Understanding Speech Utterances in Mandarin Dialogue System

LIN ZHANG[1], RU-ZHAN LU[2]
[1]Department of computer science, [2]Department of computer science and engineering
[1]Shanghai Maritime University, [2]Shanghai Jiaotong University
[1]Shanghai 200135, [2]Shanghai 200030
[1,2]P.R.China

Abstract: In this paper, we present a mandarin spoken dialogue system—STRQS (Shanghai Traffic Route Querying System), which is used for querying best traffic route between any two locations in Shanghai. A series of language processing strategies is used to understand speech utterances. The understanding processing is done in three steps: First, word segmentation and part-of-speech tagging module splits the utterance into words and labels them with semantic categories. The second step is a robust partial parsing process. Parsing is based on Unification Grammar (UG). An augmented chart algorithm with feature computing is implemented. Finally, the parsed utterance is associated with a semantic interpreter by a frame module. Semantic based analysis method we developed can directly extract information from the output of a speech recognizer, which contains errors and ill-formed components. The testing results demonstrate the robustness of our approach.

KeyWords: spoken dialogue system, partial parsing, language understanding, syntactic rules, feature computing

1 Introduction

The design of robust spoken dialogue system (SDS) is one of the most challenging issues in Natural Language Processing (NLP) [1]. SDS integrates the technology of speech recognition, language understanding, dialogue management, and speech generator. It allows a free and natural human-machine interaction. Many interactive dialogue systems in different languages have been proposed, thanks to the rapid progress in the speech technology and language understanding in those languages [2, 6, 10].

Chinese spoken dialogue system is an active research field with wide applications. Recently, several domain-specific Chinese SDSs have been developed [3, 7, 13]. But the development is not as rapid as that of English, Chinese SDS is facing more problems in language processing, mainly due to the natural specialties of Chinese. Chinese is an isolated language with few inflections, conjugations or other morphological markers. The functional relationships among syntactic constituents are not explicitly expressed in syntactic-morphological forms [4]. This makes Chinese language processing a stumbling block. At the same time, Chinese speech recognition errors make understanding even more acute. Hence, Chinese SDS needs to solve many important issues, such as tolerate speech recognition errors, make understanding from extreme flexibility of oral expressions (like words in disorder, repetitions, ellipsis, anaphora, negation, self-repair, fragments).

Many Chinese computational linguistic researchers have worked on these topics and have presented some approaches to Chinese language understanding. Based on these methods， they have developed some preliminary spoken dialogue systems. Most of the systems are designed with separated parts of speech decoding and language understanding. Some simple applications even make understanding by using method like template matching or keyword extraction [7].

Unlike other Chinese spoken dialogue systems, our system aims at directly extracting semantic information from spoken utterance. In our system, a semantic based analysis approach is implemented to reduce the effect of recognizer errors.

The paper is organized as follows. We first give a brief introduction to our spoken dialogue system—STRQS. Then, we describe the design of our language understanding components. Finally, we conclude the paper with an evaluation of the results.

2 System Architecture

Shanghai Traffic Route Querying System – STRQS is an intelligent SDS. It provides information about the

best traffic route between any two locations in Shanghai. Like the architecture of other English SDSs [10], the system has six main components: speech recognizer, language understanding, dialogue management, application retrieval, response generator and speech output, see Figure 1.

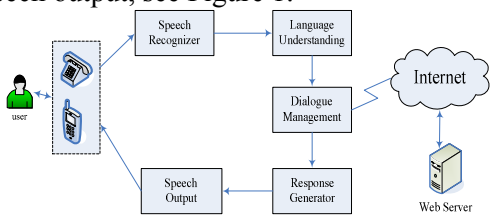

Fig.1 Block diagram of STRQS

The User can talk to STRQS, the speech recognizer module then converts the speech signal into text. The meaning of the utterance is then analyzed by the language understanding component, the system processes the utterance in a serial of steps. The processed results are then passed to the dialogue management (DM) module, which controls the whole interaction progress and instructs the response generator to properly respond to the user. If the user's intention is understood and verified, the dialogue management module will generate a query and submit it to a special transportation query Web server. The query results will be returned back to the DM. Upon receiving the results, the DM module will instruct the response generator to produce response utterance and inform the user by speech generated by the speech synthesizer.

