected the value for M which gives the optimal F-value or has less than 0.001 marginal improvement. The results are plotted on Figure 5.1 which suggests the most suitable value for M is 55. The network dimension is the same as we selected for the single BN in the English ATIS domain using the same goal identification strategy (maximum selection strategy). The selected concepts for the single BN in the Chinese ATIS domain are listed in Appendix D.

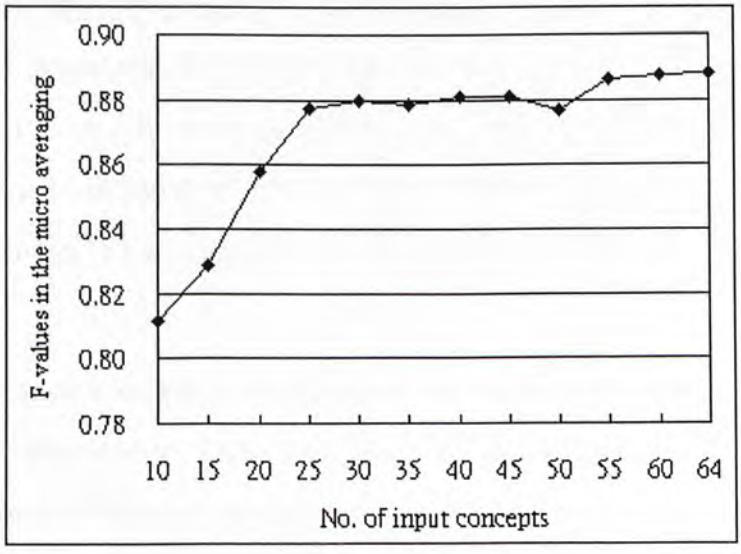

Figure 5.1: The F-values in the micro-averaging vary with the number of the input concepts in the one N-ary formulation. The results suggest that we should use 55 concepts in the single BN for the Chinese ATIS domain.

5.2.2 Overall Goal Identification

We evaluated the overall goal identification performance of the Chinese ATIS queries by the accuracy measure, macro- and micro-averaging. We compared the results between the English and Chinese using test sets 1993 and 1994.

Evaluation Metric 1: Accuracy Measure

Language	**Chinese**		**English**	
Test set	**1993**	**1994**	**1993**	**1994**
# DEL	3	3	3	2
# INS	8	6	8	6
# SUB	32	41	28	39
Total # errors	43	50	39	47
Goal identification accuracies	90.4% ($\frac{403}{446}$)	88.8% ($\frac{398}{448}$)	91.3% ($\frac{407}{446}$)	89.5% ($\frac{401}{448}$)

Table 5.3: Comparing the goal identification accuracies in Chinese and English using the one N-ary formulation with maximum selection strategies. The comparison is based on the number of deletion (DEL), insertion (INS) and substitution (SUB) errors produced in test sets 1993 and 1994.

The overall goal identification accuracies of the Chinese and English ATIS queries are tabulated in Table 5.3. Figure 5.2 and 5.3 show the comparison of the numbers of deletion, insertion and substitution errors between Chinese and English using test sets 1993 and 1994 respectively. We found that the accuracies in Chinese degraded by less than 1%. This is mainly due to the increase of the substitution errors. The degradations came from the Chinese expressions containing more semantic concepts which can lead to an

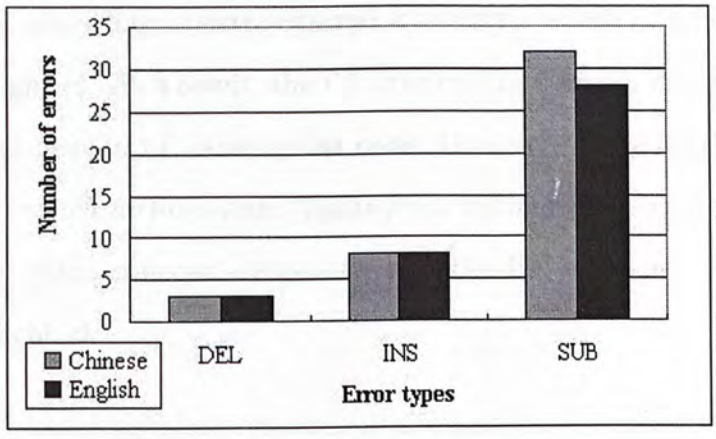

Figure 5.2: Comparing the numbers of deletion (DEL), insertion (INS) and substitution (SUB) errors between Chinese and English using the one N-ary formulation with maximum selection strategies using test set 1993.

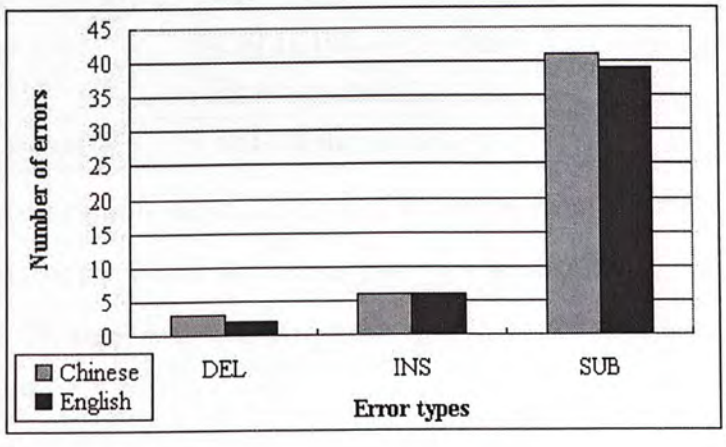

Figure 5.3: Comparing the numbers of deletion (DEL), insertion (INS) and substitution (SUB) errors between Chinese and English using the one N-ary formulation with maximum selection strategies using test set 1994.

incorrect goal inferred. Table 5.4 is an example which illustrates this effect. The Chinese query has an extra concept <FLIGHT> which is indicative to the goal flight.flight_id. As a result, the BN inferred the Chinese query to the goal flight.flight_id instead of airline.airline_code. However, if the English query is changed to "*which airlines have flights from baltimore to san francisco*", we will have an extra concept <FLIGHT> and the BN will infer to the wrong goal flight.flight_id.

Original query:	"*list which airlines fly from baltimore to san francisco*"
Semantic concepts: (English)	<WHICH> <AIRLINE> <FROM> <CITY_NAME> <TO> <CITY_NAME_1>
Reference goal:	airline.airline_code
Translated query:	"邊 間 航空 公司 有 航機 由 巴的摩爾 飛去 三藩市"
Semantic concepts: (Chinese)	<QUERY> <AIRLINE> <CHUNK> <FLIGHT> <FROM> <CITY_NAME> <TO> <CITY_NAME_1>
Inferred goal:	flight.flight_id (SUB)

Table 5.4: An example shows that the Chinese translation contains an extra concept <FLIGHT> which led to an incorrect goal inferred. A substitution error in the Chinese, which lowered the goal identification accuracies in the Chinese ATIS.

Evaluation Metric 2: Macro-Averaging

We compared the goal identification performance of the single BN in Chinese and English by macro-averaging. The results are shown on Table 5.5. The recall and precision of each goal using test sets 1993 and 1994 are listed

Language	Chinese		English	
Test set	1993	1994	1993	1994
$recall_{Macro}$	0.90	0.76	0.90	0.75
$precision_{Macro}$	0.76	0.65	0.88	0.63
F-value	0.83	0.70	0.89	0.69

Table 5.5: Comparison of the overall goal identification performance in Chinese and English using macro-averaging.

in Appendix E. Figure 5.4 and 5.5 compare the recalls, precisions and F-values between Chinese and English using test sets 1993 and 1994 respectively. Macro-averaging tends to over-emphasize the performance on the rare goals. The results show the F-value in macro-averaging of test set 1993 is lower in Chinese because the rare goals, such as airline.airline_code and airport.airport_name, had lower precisions. However, the rare goals in test set 1994 had equal performance in Chinese and English and some high frequent goals had better performances in Chinese. Therefore, the F-values in macro-averaging of test set 1994 are a little bit higher in Chinese.

Evaluation Metric 3: Micro-Averaging

We also evaluated the overall goal identification in test set 1993 and test set 1994 by micro-averaging. The results are tabulated in Table 5.6, which shows that the performance in Chinese is degraded by less than 1%. Figure 5.6 and 5.7 compare the recalls, precisions and F-values between Chinese and English using test sets 1993 and 1994 respectively. It is consistent with the results in the accuracy measure, even we measured in a query-based averaging algorithm. It is because the performances of the high frequent

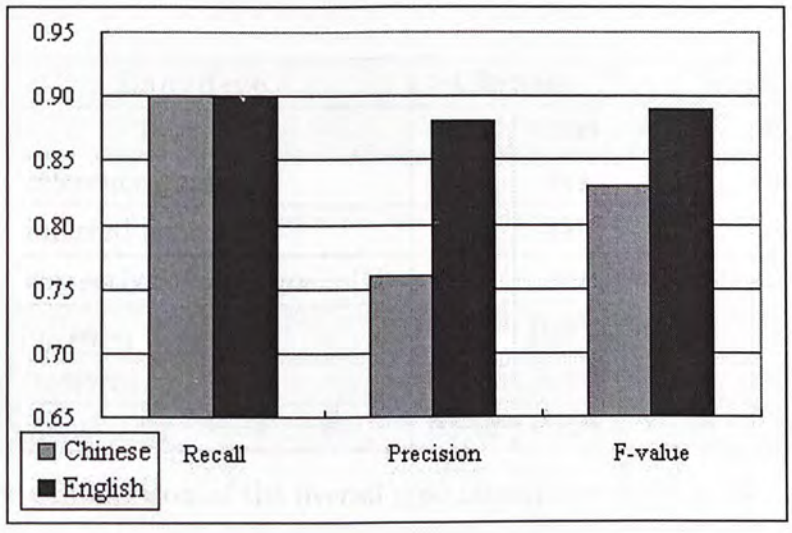

Figure 5.4: Comparing the recalls, precisions and F-values between Chinese and English using the one N-ary formulation with maximum selection strategies using test set 1993 in macro-averaging.

