A Conversational Agent for Food-ordering Dialog Based on VenusDictate

Hsien-Chang Wang, Jhing-Fa Wang, and Yi-Nan Liu
Institute of Informational Engineering
National Cheng Kung University
No. 1 University Road, Tainan, Taiwan R.O.C.
E-mail: {wangsj, wangjf, liuyn}@server2.iie.ncku.edu.tw

Abstract

In this paper, we introduce a conversational agent which is applied to the food-ordering dialog system. It uses VenusDictate as speech recognition front-end, then understands the semantics of the input sentence by extracting the keywords of the sentence, finally it interacts with the user by speech. The experimental results show that the performance of this agent is good in this application domain.

1. Introduction

The applications of natural language research can be divided into two major classes: text-based applications and dialog-based applications[James 1995]. Text-based applications involve the processing of written text, such as book, newspapers, Internet messages, email messages, and so on. Text-based natural language research is ongoing in applications such as extracting information from message or articles, language translation, summarizing text, etc. On the other hand, dialog-based applications involve human-machine communication which involves spoken language or interaction using keyboards. Important applications include database answering, automated customer service over the telephone, tutoring system, machine controlling, and so on [H. 1992, Ren 1991, Hsien 1997].

Dialog-based systems are quite different from text-based systems. For example, the language used is very different. Also, the system needs to interact with the user in order to maintain a natural, smooth-flowing dialog.

In this paper, a dialog-based food-ordering system is introduced. We build a conversational agent to perform necessary processes of a dialog system, including speech recognition, keyword extraction, intention and syntactic analysis, semantic understanding and proper response generation. We divided our paper into several sections. In Section 2, we brief the architecture of the dialog system. Section 3 is about the corpus collection and analysis. Section 4 introduces our speech recognition front-end -- VenusDictate and keyword extracting. Section 5 describes the analysis of intention and syntax of the input sentence. Section 6 describes how the interactive responding system operates. Section 7 is the experimental results, and we give a conclusion in Section 8.

2. System Architecture

Figure 2.1 shows the architecture of our food-ordering dialog system. The conversational agent plays an important role in our system. We implement this agent by four sub-processes -- speech recognition, keyword extraction, semantics derivation, and interactive response. The flow of a food-ordering dialog would be like this: the customer inputs the ordering sentence via the microphone, then the input speech is recognized by VenusDictate system and produces the candidate syllable lattices. These candidate syllables then passed to keyword extracting unit to acquire the keywords. Those keywords are then used to derive the semantics and thus determine the intention of the customer. Finally, the interactive response system replies proper message to the customer to complete the dialog. The detail processes of each sub-system are described in the following sections.