The dialogue corpus of traffic route queries we used in the system is collected from real conversations over telephone line between a guiding agent and clients. It has 104 dialogues (95KB).

2.1 Speech Recognizer

The speech recognizer we used is IBM ViaVoice platform. It is a speaker-independent Chinese dictation system. We further optimize the recognized results by our domain lexicon.

Our domain lexicon is mainly composed of Shanghai geography information database. It contains 5557 domain-related words, such as 1890 road names, 3275 names of well-known buildings/corporations, etc.

2.2 Understanding Mandarin Utterances

This module is responsible for understanding the intentions of the user's utterance. The speech recognition texts are sent through a process pipeline. A series of semantic based approach is implemented in the language understanding module to directly extract meaning from speech.

The language understanding module in our system is composed of three sub-modules: word segmentation and part-of-speech (POS) tagging, a partial syntactic parser and a semantic interpreter, as shown in Figure 2.

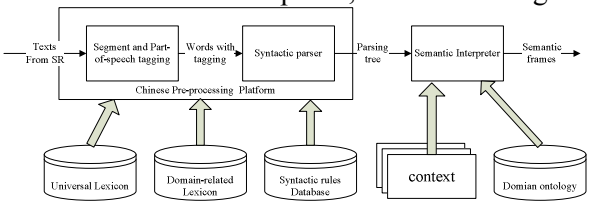

Fig.2 Language understanding procedure

2.2.1 Chinese word segmentation and part-of-speech(POS) tagging

Unlike English, there are no natural word boundaries among Chinese characters in sentences. The first step of language processing is splitting the result text from the speech recognizer into words. It is called automatic word segmentation. Among many mature algorithms to segment sentences [5], we adopted maximum matching algorithm, because it is more efficient in dealing with real-time systems.

After word segmentation, a POS tagging is performed to give each word a category tag. The special approach in our system is that each word belongs to a semantic category. For example, in universal lexicon, verb is a general category, but in our system, verb is divided into 6 semantic categories: V_By, V_From, V_To, V_Ans, V_Aux and V_Wh. The domain-specific words are divided into 29 semantic categories for later processing [8, 9].

A word segmentation and POS tagging result is illustrated by the following example.

Example 1: the user's input: 从上海图书馆到南京路怎么走 (How can I get to the Nanjing road from the Shanghai library)

Segmentation and POS tagging result:

从 V_From /上海图书馆 N_Loc / 到 V_To /南京路 N_Road /怎么 V_Wh /走 C_Vp /

(Here, C_Vp is one of predefined categories, it means the word is a grammar component followed by a verb)

2.2.2 Partial syntactic parser

Syntactic parsing is an indispensable procedure for any

natural language processing. Traditional language understanding methods emphasize on complete syntactic analysis, in which the user's input is analyzed with strict grammars and logic rules. These methods can't fit the phenomena of a spoken language with lots of repetitions, ellipsis, and disordered components. Hence, in spoken dialogue systems, it is necessary to carry out partial parsing instead of complete parsing [11], for traditional complete parsers can't find a complete and correct parse of an utterance. And also, it is more effective since some important meaning of an utterance can be found from partial parsing results --chunks.

Grammar

Our syntactic parser is based on Unification Grammar (UG). UG is a general name of augmented Context-free Grammar (CFG) and can be specified as a set of constraints between feature structures [11].

Formally, the grammar is equivalent to a Context-free Grammar and can be represented by a four-tuple: <V, N, R, S>, where:

V is a finite set of terminal symbols.

N is a finite set of non-terminal symbols.

R is a finite set of production rules, as illustrated in 2.2.2.2.

S is a special terminal, called the start symbol (S∈N).