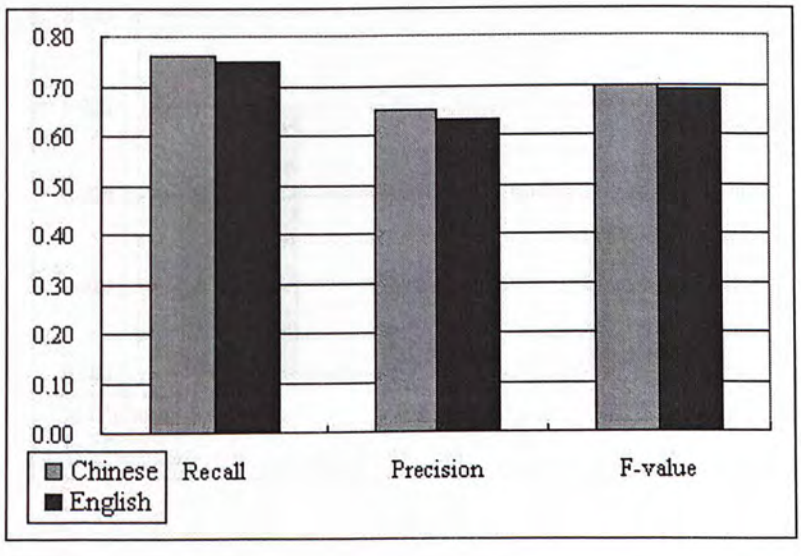

Figure 5.5: Comparing the recalls, precisions and F-values between Chinese and English using the one N-ary formulation with maximum selection strategies using using test set 1994 in macro-averaging.

Language	Chinese		English	
Test set	**1993**	**1994**	**1993**	**1994**
# reference goals (A)	446	448	446	448
# inferred goals (B)	451	451	451	452
# correctly inferred goals (C)	411	404	415	407
$recall_{Micro}$ (C/A)	0.922	0.902	0.930	0.908
$precision_{Micro}$ (C/B)	0.911	0.896	0.920	0.900
F-value	0.916	0.899	0.925	0.904

Table 5.6: Comparison of the overall goal identification performance in Chinese and English using micro-averaging.

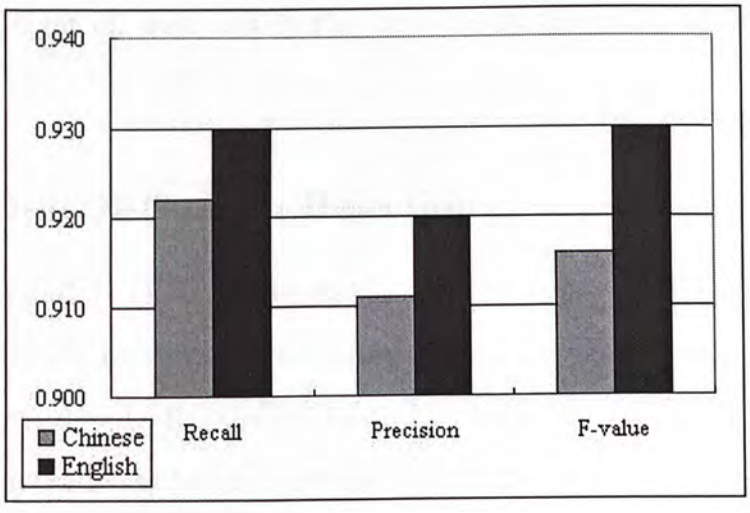

Figure 5.6: Comparing the recalls, precisions and F-values between Chinese and English using the one N-ary formulation with maximum selection strategies using test set 1993 in micro-averaging.

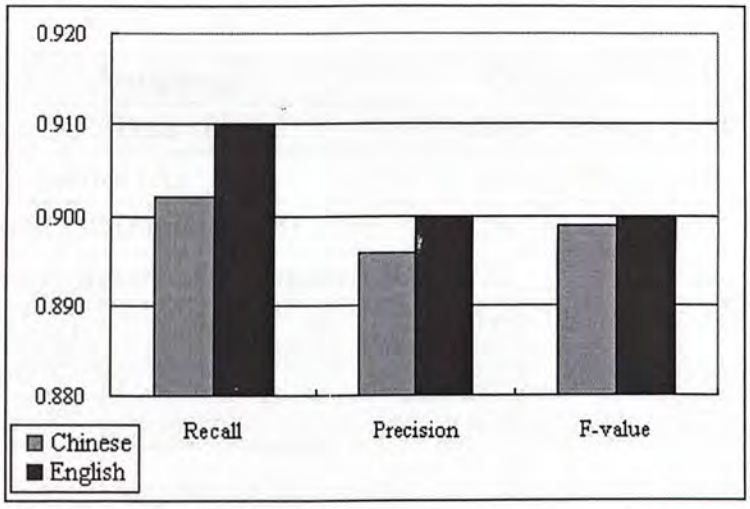

Figure 5.7: Comparing the recalls, precisions and F-values between Chinese and English using the one N-ary formulation with maximum selection strategies using using test set 1994 in micro-averaging.

goal, flight.flight_id, were nearly the same in the English and Chinese test sets.

5.2.3 Out-Of-Domain Rejection

We have 35 and 37 OOD queries in the test sets 1993 and 1994 respectively. We analyzed the rejection performance in terms of recall, precision and F-measure with $\beta = 1$. Results on Table 5.7 show that the OOD rejection in Chinese is degraded in test set 1993 but it is better than in English in test set 1994. Figure 5.8 and 5.9 compare the recalls, precisions and F-values between Chinese and English using test sets 1993 and 1994 respectively. We found that the difference in performance is due to the different hand-defined English and Chinese grammar rules, which are used for semantic tagging. There are some words in the English grammars that do not have corresponding coun-

Language	Chinese		English	
Test set	**1993**	**1994**	**1993**	**1994**
# OOD queries (A)	35	37	35	37
# inferred OOD queries (B)	24	32	30	26
# correctly inferred OOD queries (C)	22	26	25	18
recall (C/A)	0.63	0.70	0.71	0.49
precision (C/B)	0.92	0.81	0.83	0.69
F-value	0.75	0.75	0.77	0.57

Table 5.7: Comparing the OOD rejection in Chinese and English based on recall, precision and F-measure ($\beta = 1$).

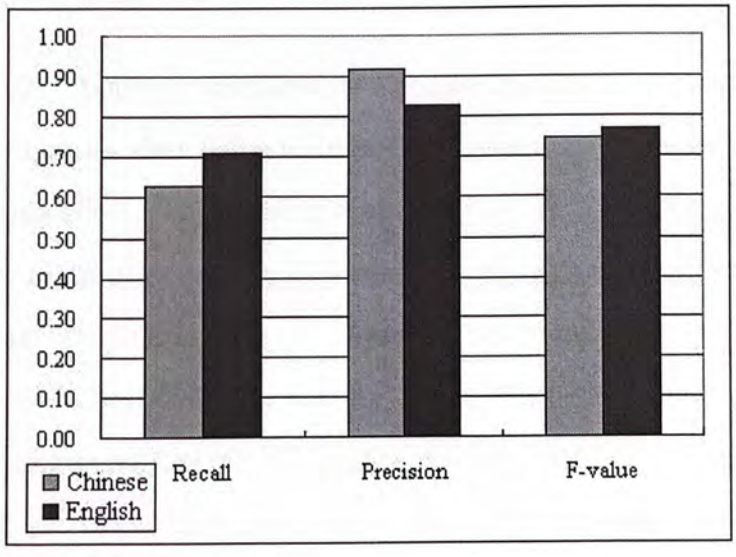

Figure 5.8: Comparing the recalls, precisions and F-values between Chinese and English using the one N-ary formulation with maximum selection strategies using test set 1993 in OOD rejection.

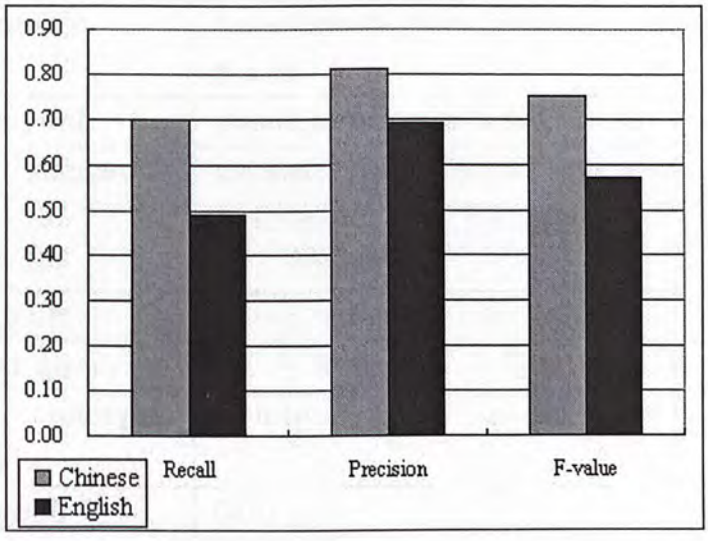

Figure 5.9: Comparing the recalls, precisions and F-values between Chinese and English using the one N-ary formulation with maximum selection strategies using using test set 1994 in OOD rejection.

terparts in the Chinese grammars. As a result, missing some semantic tags in the Chinese queries that helps to infer OOD goal. An example on Table 5.8 illustrates this effect. The missing semantic concept <TRANSPORT>, which is indicative of an in-domain goal ground_service.city_code, helps identify the OOD goal in the Chinese query. However, we found that the OOD goals in test sets 1993 and 1994 are not the same. There are seven queries with ground_service.ground_fare (OOD) goal in test set 1994 but no query with this goal in test set 1993. Therefore, the difference in grammar rules does not benefit test set 1993.

Original query:	*"what are the fares for ground transportation in denver"*
Reference goal:	ground_service.ground_fare (OOD)
Semantic concepts: (English)	<WHAT> <CHUNK> <DUMMY> <FARE> <PREP> <TRANSPORT> <PREP> <CITY_NAME>
Inferred goal:	ground_service.city_code (SUB)
Translated query:	"係 丹佛 既 地面 交通車 費 要 幾多"
Semantic concepts: (Chinese)	<CHUNK> <CITY_NAME> <CHUNK> <HOW>
Inferred goal:	OOD (✓)

Table 5.8: An example illustrates that a Chinese expression is parsed by insufficient grammar rules. Missing semantic concepts is resulted but it helps to identify OOD goal.

5.2.4 Multiple Goal Identification

We have 8 and 6 multiple goal queries in test sets 1993 and 1994 respectively. We compared the multiple goal identification in Chinese and English based on recall, precision and F-measure with $\beta = 1$. The results are shown on Table 5.9. Figure 5.10 and 5.11 compare the recalls, precisions and F-values between Chinese and English using test sets 1993 and 1994 respectively. We found that the multiple goal identification performances in Chinese and English are the same in test set 1993. However, the BN failed in identifying one multiple goal Chinese query in test set 1994. Therefore, the multiple goal identification performance in Chinese is lower in test set 1994. The reason of this failure came from the Chinese query with extra concepts, which is similar to the example on Table 5.4.