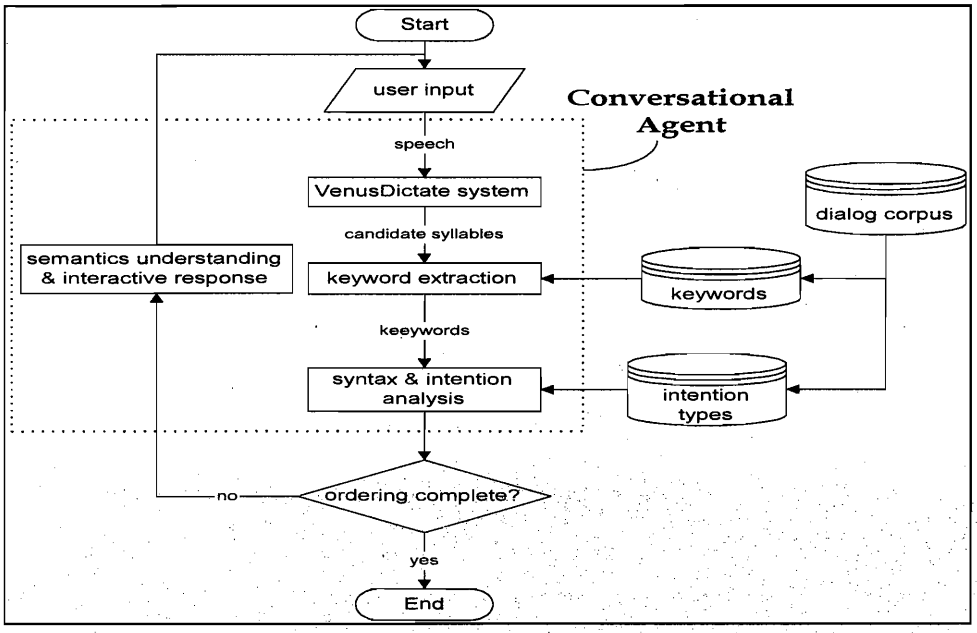

Figure 2.1 Architecture of the Food-ordering Dialog System

3. Dialog Corpus Collection and Analysis

3.1 Collecting Corpus

Dialog corpus can be divided into two major types, they are speech-format and text-format corpus. Speech-format corpora are usually used for training and evaluation of a recognition system. On the other hand, in order to analyze the syntax and semantics of the dialog, what we need is the text-format corpus.

There are two methods to collect dialog corpus. The first one is to simulate the conversation between the boss and the customer. This method has the disadvantage that the corpus will be lack of many situations in a real conversation. Also, sentences generated by such manner will probably tend to be fixed on some special patterns. The second method is to record the conversation in the food store and then transfer them into text-format corpus. We use this approach to collect corpus for our system.

Due to some reasons, we cannot persuade the boss of fast-food stores to participate our task of collecting corpus. So we choose the breakfast stores in the neighborhood of our school to record the conversation between the boss and customers. By this way, we have collected several hundreds of sentences as the corpus. The examples of these conversations can be found in the Appendix.

3.2 Keywords Classification

By analyzing the corpus, we find that some types of words play important roles in a food-ordering dialog system, such as the names of food, amount of drink, and so on. We define these words as keywords and divide them into 11 groups by their meaning. Table 3.1 lists all these 11 **keyword types** in our system. Note that we also define the abbreviation of a certain keyword as a keyword.

Keyword types	Meaning	Examples of keywords
Food	the names of food	三明治(sandwich), 漢堡(hamburger)
Drink	the names of drink	紅茶(black tea), 咖啡(coffee)
Amount	amount of food/drink	一杯(one cup), 兩個(two)
Attribute	modifier of a certain food/drink	冰的(cool), 溫的(warm)
Place	where to eat	這邊用(here), 外帶(to go)
Y_N	positive or negative modality	是的(yes), 不用(no)
Want	ordering something	我要(I want), 給我(give me)
What	ask about something	什麼是(what is),
Have	ask if there exists something	有沒有(do you have)
Price	ask for the price of something	多少(how much), 幾元(how many dollars)
NonKeyword	words without significant meaning	麻煩(would you), 哈囉(hello)

Table 3.1 Keyword types of a food-ordering dialog system

4. VenusDictate System

4.1 Introduction of VenusDictate System

VenusDictate system is a speaker dependent Mandarin word recognition system developed by Institute of Computer Science and Informational Engineering, National Cheng-Kung University [Y.W. 1991, J.S. 1991, S.H. 1990]. It allows user to input speech via microphone and then output the corresponding word candidates.

The role VenusDictate plays in our system is to transfer the input speech into the candidate syllable lattice. These syllables are then further processed to derive the semantics of the input sentence. Since VenusDictate system is now available with the Windows Application Program Interface(API) format, the integration of speech and understanding system can be easily done[J.S. 1994, H.C. 1997].

4.2 Keyword Extracting Using VenusDictate

Determining the keywords of the input sentence is an essential task in a dialog system[R.C. 1995]. In our system, the syllable lattice produced by VenusDictate is used to extract the keywords. Those keywords defined in Table 3.1 are added to the lexicon of VenusDictate, then word matching is performed by VenusDictate.