Unlike CFGs, each non-terminal symbol is not labeled a simple syntactic category (e.g. VP, NP). It is labeled domain-related semantic categories, such as Vp_From, Vp_To. Moreover, in UG, each grammar symbol in the syntactic rules carries a specific feature (called *grammar semantic feature*), which stores its corresponding grammar and semantic information.

A simple example of grammar semantic feature are given below: "淮海路 (Huaihai road)"is labeled by grammar semantic label N_road.

$$N\_Road:\begin{bmatrix} syn: \begin{bmatrix} pos: & N\_Road \\ Value: & 淮海路 \end{bmatrix} \\ sem: \begin{bmatrix} Type: & addr \\ Val: & 淮海路1号 \end{bmatrix} \end{bmatrix}$$

Fig. 3 Example of grammar semantic feature

Grammar rules

Each rule in our system's grammar rules database is consisted of three parts: *a generation formula of CFG*, *feature checking rules*, and *feature unifying rules*. A formal definition of a syntactic rule is a triple tuple: <*Rule*, *Conditions*, *Operations*>

Rule is a generation formula of CFG. There are 128 rules in the syntax rule set of our system.

Conditions are feature checking rules. They restrict the reduction of the syntactic rules

Operations are feature unifying rules. They are used to calculate the final feature values from their constituted phrases

For example, a grammar rule about Vp_To (corresponds to phrase "到淮海路去/to Huaihai road") in the rules database is given here.

Rule:

Vp_To → V_To N_Addr C_Vp

Conditions:

0: V_To.CanFollow = TRUE

Operations:

0: Vp_To.Syn.Pos = VP_To

1: Vp_To.Syn.Value = V_To.Syn.Value + N_Addr.Syn.Value + C_Vp.Syn.Value

2: Vp_To.Sem.Type = Dest

3: Vp_To.Sem.Val = N_Addr.Sem.Val

4: Vp_To.QueryName = N_Addr.Syn.Value

Parsing algorithm

An augmented chart parsing algorithm with feature checking and feature unifying operations is implemented in our system [9].

Parsing is done as the following: at first, the system tries to find a complete parse of the utterance. If it fails (it happens frequently in the spoken language), the system finds all chucks (arcs) in the utterance. Finally, the system needs to choose the best arc set as the reduction result.

A sample parsing example is analyzed as below:

Example 2: input utterance:

从淮海路出发到…恩…外滩怎么走(Start from Huaihai road to…eh…the bund)

Word segment and POS tagging result:

从 V_from / 淮海路 N_Road / 出发 C_Vp / 到 V_To / 恩 INT / 外滩 N_Loc/ 怎么 Wh / 走 C_Vp

Parsing rules: (here feature checking and feature unifying rules are omitted)

Rule1. N_Addr→N_Road

Rule2. Vp_From→V_FromN_Addr

Rule3. Vp_From→V_FromN_AddrC_Vp

Rule4. N_Addr→N_Loc

Rule5. Vp_To→V_ToN_Addr

Rule6. Whp→WhC_Vp

We relaxed the adjacent condition of traditional reduction in chart algorithm. If some interjection or

unknown words between two arcs (such as "eh"), the algorithm can skip these words and reduce them into a new arc (suppose they match under grammar and semantic rules). From the parsing tree (see Figure 4 below) of the above example, we can get the arc "到外滩", although there is an interjection word "恩(eh)" in the middle.

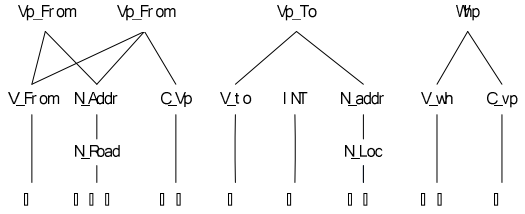

Fig. 4 Reduction result

Each arc, which is used to store the partial results of the reduction, is represented as < c, s, f >. c is a grammar symbol, s is a phrase and f is the corresponding feature of c. So the reduction result of example 2 is represented as four arcs, that is {< Vp_From, "从淮海路", F1>, < Vp_From, "从淮海路出发", F2>, < Vp_To, "到外滩", F3>, < Whp, 怎么走, F4>}, where F1,F2,F3,F4 corresponds to