Language	Chinese		English	
Test set	1993	1994	1993	1994
# MG queries (A)	8	6	8	6
# inferred MG queries (B)	13	9	13	10
# correctly inferred MG queries (C)	5	3	5	4
recall (C/A)	0.63	0.50	0.63	0.67
precision (C/B)	0.38	0.33	0.38	0.40
F-value	0.47	0.40	0.47	0.50

Table 5.9: Comparing the multiple goal (MG) identification in Chinese and English based on recall, precision and F-measure ($\beta = 1$).

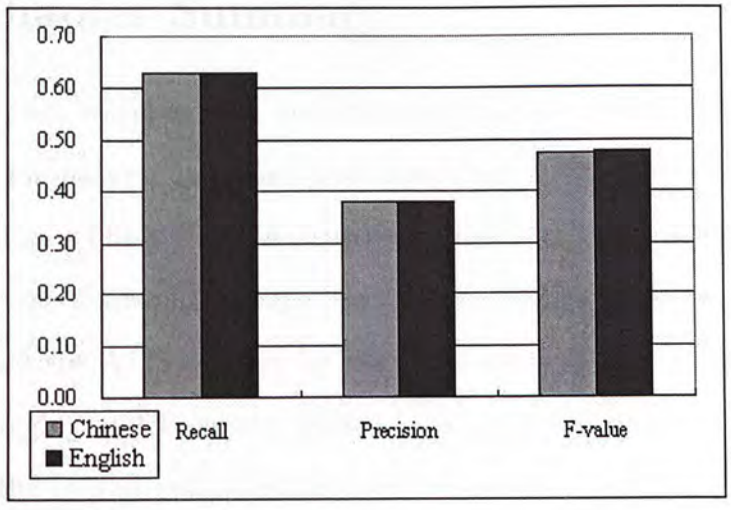

Figure 5.10: Comparing the recalls, precisions and F-values between Chinese and English using the one N-ary formulation with maximum selection strategies using test set 1993 in multiple goal identification.

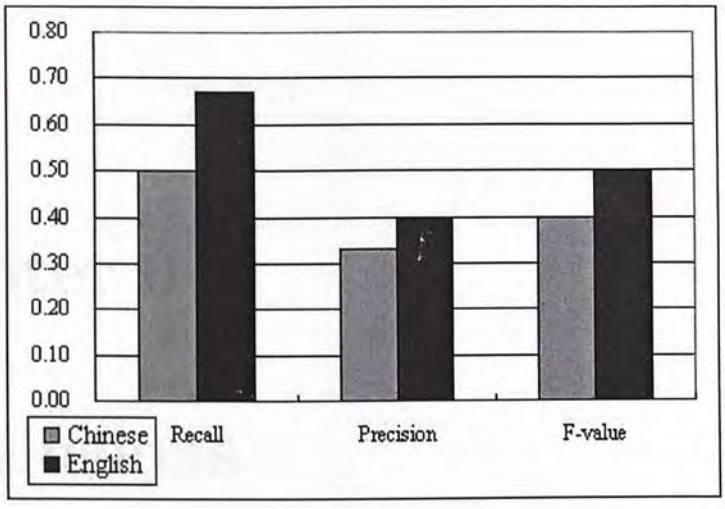

Figure 5.11: Comparing the recalls, precisions and F-values between Chinese and English using the one N-ary formulation with maximum selection strategies using using test set 1994 in multiple goal identification.

5.3 Chapter Summary

In this chapter, we presented our attempt in applying the Belief Network framework for natural language understanding in Chinese. We manually translated the ATIS-3 Class A sentences from English to Cantonese Chinese. Since the Chinese language has no delimiter for word boundaries, we pre-processed the ATIS corpora by word tokenization. Then, we performed semantic tagging and Bayesian inference as we did in English. The results show that the overall goal identification performance in Chinese suffers less than 1% degradation in both test sets under the accuracy measure. The degradation is due to more concepts in the Chinese queries. We found that the Belief Network framework is portable and usable in the English and Chinese languages.

Chapter 6

Conclusions

6.1 Summary

In this thesis, we have extended the use of a pre-existing Belief Network (BN) framework [31] for natural language understanding (NLU). A method was derived for identifying the user's communicative goal(s) out of a finite set of domain-specific goals for an information-seeking query. The problem was formulated as making N binary decisions, each performed by a BN. We have presented how to make an one N-ary decision in a *single* BN. The BN structure captures the dependencies between the communicative goals and the semantic concepts. Semantic information is stored as statistical parameters, which are used for Bayesian inference. We have proposed two goal identification strategies for the one N-ary formulation: multiple selection strategy and maximum selection strategy. Both are capable in identifying single goal, multiple goals as well as out-of-domain (OOD) goal in the single BN.

We have three goal identifiers in total: (i) a suite of BNs modeled under

the N binary formulation, (ii) a single BN modeled under the one N-ary formulation with *multiple* selection strategy and (iii) a single BN modeled under the one N-ary formulation with *maximum* selection strategy. We have proposed three evaluation metrics – the *accuracy measure*, *macro-averaging* and *micro-averaging*. We used all of them to evaluate the overall goal identification in order to have a thorough understanding on each goal identifier's performance. The experiments are based on the ATIS (Air Travel Information Service) corpora. The experimental results showed that the one N-ary formulation improved over the N binary formulation in (i) overall goal identification performance, (ii) OOD rejection and (iii) multiple goal identification. This is mainly due to the interdependency among the goals in the one N-ary formulation as $\sum_g P(G = g|\vec{C}) = 1$ for $g \in \{g_1, g_2 \dots g_N\}$. This feature reduces the number of insertion and substitution errors. Furthermore, the amount of computation is reduced over 90% in the training and up to 70% in the testing phases when we migrate from the N binary formulation to the one N-ary formulation. Our experiments also suggested that the one N-ary formulation have a better NLU performance in general when using the maximum selection strategy.

We have presented our attempt in applying the BN framework for understanding Cantonese Chinese. We manually translated the ATIS-3 Class A sentences from English to Chinese. Since the Chinese language has no explicit delimiter for word boundaries, we pre-processed the ATIS corpora by word tokenization. Then, we performed semantic tagging and Bayesian inference as we did in English.The results show that the overall goal identification accuracies in Chinese suffer less than 1% degradation due to more

concepts in the Chinese queries.

6.2 Contributions

In this work, the following contributions are made to the field of natural language understanding:

1. We have demonstrated an alternative formulation – the one N-ary formulation – for a BN framework in natural language understanding. The one N-ary formulation captures the *interdependency* among the communicative goals as $\sum_g P(G = g|\vec{C}) = 1$ for $g \in \{g_1, g_2 \ldots g_N\}$. It gave improvement over the N binary formulation in terms of the overall goal identification, out-of-domain rejection and multiple goal identification.

2. The one N-ary formulation uses a single BN while the N binary formulation needs N BNs, one for each goal. The amount of *computation* in training and Bayesian inference has been reduced in a single BN.

3. We have introduced different *evaluation metrics* for measuring the overall goal identification performance. The accuracy measure is an alignment measure in relation to the number of errors. Macro- and micro-averaging evaluate the NLU performance as a categorization problem, which classifies a query with respect to a finite set of goals. All evaluation metrics are useful to provide a thorough understanding of a goal identifier's performance.

4. The BN framework automatically learns the linguistic knowledge from training data. We have shown the BN framework in the one N-ary

formulation is *portable* across languages. We migrated from English to Cantonese Chinese.

6.3 Future Work

Possible extensions of this work include:

1. Developing a learnt BN in the one N-ary formulation. In this thesis, we adopted a naive Bayes' configuration in a single BN. The concepts are assumed to be independent of one another. The learnt BN topology have been applied on the N binary formulation by building interdependencies among the concepts [30]. The results showed improvement in the goal identification accuracies. We may also build linkages between the concepts in the single BN. The enhanced topology should further improve the goal identification performance.

2. Extending the comparison in the CUHK Restaurants domain [9]. We showed improvement of BN framework in the one N-ary formulation using the ATIS corpora. We may leverage the comparison of the two BN formulations from the ATIS domain to the the CUHK Restaurants domain, which contains single goal utterance only. In order to make a fair comparison, we should modify the goal identification strategy in the N binary and the one N-ary formulations to infer a single output.

3. Integrating communicative intention and goal in a single Bayesian inference. Communicative intention is the user's act of will in a given utterance, such as requesting suggestion and saying thanks. Communicative goal is the domain specific of a user's request. For example,

ordering food and billing in a restaurants domain. To understand a sentence, the identification of the communicative intention is as important as the domain-specific goal. Using different BNs for identifying the communicative intention and goal separately requires high computational cost and redundant procedures. We may develop a single BN to identify sentence's intention and goal together.

Bibliography

[1] B. Ambrosio. "Inference in Bayesian Networks". In *Artificial Intelligence Magazine, Vol. 20(2)*, pages 21–36, 1999.

[2] K. Arai, J. Wright, G. Riccardi, and A. Gorin. "Grammar Fragment Acquisition Using Syntactic and Semantic Clustering". In *Proceedings of the 5^{th} International Conference on Spoken Language Processing*, 1998.

[3] H. Aust, M. Oerder, F. Seide, and V. Steinbiss. "The Philips Automatic Train Timetable Information System". In *Speech Communication, Vol. 17*, pages 249–262, 1995.

[4] S. Bennacef, L. Devillers, S. Rosset, and L. Lamel. "Dialog in the RAILTEL Telephone-Based System". In *Proceedings of the 4^{th} International Conference on Speech and Language Processing*, pages 550–553, 1996.

[5] M. Boros and P. Heisterkamp. "Linguistic Phrase Spotting in a Simple Application Spoken Dialogue System". In *Proceedings of the 6^{th} European Conference on Speech Communication and Technology*, 1999.

[6] C. Bousquet, N. Vigouroux, and G. Perennou. "Stochastic Conceptual Model for Spoken Language Understanding". In *Proceeding of the Workshop SPECOM*, pages 71–74, 1999.

[7] R. Carlson and S. Hunnicutt. "Generic and Domain-Specific Aspects of the Waxholm NLP and Dialog Modules". In *Proceedings of the 4^{th} International Conference on Spoken Language Processing*, pages 677–680, 1996.

