A Multilingual-supporting Dialog System Using a Common Dialog Controller

Yunbiao XU*, Masahiro ARAKI**, Yasuhisa NIIMI**
Department of electronics & information science,
Kyoto Institute of Technology
Matsugasaki, Sakyo-ku, Kyoto, 606-8585, Japan
yunbiao@vox.dj.kit.ac.jp*, {araki,niimi}@dj.kit.ac.jp**
Tel: +81-75-724-7477(NIIMI) Fax: +81-75-724-7400

ABSTRACT

It is well known that a speech dialog system can be regarded as an integration of a speech interface which runs in the front end and a dialog controller which runs in the back end. The former is obviously language-dependent while the later could be language-independent relatively. This paper describes an approach to constructing a multilingual spoken dialog system. In this approach we extended a dialog controller for Japanese to a language-independent one and combined it with a Chinese speech interface. Experimental result shows that the proposed approach is effective in constructing quickly a multilingual-supporting dialog system using a common dialog controller.

1. INTRODUCTION

With an increase of computer system performance, a spoken dialogue system becomes more and more mature. Now, many applications, such as traffic information query (ATIS) for English[1], travel information accessing (VOTIRS 2.0) for Chinese[2], tourist information service (SDSKIT-3) for Japanese[3], have been developed. Although all of these dialog systems have excellent performance, almost all of them are designed only for one single particular language, and development works have separately taken despite of whether the oriented tasks are the same or not. So, this situation lengthens the development period for a particular task, and therefore reduces the development efficiency.

Can we construct a dialog system for a certain domain to serve more people who speak in different languages? In other words, if we can construct a dialog system in which different speech interfaces share the same dialog controller, then a world-wide supporting spoken dialog system may be built in shorter period than before.

In fact, now, there are two large-scale research projects have been progressing in the world wide. One is to develop a Universal Network Language(**UNL**) and to apply its corresponding DeConverter and EnConverter to realize communication and information exchange between different languages[4]. UNL is a language for information exchange and transfer over the Internet[5]. The 6 official languages of the United Nations, Arabic, Chinese, English, French, Russian and Spanish, and the other 11 languages are involved in the initial stage of this project. Although this project is text-based, but the combination of the speech technique and UNL will make it possible to communicate verbally between people who speak in different languages. The another project is **C-STARIII [6],** a multilingual-supporting spoken dialog platform at hotel reservation task, which supports 7 core popular languages, Chinese, English, Japanese, Italian, French, German and Korean. One can use her/his own native language to ask C-STARIII to do something when this platform is completed after 4 years.

This paper describes an approach to constructing a multilingual-supporting dialog system. For this purpose we extended a dialog controller developed for Japanese to a language-independent one, [MSDSKIT-1][7], and combined it with a Chinese speech interface through an case-frame converter. Figure 1 shows the idea which we think is very similar to that of DeConverter and EnConverter in UNL.

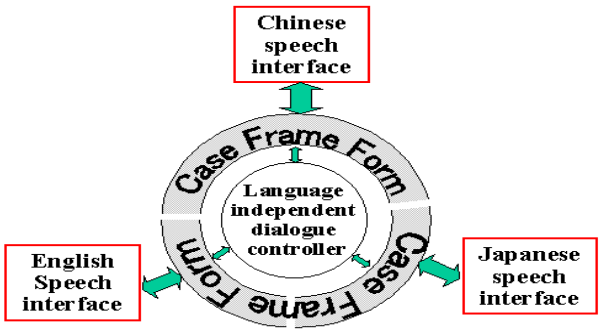

Figure 1. A multilingual-supporting dialog system

In the following, we will briefly illustrate MSDSKIT-1 in section 2, then present the speech recognizer in section 3, the language parser in section 4, the case-frame converter in section 5, and finally, the conclusion and the future work will be presented in section 6.

2. SYSTEM OVERVIEW

In general, a conventional spoken dialog system can be regarded as an integration of two relative independent parts, just as shown in figure 2; one is a speech interface and another is a dialog controller. In MSDSKIT-1 message is conveyed from one to another through case

frame forms. It talks with a user in languages such as Chinese/English/Japanese, while another language is adopted in the back end. We call the former language as ***target language***, the latter one as ***pivot language.***

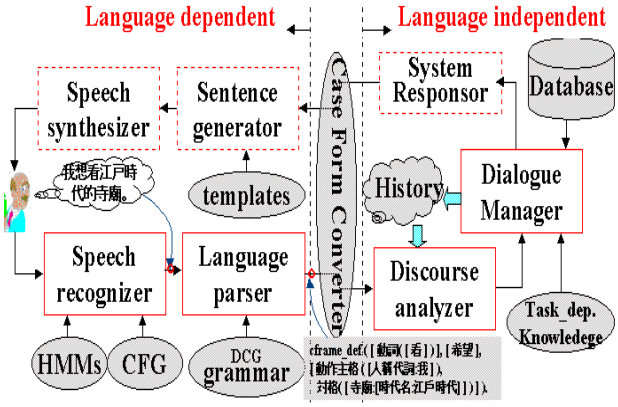

Figure 2. A multilingual dialog system using a common dialog controller

Based on a set of acoustic HMM models and CFG grammar rules, the ***speech recognizer*** converts user's utterance into a string of words as the input of the language parser. Then, the ***language parser*** analyzes the syntactic structure of its input string of words to produce a corresponding case fame form which is based on the framework of a case grammar. Each case frame form contains four kinds of information,

① a word which is used to reflect whether the utterance is an affirmative answer[Yes/Ok], a negative answer[No], or an ambiguous reply[Then], and so on, we call such words as ***Lead-word***s.