$$F1:\begin{bmatrix} syn:\begin{bmatrix} pos:Vp\_From \\ Value:从淮海路 \end{bmatrix} \\ sem:\begin{bmatrix} Type:Origin \\ Val:淮海路1号 \end{bmatrix} \end{bmatrix} \quad F2:\begin{bmatrix} syn:\begin{bmatrix} pos:Vp\_From \\ Value:从淮海路出发 \end{bmatrix} \\ sem:\begin{bmatrix} Type:Origin \\ Val:淮海路1号 \end{bmatrix} \end{bmatrix}$$

$$F3:\begin{bmatrix} syn:\begin{bmatrix} pos:Vp\_To \\ Value:到外滩 \end{bmatrix} \\ sem:\begin{bmatrix} Type:Dest \\ Val:外滩 \end{bmatrix} \end{bmatrix} \quad F4:\begin{bmatrix} syn:\begin{bmatrix} pos:Whp \\ Value:怎么走 \end{bmatrix} \\ sem:[Type:\ Query] \end{bmatrix}$$

Next step is to choose the best arc set from all reduction chucks. The method we adopted is given each arc a responding score. Two important factor related to the score is considered. The important one is coverage degree: the more words an arc contains, the higher score it has. The other one is how many words it skipped during reduction processing (mostly the skipped words are interjections, repeats, restarts, etc): the more words it skipped over, the lower score it has. Based on this method, the best arc set of example 2 is {< Vp_From, "从淮海路出发", F2>, < Vp_To, "到外滩", F3>, < Whp, "怎么走", F4>}.

2.2.3 Task model and semantic interpreter

The application task of STRQS is to guide traffic routes. The task is represented as structured frames. Usually, before the system provides information, it needs to collect a specific set of parameters from the user during several conversation turns. These parameters in our system are called *slot values*. A task frame is composed by slots.

A slot can be represented as <*No*, *Nm*, *Vu*, *Sa*>. *No* represent the sequence number, *Nm* is the slot's name, *Vu* is the value of slot, *Sa* is the state of the slot. The initial frame in STRQS is described as below.

{ task = traffic route querying:

$$\begin{bmatrix} < P1, Origin, Null, "not\_reached" > \\ < P2, Dest, Null, "not\_reached" > \\ < P3, Vehi, Null, "not\_reached" > \end{bmatrix}\}$$

Fig.5 Initial semantic frame of STRQS

The procedure of extracting semantics from the output of partial parser is as follows: First, read each arc in the best arc set in order; then get the semantic information of each arc (i.e. Sem.Type and Sem.Val) and use the extracted values to fill the corresponding slots of semantic frame. Figure 6 denotes the semantic interpretation of example 2.

{ task = traffic route querying:

$$\begin{bmatrix} < P1, Origin, 淮海路1号, "complete" > \\ < P2, Dest, 外滩, "complete" > \\ < P3, Vehi, Null, "not\_reached" > \end{bmatrix}\}$$

Fig.6 An instance of semantic frame

2.2.4 Samples of utterance understanding

Since partial parsing method is used in STRQS, it is easy to solve some spoken language phenomena, such as words in disorder, interjection. And also, the system shows robustness in understanding utterance with recognition errors and anaphora.

Example 3: input utterance:从复旦大学到同济大学如何走(Please tell me the way from Fudan university to Tongji university)

Speech recognition text:

从复旦大学同济大学如何走

In this case, the word "到(to)" does not recognized, so the destination information can't be understood. Since partial parsing is used in our system, at least we got the information of the origin location—"复旦大学". The system can get destination information in the next conversation turns as the dialogue develops.