[8] M. Castro and E. Sanchis. "A Simple Connectionist Approach to Language Understanding in a Dialogue System". In *Proceedings of the 8^{th} Ibero-American Conference on Artificial Intelligence*, pages 664–673, 2002.

[9] S. Chan and H. Meng. "Interdependencies among Dialog Acts, Task Goals and Discourse Inheritance in Mixed-Initiative Dialog". In *Proceedings of the Human Language Technology Conference*, 2002.

[10] E. Charniak. "Bayesian Networks without Tears". In *Artificial Intelligence Magazine, Vol. 12(4)*, pages 50–63, 1991.

[11] B. Chen, H. Wang, L. Chien, and L. Lee. "A*-Admissible Key-Phrase Spotting with Sub-Syllable Level Utterance Verification". In *Proceedings of the 5^{th} International Conference on Spoken Language Processing*, 1998.

[12] Nuance Communications. http://www.nuance.com/, 2003.

[13] CULEX. http://dsp.ee.cuhk.edu.hk/speech/cucorpora/, 2003.

[14] J. Glass, G. Flammia, D. Goodine, M. Phillips, J. Polifroni, S. Sakai, S. Seneff, and V. Zue. "Multilingual Spoken-Language Understanding in the MIT Voyager System". In *Speech Communication, Vol. 17*, pages 1–18, 1995.

[15] J. Gomez-Hidalgo and M. Rodriguez. "Integrating a Lexical Database and a Training Collection for Text Categorization". In *Proceedings of the ACL/EACL Workshop on Automatic Extraction and Building of Lexical Semantic Resource for Natural Language Application*, 1997.

[16] M. Hasan and K. Lau. "Semantic Category Disambiguation of Chinese Lexicon using Neural Networks". In *Proceedings of the Joint Pacific Asian Conference on Expert Systems*, pages 469–477, 1997.

[17] C. Hemphill, J. Godfrey, and G. Doddington. "The ATIS Spoken Language Systems Pilot Corpus". In *Proceedings of the DARPA Speech and Natural Language Workshop*, pages 96–101, 1990.

[18] SpeechWorks International. http://www.speechworks.com/, 2003.

[19] A. Jain and A. Waibel. "Robust Connectionist Parsing of Spoken Language". In *Proceedings of the International Conference on Acoustic, Speech and Signal Processing*, pages 593–596, 1990.

[20] V. Jensen. *An Introduction to Bayesian Networks.* UCL Press, 1996.

[21] M. Johnson, S. Bangalore, and G. Vasireddy. "MATCH: Multimodal Access To City Help". In *Proceedings of Automatic Speech Recognition and Understanding Workshop*, 2001.

[22] D. Jurafsky, C. Wooters, G. Tajchman, J. Segal, A. Stolcke, E. Fosler, and N. Morgan. "The Berkeley Restaurant Project". In *Proceedings of the 3^{rd} International Conference on Spoken Language Processing*, pages 2139–2142, 1994.

[23] T. Kawahara, M. Araki, and S. Doshita. "Comparison of Parsing and Spotting Approaches for Spoken Dialogue Understanding". In *Proceedings of the European Speech Communication Association Workshop on Spoken Dialogue Systems*, pages 21–24, 1995.

[24] A. Kellner, B. Rueber, F. Seide, and B. Tran. "PADIS – An Automatic Telephone Switchboard and Directory Information System". In *Speech Communication, Vol. 23*, pages 95–111, 1997.

[25] S. Knight, G. Gorrell, M. Rayner, D. Milward, R. Koeling, and L. Lewin. "Comparing Grammar-Based and Robust Approaches to Speech Understanding: a Case Study". In *Proceedings of the 7^{th} European Conference on Speech Communication and Technology*, 2001.

[26] L. Lamel, S. Bennacef, J. Gauvain, H. Dartigues, and J. Temem. "User Evaluation of the MASK Kiosk". In *Proceedings of the 5^{th} International Conference on Spoken Language Processing*, pages 2875–2878, 1998.

[27] K. Macherey, F. Och, and H. Ney. "Natural Language Understanding Using Statistical Machine Translation". In *Proceedings of the 7^{th} European Conference on Speech Communication and Technology*, 2001.

[28] B. Manaris. "Natural Language Processing: A Human - Computer Interaction Perspective". In *Advances in Computers, Vol. 47*, pages 1–66, 1998.

[29] H. Meng, W. Lam, and K. Low. "A Bayesian Approach for Understanding Informational-Seeking Queries". In *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, 1999.

[30] H. Meng, W. Lam, and K. Low. "Learning Belief Networks for Language Understanding". In *Proceedings of the 1999 International Workshop on Automatic Speech Recognition and Understanding*, 1999.

[31] H. Meng, W. Lam, and C. Wai. "To Believe is to Understand". In *Proceedings of the 6^{th} European Conference on Speech Communication and Technology*, 1999.

[32] H. Meng, S. Lee, and C. Wai. "CU FOREX: A Bilingual Spoken Dialog System for the Foreign Exchange Domain". In *Proceedings of International Conference on Acoustics, Speech and Signal Processing*, 2000.

[33] W. Minker, S. Bennacef, and J. L. Gauvain. "A Stochastic Case Frame Approach for Natural Language Understanding". In *Proceedings of the 4^{th} International Conference on Spoken Language Processing*, pages 1013–1016, 1996.

[34] R. Pieraccini, E. Tzoukermann, Z. Gorelov, J. Gauvain, E. Levin, C. Lee, and J. Wilpon. "A Speech Understanding System Based on Statistical Representation of Semantics". In *Proceedings of International Conference of Acoustics, Speech and Signal Processing*, pages 193–196, 1992.

[35] J. Polifroni, S. Seneff, J. Glass, and T. Hazen. "Evaluation Methodology for a Telephone-Based Conversational System". In *Proceedings of the 1^{st} International Conference on Language Resources and Evaluation*, pages 43–49, 1998.

[36] P. Price. "Evaluation of Spoken Language Systems: The ATIS Domain". In *Proceedings of the ARPA Human Language Technology Workshop*, pages 91–95, 1990.

[37] G. Riccardi, A. Gorin, A. Ljolje, and M. Riley. "A Spoken Language System for Automated Call Routing". In *Proceedings of International Conference of Acoustics, Speech and Signal Processing*, pages 1143–1146, 1997.

[38] S. Richardson. "Bootstrapping Statistical Processing into a Rule-based Natural Language Parser". In *Proceedings of the Association for Computational Linguistics Workshop*, pages 96–103, 1994.

[39] E. Sanchis and M. Castro. "Dialogue Act Connectionist Detection in a Spoken Dialogue System". In *Proceedings of the 2^{th} International Conference on Hybrid Intelligent Systems*, pages 664–651, 2002.

[40] S. Seneff. "TINA: A Natural Language System for Spoken Language Applications". In *Computational Linguistics, Vol. 18, No. 1*, pages 61–86, 1992.

[41] S. Seneff and J. Polifroni. "Dialogue Management in the Mercury Flight Reservation System". In *Proceedings of the Satellite Dialogue Workshop, ANLP-NAACL*, 2000.

[42] B. Souvignier, A. Kellner, B. Rueber, H. Schramm, and F. Seide. "The Thoughtful Elephant: Strategies for Spoken Dialog Systems". In *IEEE Transactions on Speech and Audio Processing, Vol. 8, No. 1*, pages 51–62, 2000.

[43] W. Xu and A. Rudnicky. "Can Artificial Neural Networks Learn Language Models?". In *Proceedings of the 6^{th} International Conference on Spoken Language Processing*, 2000.

[44] Y. Yang. "An Evaluation of Statistical Approaches to Text Categorization". In *Information Retrival Journal, Vol. 1*, pages 69–90, 1999.

[45] J. Zahradil, L. Muller, and P. Juza. "Key-Phrase Spotting Technique Used in Telephone Dialog System Accessing E-mails via Voice". In *Proceedings of the 6th World Multiconference on Systemics, Cybernetics and Informatics*, pages 381–384, 2002.

[46] V. Zue, S. Seneff, J. Glass, J. Polifroni, C. Pao, T. Hazen, and L. Hetherington. "JUPITER: A Telephone-Based Conversational Interface for Weather Information". In *IEEE Transactions on Speech and Audio Processing, Vol. 8, No. 1*, January 2000.

[47] V. Zue, S. Seneff, J. Polifroni, M. Phillips, C. Pao, D. Goddeau, J. Glass, and E. Brill. "PEGASUS: A Spoken Language Interface for On-Line Air Travel Planning". In *Speech Communication, Vol. 15*, pages 331–340, 1994.

Appendix A

The Communicative Goals

aircraft.aircraft_code *	count_flight
aircraft.aircraft_descriptio	days.day_name
aircraft.basic_type	fare.fare_id *
airline.airline_code *	fare_basis.fare_basis_code *
airline.airline_name *	flight.airline_code
airport.airport_code *	flight.arrival_time
airport.airport_location	flight.departure_time
airport.airport_name *	flight.flight_id *
airport.minimum_connect_time	flight.flight_number *
airport_service.miles_distant	flight.time_elapsed
airport_service.minutes_distant	food_service.meal_code
city.city_code	food_service.meal_description
class_of_service.booking_class	ground_service.city_code *
class_of_service.class_description *	ground_service.ground_fare
count_airline	ground_service.transport_type
count_fare	restriction.restriction_code

Table A.1: The 32 communicative goals in the ATIS domain. The goal with an asterisk (*) are selected for the identification. The remaining goals are treated as out-of-domain (OOD).

Appendix B

Distribution of the Communicative Goals

Goal	Frequency (Training)	Frequency (Test 1993)	Frequency (Test 1994)
aircraft.aircraft_code	13	6	1
airline.airline_code	42	6	11
airline.airline_name	25	18	6
airport.airport_code	10	16	2
airport.airport_name	25	2	2
class_of_service.class_description	15	6	3
fare.fare_id	81	26	25
fare_basis.fare_basis_code	26	11	5
flight.flight_id	1239	302	343
flight.flight_number	10	9	0
ground_service.city_code	47	19	15
OOD	12	35	37

Table B.1: The distribution of the 11 selected goals and out-of-domain (OOD) goal in the training set, test set 1993 and test set 1994.