② ***a list of case elements*** which reflect the relation between the main verb word and each case element. Each case element is constructed by a set of pairs of case marker and its attribute value.

③ ***modality information*** ,

④ ***a main verb word***,

The language-independent part consists of a discourse analyzer, a dialog manager and a system responsor(response generator). The ***dialog manager*** works under the frame-driven control scheme. Since it has been reported in detail elsewhere[9], it is outlined here. The frame-driven control scheme is based on the fact that topics in a goal-oriented dialog tend to move according to a task-dependent tree structure. So, for a given task, we can construct a few related topics as a frame, which might appear in dialogs on the task domain under consideration. We call it a topic frame hereafter. A topic frame is a set of slot. A slot is described by a slot name, a value which is initially empty, a method for filling the slot, and an act taken by the dialog manager after the slot has been filled. These are described in the pivot language.

A set of the topic frames could form implicitly a tree structure (called a static topic tree) by allowing slots to be filled with some other topic frames. We assume topics in a dialog move along this tree structure as a dialog proceeds. Thus a dialog forms a subtree of the static topic tree. We call this tree a dynamic topic tree.

The dialog manager builds up a dynamic topic tree as a dialog between a user and the system proceeds, and points a node of the tree as a current node which corresponds to the current state of the dialog. The dialog manager searches in the tree equidistantly from this current node to locate topics of user's utterance or searches in depth-first way to find an unfilled slot.

The ***discourse analyzer*** uses four kinds of information above mentioned and the discourse history to extract the dialog act and topic/focus of the user's utterance[8,9]. The discourse history is described by a transition network of dialog acts and a dynamic topic tree. Dialog acts and topics are extracted through bottom-up and top-down analyses[8]. Bottom-up candidates for dialog acts and topics are decided by applying a set of rules specially designed to the semantic interpretation of an utterance. Top-down candidates for dialog acts and topics are decided by using the current state of the dialog history. Then, the logical ANDs between the bottom-up and top-down candidates are taken to decided the dialog act and topic of user's utterance.

After inverse transform for the case frame form from the pivot language to the target language, the ***sentence generator*** generates a suitable sentence. At finally, the ***speech synthesizer*** synthesizes speech as its output.

After introducing briefly MSDSKIT-1, now, we will concentrate on describing the speech recognizer, the language parser and the case frame form converter for Chinese.

3. CHINESE SPEECH RECOGNIZER

3.1 Chinese acoustic models

It is well known that Mandarin Chinese is a monosyllabic and tonal language. The initial-final (I/F) structure is a characteristic of Chinese syllable. 22 initials and 38 finals are used to spell 408 toneless Chinese syllables.

There are several types of the Speech Recognize Unit (SRU) of Chinese. The context-independent monophone model uses the 22 initials and the 38 finals as SRU. Syllable-based model uses 408 toneless syllables or 1300 tone-syllables as SRU. The context-dependent triphone model considers both the left and the right context of the initials/finals. Because the syllable-based model has solved the co-articulation phenomenon existing between the initial and the final within a syllable, and the triphone model has solved further the co-articulation phenomenon existing between not only within a syllable but also cross syllables, the performance of it are better than that of the monophone model. But, because of sparseness of a speech data for training acoustic HMM models, we selected the 22 initials and the 38 finals as SRUs to train monophone HMMs. We think that the whole performance of the speech recognizer would be raised by adopting CFG to

compensate the deficiencies in the monophone acoustic models.

The speech data described in this paper is taken from China National Hi-Tech Project 863. It consists of 1560 different sentences which are divided into 3 groups called A, B and C. 30 male speech data(10 per group) with total of 15600 sentences are used to train HMMs. 3 male speech data(1 per group) with total of 1560 sentences are used to make an open test. The detail experiment conditions was shown in table 1.

Table 1: experiment condition

Sampling frequency	16 KHz
Window type	Hamming
Window Size	25 ms
Window shift rate	10 ms
Parameter type: MFCC_E_D_N_Z	MFCC ΔMFCC+Δpower
Vector component size	12 + 12 + 1 = 25

Some comparison experiments by HTK V3.0 have done to determine a suitable size of states and mixtures per state. The syllable accuracy is show in table 2. Finally we select the models with 5-emitting states and 6-mixtures per state as the acoustic models of MSDSKIT-1.