When we deal with keywords, the length of word matching is important. Consider two keywords named "咖啡(coffee)" and "咖啡奶(coffee milk)", the former is a substring of the latter keyword. If we perform keyword matching without considering the length, we will probably never be able to match the longer keyword "coffee milk", instead, the keyword "coffee" will be matched. To solve this problem, we match the longer keyword first.

When matching keywords with the syllable lattice produced by VenusDictate, the whole syllable lattice is matched first to check if there is any keyword with the same length of the input sentence. If none was matched, the length of the syllable lattice is reduced by one. Those parts of syllable lattice that are matched with keywords are removed from the syllable lattice. This process will continue until the syllable lattice becomes empty. The algorithm of this process is listed below in Algorithm 4.1.

- Input: Syllable lattice with length N.
- Output: Keywords matched.
- Method:

  Step 1. Let $k=N$, if $k=0$ then goto Step 3.

  Step 2. Try to match keyword with length k.

  if success, { $N=N-k$, output keyword, goto Step 1.}

  else, { $k=k-1$, goto Step 2.}

  Step 3. End.

Algorithm 4.1 Extracting keywords from syllable lattice

Those keywords extracted by VenusDictate are passed to the intention and syntax analysis unit for further processing.

5. Intention and Syntax Analysis

Knowing the intention of the customer is important in a dialog system. Also, the syntax is an important information for the system to decide if the input sentence is nonsense. Our system uses an approach that acquires the intention first, then checks whether the input sentence is grammatical legal.

5.1 Intention Types

After analyzing the corpus, the patterns of the ordering sentences can be divided by their meaning into five **intention types**. Those five intention types in our system are shown below.

1. ***S_What***: The sentence that contains the keyword type, ***What,*** has this intention

type. The intention of the customer is to ask about the description of something.

For example: "請問*什麼是*馬來糕？" (What is the Malay-Cake?)

2. ***S_Price***: The sentence that contains keyword type ***Price.*** The intention is to ask the price of something.

   For example: "請問紅茶*多少錢*？" (What is the price of black tea?)

3. ***S_Have***: The sentence which contains keyword type ***Have.*** The intention is to ask the existence of something.

   For example: "請問*有沒有*三明治？" (Do you have sandwiches?)

4. ***S_Want***: The sentence which contains keyword type ***Want***, or contains none of the above keyword types. The intention is to order something.

   For example: "*我要*兩個漢堡，一杯豆漿" (Give me two hamburger and one cup of soybean milk.)

5. ***Y_N***: The sentence that contains keyword type ***Y_N*** representing Yes or NO.

As described above, the five keyword types, ***What, Price, Have, Want, and Y_N***, are important when determining the intention type of an input sentence. If we can find one of these keyword types in the input sentence, we can easily determine the related intention type.

5.2 Syntax Analysis

Once the intention of the input sentence is known, we perform syntactic analysis for this sentence. The structures of ordering sentences can be described by the state transition diagrams as shown in Figure 5.1. In those diagrams, each link represents one of the keyword types in Table 3.1. The starting state is q_0 and the ending state is q_z. The abbreviated state transition diagram S_{order} and $S_{F/D}$ is detailed in Figure 5.2 and Figure 5.3.