Appendix C

The Hand-Designed Grammar Rules

<AIRCRAFT> aircraft, plane, aircrafts, planes, airplane, airplanes, aeroplane, aeroplanes
<AIRCRAFT_CODE> d ten, seventy three s, seven fifty seven, m eighty, seven thirty three, m eight zero, seventy two s, d nine s, d c tens, d c ten, <MANUFACTURE> + <DIGIT>, <AIRCRAFT> + <DIGIT>
<AIRLINE> airline, airlines
<AIRLINE_NAME> american, american airline, american airlines, american flights, air canada, alaska airlines, alaska airline, continental, continental airline, continental airlines, canadian airline, canadian airlines, canadian airlines international, delta, delta airline, delta airlines, tower air, america west, northwest, northwest airline, nationair, t w, united, southwest, southwest air, southwest airlines, midwest express, united airline, united airlines, trans world airlines, trans world airline, a a, a c, a s, c o, c p, d l, f f, h p, n w, n x, t w a, u a, u s, u s air, w n, y x, k w

<AIRPORT>
airport, airports

<AIRPORT_NAME>
boston airport, love field, dulles, houston intercontinental, kennedy, kennedy airport, john f kennedy, john f kennedy airport, midway, los angeles international, los angeles international airport, los angeles airport, la guardia, la guardia airport, orlando airport, orlando international, general mitchell, general mitchell international, general mitchell international aiport, ontario airport, ontario international, o'hare, saint petersburg airport, san francisco international, san francisco international airport, san francisco airport, salt lake airport, salt lake city airport, toronto international, toronto international airport, lester pearson airport, newark airport, b n a, b o s, b u r, d a l, d f w, e w r, h o u, i a d,i a h, j f k, l a x, m c o, m a, m k e, o r d, p i e, s f o, s l c, c v g, t p a, l g a, b w i, d t w, y y z

<BACK>
returns, return, returning

<CITY>
cities, city

<CITY_NAME>
westchester, westchester county, atlanta, baltimore, boston, burbank, charlotte, chicago, cincinnati, cleveland, columbus, dallas, denver, detroit, fort worth, houston, indianapolis, kansas city, vegas, las vegas, long beach, los angeles, memphis, miami, milwaukee, inneapolis, montreal, nashville, new york, new york's, new york city, newark, oakland, ontario, orlando, philadelphia, phoenix, pittsburgh, salt lake, salt lake city, san diego, san francisco, san jose, seattle, st. louis, saint louis, st. paul, saint paul, st. petersburg, saint petersburg, tacoma, tampa, toronto, washington, l a, philly, canada

<CLASS>
classes, class

<CLASS_NAME>
business, business class, first class, coach, economy

<CODE>
code, codes

<CODE_NAME> s, s slash, a p, a p slash, h, f, y, y n, q, q oh, b, q o, s a, a p fifty eight, b h, a p slash fifty seven
<COMPARISON> less than, more than, equal, equal to, same, same as
<CONNECTIONS> connection, connections, combination, combinations, connecting, connecting flights, direct flights, connecting flight
<CONNECTIVE> slash, and, or, either, but, also
<COST> <DIGIT> + <MONEY_UNIT>
<DAY> second, third, fourth, fifth, sixth, seventh, eighth, ninth, tenth, eleventh, twelfth, thirteenth, fourteenth, fifteenth, sixteenth, seventeenth, eighteenth, nineteenth, twentieth, twenty first, twenty second, twenty third, twenty fourth, twenty fifth, twenty sixth, twenty seventh, twenty eighth, twenty ninth, thirtieth, thirty first
<DAY_NAME> day, days, week, weeks, weekday, weekend, week days, week day, weekdays, monday, tuesday, wednesday, thursday, friday, saturday, sunday, during the week, today, yesterday, tomorrow, tonight, monday's, tuesday's, wednesday's, thursday's, friday's, saturday's, sunday's, now, mondays, tuesdays, wednesdays, thursdays, fridays, saturdays, sundays
<DIGIT> oh, zero, one, two, three, four, five, six, seven, eight, nine, ten, eleven, twelve, thirteen, fourteen, fifteen, sixteen, seventeen, eighteen, nineteen, twenty, thirty, forty, fifty, sixty, seventy, eighty, ninety, hundred, thousand, hundreds, thousands, single, double, ones, twos, threes, fours, fives, sevens, eights, nines, tens, twentys, thirtys, fortys, fiftys, sixtys, seventys, eightys, ninetys
<DUMMY> may i, need to, want to, like to, would like to, i would like, i would like to, show me, ineed, i want, i need to, i want to, trying to, try to, the, a, an, please

<FARE> fare, costs, cost, price, fares, airfare, airfares, prices, air fare, air fares, flight fare, flight fares, flight price
<FIRST> first
<FLIGHT> flight, flights, fly, flies, flying
<FLIGHT_DAYS> everyday, daily
<FLIGHT_NUM> flight number, flight numbers
<FLIGHT_NUMBER> <FLIGHT> + <DIGIT>, <AIRLINE_NAME> + <DIGIT>
<FROM> from, departing from, depart from, leave from, leaving, start from, starting from, flying from, fly from, flies from, takeoff from, goes from, go from, take off, takes off, taking off, travel from, departs, depart, departure, departing, leave, leaves, leaving from, takeoff, takeoffs, come from, coming from, comes from
<HOW> how much, how many, how far, how long, how about
<KIND> kind, type, types, kinds, sort
<MANUFACTURER> boeing, mcdonell donglas
<MEAL> meal, meals
<MEAL_DESCRIPTION> dinner, lunch, snack, supper, breakfast, snacks
<MEAN> mean, stand for, meaning, stands for
<MODIFIER> late, early, earliest, earlier, mid, latest, last, later, next, red eye
<MONEY_UNIT> dollar, dollars

<MONTH> january, february, march, april, may, june, july, august, september, october, november, december
<ONE_WAY> one way
<PERIOD> morning, afternoon, evening, day, night, midday, mid-day, breakfast time, lunch time, dinner time, lunchtime, dinnertime, noontime, noon, mornings, nights, midnight, mid-night
<PRE_TIME> before, after, at, around, about, by
<PREP> on, in, between, with, of, for, up, out, under, off
<RESTRICTION> restriction, restrictions
<ROUND_TRIP> round trip, round trip flight, round trip ticket, round trips, and back
<SERVE> serve, served, serves, service, serving
<STATE_CODE> d c
<STATE_NAME> arizona, california, colorado, florida, indiana, michigan, minnesota, missouri, nevada, new jersey, new york, north carolina, ohio, quebec, tennessee, texas, utah, washington
<STOPS> nonstops, nonstop, one stop, at least one stop
<SUPERLATIVE> cheapest, closest, expensive, highest, lowest, shortest, smallest, minimum, maximum, most, least
<TIME> time, times
<TIME_UNIT> a m, p m, o' clock, o'clock, o clock, hour, hours

<TIME_VAULE> <DIGIT> + <TIME_UNIT>, <PRE_TIME> + <DIGIT>
<TO> be there, into, to, arrive to, arriving to, arrives to, arrived to, landing in, land in, fly to, destination, back to, go to, arrive, arrives, arriving, arrived, landed, land, lands, landing, landings, arrival, reach, reaches, reaching
<TRANSPORT> transport, transportation, ground transportation, ground transport
<TRANSPORT_TYPE> rental car, rent a car, need a car, taxi, limousine, train
<VIA> via, by way, stop, stopover, stopovers, stopping, stoping in, stops in, stopover in, stop over in, stopping over in, layover in, laying over in, make a stop, goes through, go through
<WHAT> what're, what's, what
<WHERE> where, anywhere
<WHICH> which
<YEAR> nineteen ninety three

Table C.1: The hand-designed grammar rules in the English ATIS domain.

<ABBREVIATION>
縮寫, 簡寫

<AIRCRAFT>
飛機

<AIRCRAFT_CODE>
d_ten, seventy_three_s, seven_fifty_seven, m_eighty, seven_thirty_three, m_eight_zero, seventy_two_s, d_nine_s, d_c_tens, d_c_ten, d_c_ten

<AIRLINE>
航空, 航空 公司

<AIRLINE_NAME>
a_a, a_c, a_s, c_o, c_p, d_l, f_f, h_p, n_w, n_x, t_w, t_w_a, tower_air, u_a, u_s, u_s_air, w_n, n_w_airline, y_x,川角州, 川角州 航空, 中西 航空, 内陸, 内陸航空, 加拿大 航空, 加拿大 國際航空,加拿大 楓葉 航空, 加航, 西北, 西北 航空, 西南, 西南 航空, 阿拉斯加航空, 西方 航空, 美國 西方 航空, 美國 航空, 國民, 國民 航空, 楓葉 航空, 環球 航空, 聯寺, 聯寺 航空, <AIRLINE_NAME> + 公司, <AIRLINE_NAME> + 航空, <AIRLINE_NAME> + 航空 公司

<AIRPORT>
機場

<AIRPORT_NAME>
b_n_a, b_o_s, b_u_r, b_w_i, c_v_g, d_a_l, d_f_w, d_t_w, e_w_r, h_o_u, i_a_d, i_a_h, j_f_k, l_a_x, l_g_a, love_field, m_c_o, m_i_a, m_k_e, o_r_d, p_i_e, s_f_o, s_l_c, t_p_a, y_y_z, 三藩市 國際, 三藩市 國際 機迷尼加拉瓜 機場, 甘迺迪 機場, 休斯頓 國際 機場, 休斯頓 機場, 多倫多 國際 機場, 安大略 國際 機場, 安大略 機場, 米契爾 國際 機場, 杜勒斯, 杜勒斯 機場, 波士頓 機場, 洛杉磯 國際 機場, 洛杉磯 機場, 紐華克 機場, 奧蘭多 國際 機場, 奧蘭多 機場, 聖彼得堡 機場, 達拉斯 瓦司堡, 雷斯特 皮爾生 機場, 鹽湖城 機場, 皮爾生 機場, <AIRPORT_NAME> + 機場, <AIRPORT_NAME> + 國際 機場