Example 4: user's input: 我怎么乘公交车到南京路 (How can I get to Nanjing road by bus)

Speech recognition text:

我怎么*从*公交车(*from* bus)到南京路

These kind of errors often occurs since the

pronunciation of "乘/cheng/" and "从/cong/" is very similar in Mandarin. Traditional pure syntactic analysis can't find the error, because 从/V and 公交车/ N can be reduced to VP by using syntactic rule (VP→V N). But in our segmentation and POS tagging, 从/V_From and 公交车/N_Vehi are labeled as grammar semantic categories. Then during the next parser processing, "从 公交车" can't be reduced to a VP, because no rule matches. These kinds of recognition errors can then easily found in STRQS.

Considering the speech recognizer always makes a mistake about "乘" and "从", an additional correction rule (see below) is used in our system.

^(Sem.Type.By, Syn.Value.从)＋(Sem.Type.Vehicle)
=>(^change.Syn.Value.乘，^change.Sem.Type.From)
// If the current word is *从* and the semantic type of the following word is *Vehicle*, then change the word "从" into word "乘".

Example 5: user's input: 我想去淮海路… 请问怎样从外滩到那儿(I want to go to Huaihai road…How can I get there from the bund)

For pronoun anaphora in sentences, we also rely on its context to handle it. The simplest anaphora resolution method is to let the pronoun denote the nearest noun phrase before it. In STRQS, a checking condition in anaphora resolution is added into the algorithm. That is, a noun phrase must match the pronoun both in grammar and semantic.

In example 5，there's 2 noun phrases: "淮海路"and "外滩". the pronoun "那儿" follows the word "到", so the semantic type of the pronoun is *Dest* (destination). Among the two noun phrases ("淮海路"and "外滩") in the utterance, "外滩" is the nearest one to the pronoun, but the semantic Type of"外滩"is a starting location. They do not match in semantic. So "那儿" does not point to "外滩" but to "淮海路" which matches the pronoun both in grammar and semantic. This method makes anaphora resolution very simple and effective.

2.3 Dialogue Management

The major goal of the dialogue management is to manage the human computer interaction in a co-operative manner by following the dialogue scenarios. In our system, both the user and the system are allowed to take active roles in a mixed-initiative manner during the conversation. That is to say, the system can take an initiative to ask the value of a key slot, and the user can answer it or say something else. With this mechanism, the system gains flexibility and stability.

The dialogue manager takes the results of the semantic interpreter as input and forms a node-dependent structure that contains all the useful information supplied by the user. It activates the response generator to prompt the user to give extra information or verify the validity of the data provided at the previous node. Next, it queries the Web server, gets the query result, notifies the response generator of the system message.

2.4 Response Generator

The speech generator combines a NLG (Natural Language Generator) and a TTS (Text-to-Speech) synthesizer. The NLG constructs our system's response. It generates a natural language message from the response semantic frame. To allow for flexible language generation, a set of message temples is predefined.

In order to respond in voice, the Chinese TTS system is integrated in IBM ViaVoice Telephony. The speech output of the system is a combination of prerecorded messages and synthesized speech. In this way, the system responds fast and is quite intelligible.

3 Testing

Our system runs on a Pentium IV PC using an industry-standard telephone line interface card (Dialogic D41ESC). It is connected to the Internet. All the modules of the Dialogue Component are programmed in Visual C++.

In the testing, 23 undergraduate students in Shanghai Jiaotong University talked to our system. To ensure that example utterances are true, natural, spoken and versatile, these students know nothing about how the system implemented. Then we collected 68 dialogues, only the user's sentences of the dialogue are used. Next, repeated or similar utterances were eliminated. So, 177 users' oral utterances with different styles were used in the test. The performance of language understanding module then was tested on this corpus.

Table 1 Detailed information of the testing sentences

Type	Example	Number
Simple sentence	怎么去玉佛寺(How to get to Jade Buddha temple)	98

Complex sentences	Contains interjection	到淮海东路，哦不对，是去淮海西路(To Huaihai Dong road, eh, no, to Huaihai Xi road)	9
	Contains anaphora	什么车到那儿？(Which bus shall I take to there)	24
	Multiple incomplete sentences	在复旦，要去同济，坐什么车呀？(Now I'm in Fudan Univ, to Tongji, which bus shall I take)	46

The testing results are presented in table 2.