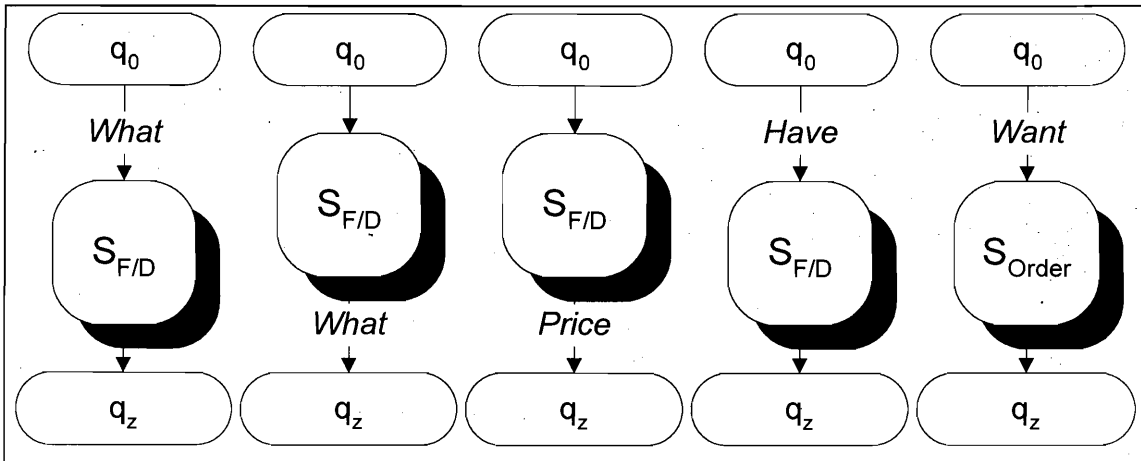

Fig. 5.1a Fig. 5.1b Fig. 5.1c Fig. 5.1d Fig. 5.1e

Figure 5.1a-5.1e Transition diagrams of intention types

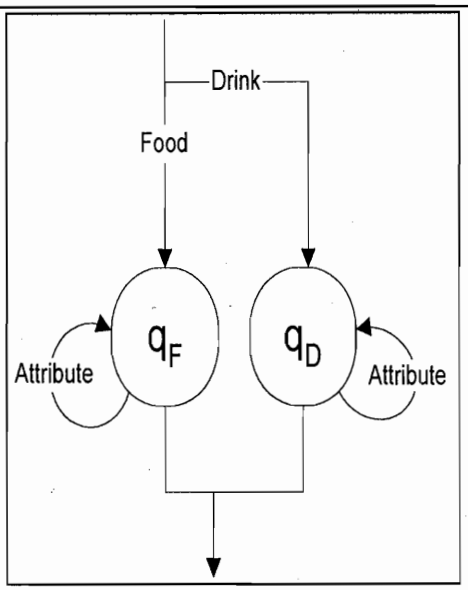

Figure 5.2

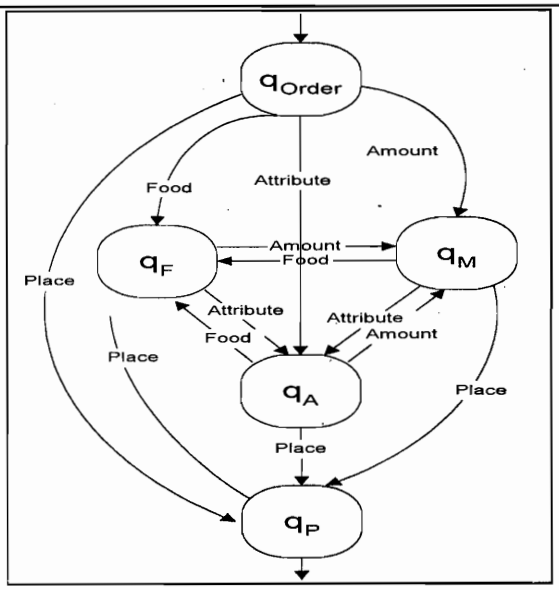

Figure 5.3

Figure 5.2 Detailed state transition diagram of $S_{F/D}$

Figure 5.3 Detailed state transition diagram of S_{order} (Note: This figure shows only ***Food*** keyword type, the figure of ***Drink*** is omitted)

We allow the keyword type ***NonKeyword*** to appear in any place of the input sentence, so we omit them in our state transition diagram to reduce the complexity of the figures.

If an input sentence can be verified by the state transition diagrams, we call it a legal sentence. For those illegal sentences, we will prompt to the user to input again. The following sentence "我要一杯是的多少錢？" (I want a cup of yes how much?") is an illegal sentence since it can not pass our state transition diagrams. For an illegal input sentence, the system asks the customer to input again.