<ALL>
所有, 全部

<AND>
同, 同 埋

<ANY>
任何

<BETWEEN>
之間, 來往, 往來, 至

<BOOK> 訂
<CAPACITY> 載客量
<CITY> 城市
<CITY_NAME> 亞特蘭大, 巴的摩爾, 波士頓, 波班克, 加拿大, 夏洛特, 芝加哥, 辛辛那提, 克里夫蘭, 哥倫布, 達拉斯, 丹佛, 底特律, 瓦司堡, 休斯頓, 印第安納波里, 堪薩斯城, 拉斯維加斯, 長堤, 洛杉磯, 尼加拉瓜, 孟斐斯, 邁阿密, 密耳瓦基, 明尼亞波利斯, 蒙特利爾, 納什維爾, 紐約, 紐約市, 紐華克, 奧克蘭, 安大略, 奧蘭多, 費城, 費尼克斯, 匹茲堡, 鹽湖, 鹽湖城, 聖地牙哥, 三藩市, 聖約瑟, 西雅圖, 聖路易斯, 聖保羅, 聖彼得堡, 他科馬, 坦帕, 多倫多, 華盛頓, 西赤斯特, 西赤斯特城
<CLASS_NAME> 商務, 頭等, 經濟, <CLASS_NAME> + 客位, <CLASS_NAME> + 機位
<CODE> 編號, 號碼
<CODE_NAME> s, s_slash, a_p, a_p_slash, h, f, y, y_n, q, q_oh, b, q_o, s_a, a_p_fifty_eight, b_h
<COMPARISON> 平過, 少過, 多過, 等於, 低過
<CONNECTIONS> 接駁機, 接駁服務, 直航機, 直航

<DAY>
一 日, 二 日, 三 日, 四 日, 五 日, 六 日, 七 日, 八 日, 九 日, 十 日, 十 一 日, 十 二 日, 十 三 日, 十 四 日, 十 五 日, 十 六 日, 十 七 日, 十 八 日, 十 九 日, 二 十 日, 廿 日, 二 十 一 日, 廿 一 日, 二 十 二 日, 廿 二 日, 二 十 三 日, 廿 三 日, 二 十 四 日, 廿 四 日, 二 十 五 日, 廿 五 日, 二 十 六 日, 廿 六 日, 二 十 七 日, 廿 七 日, 二 十 八 日, 廿 八 日, 二 十 九 日, 廿 九 日, 三 十 日, 卅 日, 三 十 一 日, 卅 一 日, 一 號, 二 號, 三 號, 四 號, 五 號, 六 號, 七 號, 八 號, 九 號, 十 號, 十 一 號, 十 二 號, 十 三 號, 十 四 號, 十 五 號, 十 六 號, 十 七 號, 十 八 號, 十 九 號, 二 十 號, 廿 號, 二 十 一 號, 廿 一 號, 二 十 二 號, 廿 二 號, 二 十 三 號, 廿 三 號, 二 十 四 號, 廿 四 號, 二 十 五 號, 廿 五 號, 二 十 六 號, 廿 六 號, 二 十 七 號, 廿 七 號, 二 十 八 號, 廿 八 號, 二 十 九 號, 廿 九 號, 三 十 號, 卅 號, 三 十 一 號, 卅 一 號

<DAY_NAME>
星 期 一, 星 期 二, 星 期 三, 星 期 四, 星 期 五, 星 期 六, 星 期 日, 禮 拜 一, 禮 拜 二, 禮 拜 三, 禮 拜 四, 禮 拜 五, 禮 拜 六, 禮 拜 日, 聽 日, 聽 晚, 今 日, 今 晚, 平 日, 而 家, 第 二 日, 黎 緊 + <DIGIT> + 日

<DIGIT>
零, 一, 二, 兩, 三, 四, 五, 六, 七, 八, 九, 十, 廿, 卅, <DIGIT> + <DIGIT>

<DISTANCE>
距 離

<DOWNTOWN>
市 區, 市 中 心, downtown

<DUMMY>
alright, hi, okay, 請, 請 你, 其 實, 唔 該

<FARE>
收 費, 價 錢, 票 價

<FLIGHT>
航 機, 航 班, 班 機, 機, 客 機

<FLIGHT_DAYS>
每 日

<FLIGHT_NUM>
航 機 編 號, 班 機 編 號, 航 機 號 碼, 航 班 編 號

<FLIGHT_TYPE>
早 班 機, 早 機, 夜 機

<FROM>
由, 起飛, 飛出, 離開, 出發, 起程, 開出

<HOW>
幾多, 幾多 班, 幾多 錢, 幾多 班, 幾 耐, 幾 遠, 幾 錢, 幾多 號, 幾 號, 有 + <HOW>

<MANUFACTURER>
波音, mcdonell donglas

<MEAL>
飛機餐, 菜單, 膳食

<MEAL_DESCRIPTION>
晚餐, 早餐, 晚飯, 午餐, 零食, 小食

<MEAN>
係 乜 野, 點解

<MONEY>
蚊

<MONTH>
一月, 二月, 三月, 四月, 五月, 六月, 七月, 八月, 九月, 十月, 十一月, 十二月

<ONE_WAY>
單程

<OR>
或, 或者

<PERIOD>
上 午, 午前, 午後, 早, 早上, 晏晝, 晚, 晚上, 晨早, 傍晚, 朝, 朝早, 正午, 朝 頭 早, 午夜, <MEAL_DESCRIPTION> + 時間, <MEAL_DESCRIPTION> + 時候

<PERIOD_UNIT>
上晝, 下晝, 夜晚, 中午

<PRE>
前, 之前, 之後, 後, 左右, 大約, 大概, 下個, 下

<QUANT>
架, 班, 張, 一 架, 一 班, 一 張, 一 個, 一 班機

<QUERY>
話 俾 我, 話 我, 列出, 我 想, 我 想要, 我 要, 我 需要, 講 俾 我, 講 我, 乜 野, 乜 野 係, 邊, 有 邊, 邊 班, 有 邊 班, 邊 個, 邊 間, 邊 架, 有 邊 個, 有 邊 間, 有 邊 架, 代表 乜, 搵, <DUMMY> + <QUERY>, <QUERY> + 聽, <QUERY> + 睇 下, <QUERY> + 知道, <QUERY> + 知, <QUERY> + 搭,

<RESTRICT>
限制

<RETURN>
返回, 返 黎, 回程

<ROUND_TRIP>
來回

<SCHEDULE>
時間表

<SERVE>
服務, 提供, 提供 服務

<STATE_CODE>
d_c, d_c 省

<STATE_NAME>
亞利桑那, 亞利桑那州, 加利福尼亞州, 加州, 科羅拉多, 科羅拉多州, 佛羅里達, 佛羅里達州, 印第安納, 印第安納州, 密西根, 密西根州, 明尼蘇達, 明尼蘇達州, 米蘇里, 米蘇里州, 內華達, 內華達州, 新澤西, 新澤西州, 紐約州, 北卡羅來納, 北卡羅來納州, 俄亥俄, 俄亥俄州, 魁北克, 田納西, 田納西州, 得克薩斯, 得克薩斯州, 猶他, 猶他州, 華盛頓州, <STATE_NAME> + 州

<STOPS>
不停 站, 直飛, 中途 站

<SUPERLATIVE>
最 少, 最 近, 最 高, 最 早, 最 遲, 最 後, 最 平, 最 貴, 最 便宜, 最 短, 最 短程, 最 小型, 最 大型, 第一

<TICKET>
機票

<TIME_UNIT>
點, 點鐘, 點 半, 分

<TIMES>
時間

<TO> 去，飛去，飛往，住，飛，飛到，到，到 達，降落，再去，去 到，再 + <TO>
<TRANSPORT> 地面 交通
<TRANSPORT_TYPE> 車，火車，的士
<VALUE_UNIT> 百，千，萬，億
<VIA> 停，中途，中途 停，停留，經，途經，<VIA> + <VIA>

Table C.2: The hand-designed grammar rules in the Chinese ATIS domain.

Appendix D

The Selected Concepts for each Belief Network

Goal: aircraft.aircraft_code		
<AIRCRAFT>	<AIRLINE>	<AIRLINE_NAME>
<AIRPORT>	<AIRPORT_NAME>	<BACK>
<CITY>	<CITY_NAME>	<CITY_NAME_1>
<CITY_NAME_2>	<CITY_NAME_3>	<CLASS_NAME>
<CODE>	<CODE_NAME>	<COMPARISON>
<CONNECTIVE>	<DAY>	<DAY_NAME>
<DIGIT>	<FARE>	<FLIGHT>
<FLIGHT_DAYS>	<FLIGHT_NUM>	<FLIGHT_NUMBER>
<FROM>	<HOW>	<KIND>
<MEAL>	<MEAL_DESCRIPTION>	<MEAN>
<MODIFIER>	<MONTH>	<ONE_WAY>
<PERIOD>	<PREP>	<PRE_TIME>
<ROUND_TRIP>	<SERVE>	<STATE_CODE>
<STATE_NAME>	<STOPS>	<SUPERLATIVE>
<TIME>	<TIME_VALUE>	<TO>
<TRANSPORT>	<TRANSPORT_TYPE>	<VIA>
<WHAT>	<WHICH>	

Goal: airline.airline_code		
<AIRCRAFT>	<AIRLINE>	<AIRLINE_NAME>
<AIRPORT>	<AIRPORT_NAME>	<BACK>
<CITY>	<CITY_NAME>	<CITY_NAME_1>
<CITY_NAME_2>	<CITY_NAME_3>	<CLASS_NAME>
<CODE>	<CODE_NAME>	<COMPARISON>
<CONNECTIONS>	<CONNECTIVE>	<COST>
<DAY>	<DAY_NAME>	<DIGIT>
<FARE>	<FLIGHT>	<FLIGHT_DAYS>
<FLIGHT_NUM>	<FLIGHT_NUMBER>	<FROM>
<HOW>	<KIND>	<MEAL>
<MEAN>	<MODIFIER>	<MONTH>
<ONE_WAY>	<PERIOD>	<PRE_TIME>
<ROUND_TRIP>	<SERVE>	<STATE_CODE>
<STATE_NAME>	<STOPS>	<SUPERLATIVE>
<TIME>	<TIME_VALUE>	<TO>
<TRANSPORT>	<TRANSPORT_TYPE>	<VIA>
<WHERE>	<WHICH>	

Goal: airline.airline_name		
<AIRCRAFT>	<AIRLINE>	<AIRLINE_NAME>
<AIRPORT>	<AIRPORT_NAME>	<BACK>
<CITY>	<CITY_NAME>	<CITY_NAME_1>
<CITY_NAME_2>	<CITY_NAME_3>	<CLASS_NAME>
<CODE>	<CODE_NAME>	<COMPARISON>
<CONNECTIVE>	<COST>	<DAY>
<DAY_NAME>	<DIGIT>	<FARE>
<FLIGHT>	<FLIGHT_DAYS>	<FLIGHT_NUM>
<FLIGHT_NUMBER>	<FROM>	<HOW>
<MEAL>	<MEAL_DESCRIPTION>	<MEAN>
<MODIFIER>	<MONTH>	<ONE_WAY>
<PERIOD>	<PREP>	<PRE_TIME>
<ROUND_TRIP>	<SERVE>	<STATE_CODE>
<STATE_NAME>	<STOPS>	<SUPERLATIVE>
<TIME>	<TIME_VALUE>	<TO>
<TRANSPORT>	<TRANSPORT_TYPE>	<VIA>
<WHAT>	<WHICH>	