Table 2 Testing results of language understanding

Cases		Number of cases	Number of correct outputs	Precision rate of cases
Simple sentences		98	91	92.8%
Complex sentences	interjection	9	7	77.7%
	Anaphora	24	21	87.5%
	Multiple incomplete sentences	46	38	82.6%
Sum		177	157	88.7%

The result shows a good coverage rate of the grammar and the robustness of language understanding the ill-formed sentences.

4 Conclusion

In this paper, a mandarin spoken dialogue system is presented. We have emphasis on the spoken language processing module. The difference of our system and other Chinese spoken dialogue systems lies on that we directly extract semantic information from spoken utterance. Since speech recognition sentences unavoidably contain errors, a series of language processing strategies (such as grammar semantic tagging, partial parsing schema, and semantic feature based chart algorithm) are developed to reduce the effect of recognition errors. The testing result demonstrates the robust understanding of our approach.

During the testing analysis, we find that most of errors result from cases that sentences include complex locations, such as "华山路广元路交叉口附近(Near the cross of Huashan road and Guangyuan road)". We also realize that a large portion of errors arise from cases that shallow surface analysis usually fails to really understand underlying meanings. For example, "我赶时间(I'm pressed for time)", which means that the user want to take an express vehicle such as a metro or a taxi, but the system cannot currently understand. These would be our future work.

Acknowledgements

The research work described in this paper is supported by the National '863' Hi-Tech Program under grant number 2001AA114210, P.R.China.

Thanks a lot to Dharmendra Sharma and Wanli Ma (both are scholar of University of Canberra, Australia) for their helpful comments about this paper.

References

[1] V. Pallotta, A. Ballim. Robust Dialogue Understanding in HERALD. *In Proc. of RANLP 2001 - EuroConference on Recent Advances in Natural Language Processing*, Tzigov-Chark, Bulgaria, September, 2001,

[2] B. Pellom, W. Ward, S. Pradhan, The CU Communicator: An Architecture for Dialogue Systems, *In Proc. of ICSLP 2000*, Beijing China, November 2000.

[3] Bor-shen Lin, Lin-shan Lee. Computer-aided Design/Analysis for Chinese Spoken Dialogue Systems. *In: International Symposium of Chinese Spoken Language Processing (ISCSLP),* 2000, pp57-60

[4] Lin-shan Lee, Structural Features of Chinese Language -- Why Chinese Spoken Language Processing is Special and Where We Are, *keynote Speech, 1998 International Symposium on Chinese Spoken Language Processing*, Singapore, 1998, pp1-15.

[5] Kai-ying Liu, *Automated word segmentation and POS tagging of Chinese Texts*, the commercial press, 2000.

[6] J. Allen, et al. A robust system for natural spoken dialogue. *In: Proc. 34th Annual Meeting of the ACL*, 1996. pp62–70.

[7] Y.-F. Huang, F. Zheng,et al. Language understanding component for Chinese dialogue system, *In: Proc. of ICSLP-2000*. Beijing, China, October 2000

[8] Xuan Chen. Chinese Syntactic Analysis System Based on Complex Features. M.S. Dissertation, Shanghai Jiaotong University, 2002

[9] Jia-ju Mao, Rong Guo, Ru-zhan Lu. Chinese spoken language understanding in SHTQS. *Journal of Harbin Institute of Technology (New Series)*, Vol. 12, No. 2, 2005, pp225-230.

[10] James R Glass. Challenges for Spoken Dialogue System. *In: Proc. of IEEE ASRU Workshop*, 1999

[11] Xuedong Huang, et al. *Spoken Language Processing: a guide to theory, algorithm and system development*. Prentice Hall PTR, 2001

[12] C. Huang, P. Xu, et al. LODESTAR: A Mandarin Spoken Dialogue System for Travel Information Retrieval, *EuroSpeech*, Vol 3, 1999, pp1159-1162