6. Semantic Understanding and Interactive Response

The semantics of each legal input sentence contains its intention type and some extracted keyword types. For example, "Give me a cup of coffee" has the semantics:

"S_Want" + "Amount" + "Drink"

Knowing the semantics of the input sentence, we can generate proper response by considering its intentional type. The response of our dialog system is generated by combining several keywords with phrases. We will describe the response of intention types ***S_What, S_Have, S_Price,*** and ***S_Want*** respectively in the following subsections. Before that, we give some notations used in the response sentence.

- *?F/D*: The name of food or drink mentioned in the customer's ordering sentence (may include amount and attribute).

- *Description(F/D)*: The description of the mentioned food or drink.
- *Price(F/D)*: The price of the mentioned food or drink.

Note that *Description(F/D)* and *Price(F/D)* are stored in the knowledge base.

6.1 Response for Intention Type S_What

If the semantics of the input is ***S_What***, the response sentence will be generated in the form:

The ?(F/D) is Description(F/D).

For instance, the customer may ask: "What is the Malay-Cake?" Then our system will response "The Malay-Cake is *Description(Malay-Cake)*." The *Description(Malay-Cake)* is the description of Malay-Cake and is stored in the knowledge base.

6.2 Response for Intention Type S_Have and S_Price

If the input sentence has intention type ***S_Have***, the response sentence will be:

Yes, we have ?(F/D).

or

No, we do not sell ?(F/D).

If the intention type of the input sentence is ***S_Price***, the response will be:

The ?(F/D) cost Price(F/D).

6.3 Response for Intention Type S_Want

The most complex intention type of the customer is **S_Want**. When ordering, the pattern of the input sentence may vary from person to person. To handle this kind of problem, we define that if the customers want to order something, the dialog should not finish until five keyword types had been input. Those five keyword types are ***Food***, ***Drink***, ***Place***, ***Amount***, ***Attribute***.

If an input sentence lacks of some keyword types, the response system will ask the customer for those items. Consider a food-ordering dialog shown below.

Customer:	Hi, I want soybean milk.
Boss:	How many cups? Cool or Warm?
Customer:	One cup, cool.
Boss:	Do you want some food?
Customer:	No.
Boss:	Do you want to eat in the store?
Customer:	Take-out.

In the process of the interactive dialog, system determines which keyword type is absent, then generates the corresponding query sentence to ask the user to input that keyword type. The dialog will not finish until all five keyword types are all filled. Table 6.1 shows how this is done.

	Food	Drink	Attribute	Amount	Place
customer		soybean milk			
system			ask Attribute	ask Amount	
customer			cool	one cup	
system	ask Food				
customer	No				
system					ask Place
customer					take-away

Table 6.1 Illustration of how interactive response system work.

When the ordering dialog is complete, our system repeats the customer's order and totals the price of food and drink. To make the system more friendly, the response sentences are chosen from some predefined sentences randomly. For instance, in the beginning of the dialog, the greeting of the boss may be "Welcome", "Hello, what do you want", or "What can I do for you?" etc. Furthermore, if the customer has just ordered food, the response sentence may be "Do you want some drink?" or "Take-out?". In this way, we make our response sentence more flexible.

7. Experimental Results

We implement our food-ordering dialog system by Microsoft Visual C++ 4.0, and integrated it with the VenusDictate system. The platform is a Pentium 120 PC with Windows 95 operating system.

Firstly, we test the performance of the single keyword recognition rate of the VenusDictate system. The tester pronounces each keyword of 11 keyword types three times. Then we calculate the Top 1 correct rate. VenusDictate system allows the user to input speech by two pronunciation manners, word-connected and semi-continuous. Our tests include both input methods. The Correct-Rate-I is the recognition rate of word-connected input method, and Correct-Rate-II is that of semi-continuous method. The result is shown in Table 8.1. The recognition rate of the word-connected version is better than that of the semi-continuous version of VenusDictate. The reason is that continuous speech recognition sometimes causes insertion or deletion problems.