Goal: airport.airport_code		
<AIRCRAFT>	<AIRLINE>	<AIRLINE_NAME>
<AIRPORT>	<AIRPORT_NAME>	<BACK>
<CITY>	<CITY_NAME>	<CITY_NAME_1>
<CITY_NAME_2>	<CITY_NAME_3>	<CLASS_NAME>
<CODE>	<CODE_NAME>	<COMPARISON>
<CONNECTIVE>	<DAY>	<DAY_NAME>
<DIGIT>	<FARE>	<FLIGHT>
<FLIGHT_DAYS>	<FLIGHT_NUM>	<FLIGHT_NUMBER>
<FROM>	<HOW>	<MEAL>
<MEAL_DESCRIPTION>	<MEAN>	<MODIFIER>
<MONTH>	<ONE_WAY>	<PERIOD>
<PREP>	<PRE_TIME>	<ROUND_TRIP>
<SERVE>	<STATE_CODE>	<STATE_NAME>
<STOPS>	<SUPERLATIVE>	<TIME>
<TIME_VALUE>	<TO>	<TRANSPORT>
<TRANSPORT_TYPE>	<VIA>	<WHAT>
<WHERE>	<WHICH>	

Goal: airport.airport_name		
<AIRCRAFT>	<AIRLINE>	<AIRLINE_NAME>
<AIRPORT>	<AIRPORT_NAME>	<BACK>
<CITY>	<CITY_NAME>	<CITY_NAME_1>
<CITY_NAME_2>	<CITY_NAME_3>	<CLASS_NAME>
<CODE>	<CODE_NAME>	<COMPARISON>
<CONNECTIVE>	<COST>	<DAY>
<DAY_NAME>	<DIGIT>	<FARE>
<FLIGHT>	<FLIGHT_DAYS>	<FLIGHT_NUM>
<FLIGHT_NUMBER>	<FROM>	<HOW>
<MEAL>	<MEAL_DESCRIPTION>	<MEAN>
<MODIFIER>	<MONTH>	<ONE_WAY>
<PERIOD>	<PREP>	<PRE_TIME>
<ROUND_TRIP>	<SERVE>	<STATE_CODE>
<STATE_NAME>	<STOPS>	<SUPERLATIVE>
<TIME>	<TIME_VALUE>	<TO>
<TRANSPORT>	<TRANSPORT_TYPE>	<VIA>
<WHAT>	<WHICH>	

Goal: class_of_service.class_description		
<AIRCRAFT>	<AIRLINE>	<AIRLINE_NAME>
<AIRPORT>	<AIRPORT_NAME>	<BACK>
<CITY>	<CITY_NAME>	<CITY_NAME_1>
<CITY_NAME_2>	<CITY_NAME_3>	<CLASS_NAME>
<CODE>	<CODE_NAME>	<COMPARISON>
<CONNECTIVE>	<COST>	<DAY>
<DAY_NAME>	<DIGIT>	<FARE>
<FLIGHT>	<FLIGHT_DAYS>	<FLIGHT_NUM>
<FLIGHT_NUMBER>	<FROM>	<HOW>
<MEAL>	<MEAL_DESCRIPTION>	<MEAN>
<MODIFIER>	<MONTH>	<ONE_WAY>
<PERIOD>	<PREP>	<PRE_TIME>
<ROUND_TRIP>	<SERVE>	<STATE_CODE>
<STATE_NAME>	<STOPS>	<SUPERLATIVE>
<TIME>	<TIME_VALUE>	<TO>
<TRANSPORT>	<TRANSPORT_TYPE>	<VIA>
<WHAT>	<WHICH>	

Goal: fare.fare_id		
<AIRCRAFT>	<AIRCRAFT_CODE>	<AIRLINE>
<AIRPORT>	<AIRPORT_NAME>	<BACK>
<CITY>	<CITY_NAME>	<CITY_NAME_1>
<CITY_NAME_2>	<CITY_NAME_3>	<CLASS_NAME>
<CODE>	<CODE_NAME>	<COMPARISON>
<CONNECTIVE>	<COST>	<DAY>
<DAY_NAME>	<DIGIT>	<FARE>
<FLIGHT>	<FLIGHT_DAYS>	<FLIGHT_NUM>
<FLIGHT_NUMBER>	<FROM>	<HOW>
<KIND>	<MEAL>	<MEAL_DESCRIPTION>
<MEAN>	<MODIFIER>	<ONE_WAY>
<PERIOD>	<PREP>	<PRE_TIME>
<RESTRICTION>	<ROUND_TRIP>	<SERVE>
<STATE_NAME>	<SUPERLATIVE>	<TIME>
<TIME_VALUE>	<TO>	<TRANSPORT>
<TRANSPORT_TYPE>	<VIA>	<WHERE>
<WHICH>	<YEAR>	

Goal: fare_basis.fare_basis_code		
<AIRCRAFT>	<AIRLINE>	<AIRLINE_NAME>
<AIRPORT>	<AIRPORT_NAME>	<BACK>
<CITY>	<CITY_NAME>	<CITY_NAME_1>
<CITY_NAME_2>	<CITY_NAME_3>	<CLASS_NAME>
<CODE>	<CODE_NAME>	<COMPARISON>
<COST>	<DAY>	<DAY_NAME>
<DIGIT>	<FARE>	<FLIGHT>
<FLIGHT_DAYS>	<FLIGHT_NUM>	<FLIGHT_NUMBER>
<FROM>	<HOW>	<KIND>
<MEAL>	<MEAL_DESCRIPTION>	<MEAN>
<MODIFIER>	<MONTH>	<ONE_WAY>
<PERIOD>	<PREP>	<PRE_TIME>
<ROUND_TRIP>	<SERVE>	<STATE_CODE>
<STATE_NAME>	<STOPS>	<SUPERLATIVE>
<TIME>	<TIME_VALUE>	<TO>
<TRANSPORT>	<TRANSPORT_TYPE>	<VIA>
<WHAT>	<WHICH>	

Goal: flight.flight_id		
<AIRCRAFT>	<AIRLINE>	<AIRPORT>
<AIRPORT_NAME>	<BACK>	<CITY>
<CITY_NAME>	<CITY_NAME_1>	<CITY_NAME_2>
<CITY_NAME_3>	<CLASS>	<CLASS_NAME>
<CODE>	<CODE_NAME>	<COMPARISON>
<CONNECTIVE>	<COST>	<DAY>
<DAY_NAME>	<FARE>	<FLIGHT>
<FLIGHT_DAYS>	<FLIGHT_NUM>	<FLIGHT_NUMBER>
<FROM>	<HOW>	<KIND>
<MANUFACTURER>	<MEAL_DESCRIPTION>	<MEAN>
<MODIFIER>	<MONTH>	<PERIOD>
<PREP>	<PRE_TIME>	<RESTRICTION>
<ROUND_TRIP>	<SERVE>	<STATE_NAME>
<STOPS>	<SUPERLATIVE>	<TIME>
<TIME_VALUE>	<TO>	<TRANSPORT>
<TRANSPORT_TYPE>	<VIA>	<WHAT>
<WHERE>	<YEAR>	

Goal: flight.flight_number		
<AIRCRAFT>	<AIRLINE>	<AIRLINE_NAME>
<AIRPORT>	<AIRPORT_NAME>	<BACK>
<CITY>	<CITY_NAME>	<CITY_NAME_1>
<CITY_NAME_2>	<CITY_NAME_3>	<CLASS_NAME>
<CODE>	<CODE_NAME>	<COMPARISON>
<CONNECTIVE>	<COST>	<DAY>
<DAY_NAME>	<DIGIT>	<FARE>
<FLIGHT>	<FLIGHT_DAYS>	<FLIGHT_NUM>
<FLIGHT_NUMBER>	<FROM>	<HOW>
<KIND>	<MEAL>	<MEAL_DESCRIPTION>
<MEAN>	<MODIFIER>	<MONTH>
<ONE_WAY>	<PERIOD>	<PREP>
<PRE_TIME>	<ROUND_TRIP>	<SERVE>
<STATE_CODE>	<STATE_NAME>	<STOPS>
<SUPERLATIVE>	<TIME>	<TIME_VALUE>
<TO>	<TRANSPORT>	<TRANSPORT_TYPE>
<VIA>	<WHICH>	

Goal: ground_service.city_code		
<AIRCRAFT>	<AIRLINE>	<AIRLINE_NAME>
<AIRPORT>	<BACK>	<CITY>
<CITY_NAME>	<CITY_NAME_1>	<CITY_NAME_2>
<CITY_NAME_3>	<CLASS_NAME>	<CODE>
<CODE_NAME>	<COMPARISON>	<CONNECTIVE>
<COST>	<DAY>	<DAY_NAME>
<DIGIT>	<FARE>	<FLIGHT>
<FLIGHT_DAYS>	<FLIGHT_NUM>	<FLIGHT_NUMBER>
<FROM>	<HOW>	<KIND>
<MEAL>	<MEAL_DESCRIPTION>	<MEAN>
<MODIFIER>	<MONTH>	<ONE_WAY>
<PERIOD>	<PREP>	<PRE_TIME>
<ROUND_TRIP>	<SERVE>	<STATE_CODE>
<STATE_NAME>	<STOPS>	<SUPERLATIVE>
<TIME>	<TIME_VALUE>	<TO>
<TRANSPORT>	<TRANSPORT_TYPE>	<VIA>
<WHAT>	<WHICH>	

Table D.1: Each Belief Network in the N binary formulation has 50 selected concepts with the highest values of Information Gain relating to the its goal in the English ATIS domain.

<AIRCRAFT>	<AIRCRAFT_CODE>	<AIRLINE>
<AIRLINE_NAME>	<AIRPORT>	<AIRPORT_NAME>
<BACK>	<CITY>	<CITY_NAME>
<CITY_NAME_1>	<CITY_NAME_2>	<CITY_NAME_3>
<CITY_NAME_4>	<CITY_NAME_5>	<CLASS>
<CLASS_NAME>	<CODE>	<CODE_NAME>
<COMPARISON>	<CONNECTIONS>	<CONNECTIVE>
<COST>	<DAY>	<DAY_NAME>
<DIGIT>	<FARE>	<FLIGHT>
<FLIGHT_DAYS>	<FLIGHT_NUM>	<FLIGHT_NUMBER>
<FROM>	<HOW>	<KIND>
<MANUFACTURE>	<MEAL>	<MEAL_DESCRIPTION>
<MEAN>	<MODIFIER>	<MONTH>
<ONE_WAY>	<PERIOD>	<PREP>
<PRE_TIME>	<RESTRICTION>	<ROUND_TRIP>
<SERVE>	<STATE_CODE>	<STATE_NAME>
<STOPS>	<SUPERLATIVE>	<TIME>
<TIME_VALUE>	<TO>	<TRANSPORT>
<TRANSPORT_TYPE>	<VIA>	<WHAT>
<WHERE>	<WHICH>	<YEAR>

Table D.2: The 60 selected concepts of the single Belief Network in the one N-ary formulation using the multiple aposterior strategy in the English ATIS domain.

<AIRCRAFT>	<AIRCRAFT_CODE>	<AIRLINE>
<AIRLINE_NAME>	<AIRPORT>	<AIRPORT_NAME>
<BACK>	<CITY>	<CITY_NAME>
<CITY_NAME_1>	<CITY_NAME_2>	<CITY_NAME_3>
<CLASS>	<CLASS_NAME>	<CODE>
<CODE_NAME>	<COMPARISON>	<CONNECTIONS>
<CONNECTIVE>	<COST>	<DAY>
<DAY_NAME>	<DIGIT>	<FARE>
<FLIGHT>	<FLIGHT_NUM>	<FLIGHT_NUMBER>
<FROM>	<HOW>	<KIND>
<MEAL>	<MEAL_DESCRIPTION>	<MEAN>
<MODIFIER>	<MONTH>	<ONE_WAY>
<PERIOD>	<PRE_TIME>	<PREP>
<RESTRICTION>	<ROUND_TRIP>	<SERVE>
<STATE_CODE>	<STATE_NAME>	<STOPS>
<SUPERLATIVE>	<TIME>	<TIME_VALUE>
<TO>	<TRANSPORT>	<TRANSPORT_TYPE>
<VIA>	<WHAT>	<WHERE>
<WHICH>		

Table D.3: The 55 selected concepts of the single Belief Network in the one N-ary formulation using the maximum aposterior strategy in the English ATIS domain.

<ABBREVIATION>	<AIRCRAFT_CODE>	<AIRLINE>
<AIRLINE_NAME>	<AIRPORT>	<AIRPORT_NAME>
<ALL>	<AND>	<BETWEEN>
<BOOK>	<CAPACITY>	<CITY>
<CITY_NAME>	<CITY_NAME_1>	<CITY_NAME_2>
<CITY_NAME_3>	<CLASS_NAME>	<CODE>
<CODE_NAME>	<COMPARISON>	<CONNECTIONS>
<DAY>	<DAY_NAME>	<DIGIT>
<FARE>	<FLIGHT>	<FLIGHT_NUM>
<FLIGHT_NUMBER>	<FLIGHT_TYPE>	<FROM>
<HOW>	<MEAL_DESCRIPTION>	<MEAN>
<MONTH>	<ONE_WAY>	<OR>
<PERIOD>	<PERIOD_UNIT>	<PRE>
<QUANT>	<RETURN>	<ROUND_TRIP>
<SERVE>	<STATE_CODE>	<STATE_NAME>
<STOPS>	<SUPERLATIVE>	<TICKET>
<TIME_VALUE>	<TIMES>	<TO>
<TRANSPORT>	<TRANSPORT_TYPE>	<VALUE_UNIT>
<VIA>		

Table D.4: The 55 selected concepts of the single Belief Network (BN) in the one N-ary formulation using the maximum aposterior strategy. The BN is modeled for the Chinese ATIS queries.

Appendix E

The Recalls and Precisions of the Goal Identifiers in Macro-Averaging

Goal	Goal index	Frequency (Test set 1993)	Frequency (Test set 1994)
aircraft.aircraft_code	1	0	0
airline.airline_code	2	4	11
airline.airline_name	3	18	6
airport.airport_code	4	16	2
airport.airport_name	5	2	2
class_of_service.class_description	6	5	3
fare.fare_id	7	26	25
fare_basis.fare_basis_code	8	11	5
flight.flight_id	9	301	342
flight.flight_number	10	9	0
ground_service.city_code	11	19	15
OOD	12	35	37

Table E.1: Each goal with its corresponding index and the frequencies in the test sets. The queries mixed with in-domain and OOD goals are extracted before the evaluations. Therefore, the goal aircraft.aircraft_code has zero frequencies in the test sets.

Goal index	# queries (A)	# inferred queries (B)	# correctly inferred queries (C)	Recall (C/A)	Precision (C/B)
1	0	9	0	N/A	0
2	4	21	4	1.00	0.19
3	18	11	11	0.61	1.00
4	16	19	15	0.94	0.79
5	2	2	2	1.00	1.00
6	5	12	5	1.00	0.42
7	26	25	21	0.81	0.84
8	11	12	11	1.00	0.92
9	301	310	290	0.96	0.94
10	9	9	9	1.00	1.00
11	19	16	14	0.74	0.88
12	35	40	24	0.69	0.60

Table E.2: The recalls and precisions of each goal in the N binary formulation using test set 1993 in English ATIS.

Goal index	# queries (A)	# inferred queries (B)	# correctly inferred queries (C)	Recall (C/A)	Precision (C/B)
1	0	7	0	N/A	0
2	11	15	11	1.00	0.73
3	6	6	6	0.61	1.00
4	2	3	1	0.50	0.33
5	2	0	0	0	N/A
6	3	7	2	0.67	0.29
7	25	28	22	0.88	0.79
8	5	7	3	0.60	0.43
9	342	353	330	0.96	0.93
10	0	0	0	N/A	N/A
11	15	18	12	0.80	0.67
12	37	27	10	0.27	0.37

Table E.3: The recalls and precisions of each goal in the N binary formulation using test set 1994 in English ATIS.

Goal index	# queries (A)	# inferred queries (B)	# correctly inferred queries (C)	Recall (C/A)	Precision (C/B)
1	0	9	0	N/A	0
2	4	4	4	1.00	1.00
3	18	19	18	1.00	0.95
4	16	9	9	0.56	1.00
5	2	5	2	1.00	0.40
6	5	13	5	1.00	0.38
7	26	31	22	0.85	0.71
8	11	13	11	1.00	0.85
9	301	304	296	0.98	0.97
10	9	10	9	1.00	0.90
11	19	33	19	1.00	0.58
12	35	34	25	0.71	0.74

Table E.4: The recalls and precisions of each goal in the one N-ary formulation using *multiple* selection strategy in English ATIS test set 1993.

Goal index	# queries (A)	# inferred queries (B)	# correctly inferred queries (C)	Recall (C/A)	Precision (C/B)
1	0	5	0	N/A	0
2	11	11	11	1.00	1.00
3	6	7	6	1.00	0.86
4	2	5	2	1.00	0.40
5	2	2	1	0.50	0.50
6	3	8	2	0.67	0.25
7	25	31	22	0.88	0.71
8	5	9	4	0.80	0.44
9	342	340	328	0.96	0.96
10	0	0	0	N/A	N/A
11	15	21	15	1.00	0.71
12	37	31	22	0.59	0.71

Table E.5: The recalls and precisions of each goal in the one N-ary formulation using *multiple* selection strategy in English ATIS test set 1994.

Goal index	# queries (A)	# inferred queries (B)	# correctly inferred queries (C)	Recall (C/A)	Precision (C/B)
1	0	4	0	N/A	0
2	4	4	4	1.00	1.00
3	18	18	18	1.00	1.00
4	16	8	8	0.50	1.00
5	2	2	2	1.00	1.00
6	5	13	5	1.00	0.38
7	26	20	20	0.77	1.00
8	11	13	11	1.00	0.85
9	301	299	295	0.98	0.99
10	9	9	9	1.00	1.00
11	19	31	18	0.95	0.58
12	35	30	25	0.71	0.83

Table E.6: The recalls and precisions of each goal in the one N-ary formulation using *maximum* selection strategy in English ATIS test set 1993.

Goal index	# queries (A)	# inferred queries (B)	# correctly inferred queries (C)	Recall (C/A)	Precision (C/B)
1	0	2	0	N/A	0
2	11	11	11	1.00	1.00
3	6	7	6	1.00	0.86
4	2	2	1	0.50	0.50
5	2	1	0	0	0
6	3	9	3	1.00	0.33
7	25	26	21	0.84	0.81
8	5	9	4	0.80	0.44
9	342	339	329	0.96	0.97
10	0	0	0	N/A	N/A
11	15	20	14	0.93	0.70
12	37	26	18	0.49	0.69

Table E.7: The recalls and precisions of each goal in the one N-ary formulation using *maximum* selection strategy in English ATIS test set 1994.

Goal index	# queries (A)	# inferred queries (B)	# correctly inferred queries (C)	Recall (C/A)	Precision (C/B)
1	0	0	0	N/A	N/A
2	4	8	4	1.00	0.50
3	18	23	18	1.00	0.78
4	16	10	10	0.63	1.00
5	2	4	2	1.00	0.50
6	5	13	5	1.00	0.38
7	26	24	21	0.81	0.88
8	11	13	11	1.00	0.85
9	301	295	292	0.97	0.99
10	9	9	9	1.00	1.00
11	19	28	17	0.89	0.61
12	35	24	22	0.63	0.92

Table E.8: The recalls and precisions of each goal in the one N-ary formulation using *maximum* selection strategy in Chinese ATIS test set 1993.

Goal index	# queries (A)	# inferred queries (B)	# correctly inferred queries (C)	Recall (C/A)	Precision (C/B)
1	0	1	0	N/A	0
2	11	20	6	0.55	0.30
3	6	89	6	1.00	0.75
4	2	1	1	0.50	1.00
5	2	0	0	0	N/A
6	3	9	3	1.00	0.33
7	25	25	23	0.92	0.92
8	5	9	5	1.00	0.56
9	342	330	319	0.93	0.97
10	0	0	0	N/A	N/A
11	15	17	15	1.00	0.88
12	37	32	26	0.70	0.81

Table E.9: The recalls and precisions of each goal in the one N-ary formulation using *maximum* selection strategy in Chinese ATIS test set 1994.