	Food	Drink	Amount	Attribute	Place	Y_N	What	Have	Price	Want	Non-keyword	Average correct rate
Correct -Rate-I	95.5	99.7	89.5	83.3	96.4	88.7	93.7	95.7	98.5	92.6	94.6	93.4
Correct -Rate-II	80.5	73.1	84.4	61.1	87.7	73.5	86.7	81.7	88.5	90.2	82.5	80.9

Table 7.1 Single keyword Recognition rate of VenusDictate

Secondly, to test the performance of our system, we randomly choose 50 food ordering dialogs from the corpus to be tested. Since our system allows the user to complete his order in one sentence (***fully***) or in several sentences (***partially***), our test contains these two types of input methods. The result of "fully" and "partially" input method is shown in Table 8.2.

There are two kinds of test performed for both "fully" and "partially" input methods. The first one is the number of success within one trial which is abbreviated as SW1T. It means that the tester successfully orders his food in the first trail via VenusDictate. The second one is SW3T(success within 3 trials) which means the order succeeds within three trials via VenusDictate. Note that the SW3T includes the SW1T ones. From Table 8.2, we find that the correct rate of fully ordering method is poor than that of partially one. The reason is that a fully order contains longer input speech, and may cause more recognition errors.

	testing sentences	SW1T		SW3T	
		# of correct sentences	correct rate	# of correct sentences	correct rate
fully	50	25	50%	40	80%
partially	50	36	72%	46	92%

Table 7.2 Experimental result of our system.

8. Conclusion

In this paper, we describe the implementation of a conversational agent for food-ordering dialog system. We use VenusDictate as the speech recognition front-end, then determine the syntax and intention of the input sentence, finally generate proper response to interact with the customer.

The conversational agent proposed in this paper has a flexible architecture. The speech recognition front-end can be replaced by another speech recognition system, such as a speaker independent recognition system or a recognition system which works over the telephone-network. Also, a text-to-speech system can be easily integrated into this agent.

The collection of dialog corpus is a difficult task which costs much money and manpower. However, in order to establish a practical dialog system, it is an unavoidable important task. We wish that there will be more manpower invested into this task and the collected corpus can be shared.

There are many dialog-based applications, such as automated service over the telephone, tutoring system, etc. With the experience of building this food-ordering system, we hope to develop an automatic or semi-automatic system which can help to transplant from one application domain to another easily.

References

James Allen, Natural Language Understanding, The Benjamin/Cummings Publish Company, INC. 1995

H. Tsuboi and Y. Takebayashi, "A real-time task-oriented speech understanding system using keyword spotting," Proc. ICASSP, pp.197-200,1992.

Ren-Jong Hseu, "Automatic Chinese Telephone Operator Assistant, ACTOA," Proceedings of ROCLING IV, pp.167-191, 1991.

Hsien-C. Wang, Jhing-F. Wang and Din-Y. Liou, "Natural Language Understanding for Telephone Transfer Dialogue", Proceeding of ICCPOL '97, pp. 7-12.

Y.W. Jeng, J. F. Wang, "Large Vocabulary Size Speech Recognition System", master thesis, Inst. of Info. Eng., Natl. Cheng Kong Univ. 1991.

J. S. Shyuu, J. F. Wang, "A Speaker Independent Continuous Mandarin Digit Recognition System", master thesis, Inst. of Info. Eng., Natl. Cheng Kong Univ. 1991.

S. H. Lee and H. J. Lee, "A Unification-Based Approach for Chinese Inquiry Sentences Processing," Proceedings of ROCLING III, pp.441-466, 1990.

J. S. Shyuu, J. F. Wang and C. H. Wu, "An user friendly interface, high reliability, large vocabulary Mandarin speech recognition system", National human interface speech communication conference III, China, 1994.

H. C. Wang, J. S. Shyuu, and J. F. Wang, "Natural Language Understanding Based on VenusDictate", Proceeding of CSIA '97, pp. 185-190.

R. C. Rose, "Keyword detection in conversational speech utterances using hidden Markov model based continuous speech recognition," Computer Speech and Language 9, 309-333, 1995.