Table 2: Estimations by HTK

Size of mixture per state	Syllable accuracy
5-emitting states 1-mixture	43.25%
5-emitting states 3-mixtures	56.87%
5-emitting states 6-mixtures	61.95%

In MSDSKIT-1, we adopted JULIAN (developed by Kyoto University) as a baseline speech recognizer in which a context free grammar (CFG) can be used as a language model (LM).

3.2 CFG grammar rules

Writing accurately CFG grammar rules for a particular task is a time-consuming job. For the purpose of simplification, after defining and collecting the Part-of-Speech(POS) dictionary, we drafted some networks, such as figure 3, to reflect the possible utterance's structures

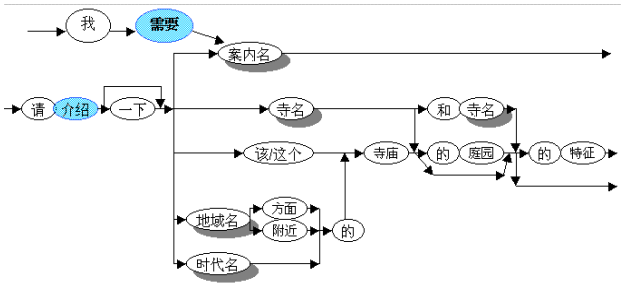

Figure 3. A grammar Network

Such networks are very helpful to shorten writing CFG grammar rules. In MSDSKIT-1, 77 rules with total of 61 POS categories and 392 words are collected in current implementation. Then we make test experiments by speaking 56 pre-defined sentences representing a broad variety of differing degrees of difficulty. The test experiments are performed in a general lab room. The raw speech of a male was directly input through DAT-Link machine, the average number of sentences recognized correctly is 53, i.e. 94.6%.

4. CHINESE LANGUAGE PARSER

The language parser in MSDSKIT-1 performs the syntactic analysis for input word strings to determine their semantic representation which are expressed with a frame-based data structure called a case frame form.

In such a Chinese language parser, we use the definite clause grammar (DCG) based on some preparation such as

① a BNF dictionary which groups all words in POS,
② a case marker table which groups semantically all noun words listed in the BNF dictionary.
③ a case frame definition table which describes all possible user's utterances.

4.1 Case Frames

The semantic interpretation of an utterance is performed based on the case grammar in which the meaning of a sentence is represented by a case frame associated with a main verb of that sentence. A case frame is described by a set of slots, each indicating one of such relations between a verb and a noun phrases, like an agent, object and instrument. Noun phrases included in an utterance are assigned to some slot of the case frame based on case markers of the noun phrases. Thus, the semantic interpretation of an utterance is represented by a list of three terms, a main verb, case information with slots filled, modality information including the style of an utterance.

Here, we show the model of the case frames: *(V, M, C)* where *V* denotes the value of verb, *M* denotes the modality information which contains one to several elements, i.e.,

$$M = [M_0, M_1, ..., M_n].$$

C denotes the case elements with one or several slots, in which each slot indicates the relation between the verb *V* and the noun phrase filling this slot, i.e.,

$$C = case\_id_1(case_1), ..., case\_id_k(case_k).$$

It also can be wrote as

$$C = [C_0, C_1, ..., C_k].$$

For example, the sentence of "I need the sight-seeing guidance" can be represented by the following case frame.

case_frame(Verb(***need***), Mood(***statement***),
[Actor([Who:***I***]),
Object([NameOfFuction:***the sight-seeing guidance***])
])

Here, "Actor" and "Object" indicate the relation to the verb word "need". "Who" and "NameOfFunction" specify the case markers of the phrases to fill case elements. Each message included in the case frame are very helpful to the language parser. We will introduce

again what a role such message will play in the successor section.

4.2 DCG grammar rules

The language parser uses the definite clause grammar (DCG) to combine the CFG used in the speech recognizer with case frames above mentioned. The characteristic of DCG compared with CFG is that DCG allows us to use enhancement items[10], such as following sample,

```
ss(Lead, V, M,C)→
    Lead,{ member(Lead,[yes,no,then,and,ok,non]) },
    s(V,M,C),
    EndSymbol.
s( V, [ M0,…,Mn], [C0,…,Ck] )→
    case_element[ C0 ],
    { case_frame( V,[M0,…,Mn],[C0,…,Ck] ) },
    verb[ V,M0,…,Mn ],
    …
    case_element[ Ck].
```

The language parser finally output its parse result with a frame-based data structure called a case frame form which is described by a set of words as following.

```
Sentence: I need the sight-seeing guidance/我需要觀光導遊。
ss([[non],[Verb(需要)],[陳述], [Actor([Who我]),Object([NameofFuction:觀光導遊])]])
+— s([Verb(需要)],[陳述], [Actor([Who我]),Object([NameofFuction:觀光導遊])])
|   +— case_element(Actor([Who我]))
|   |    +— NP([Who:我])
|   |       +— Noun([Who],[Who(我)])
|   |          +— 我
|   +— Verb([Verb(需要)], [陳述])
|   |    +— Verb(Verb(需要))
|   |       +— 需要
|   +— case_element(Object([NameofFuction:觀光導遊]))
|        +— NP([NameofFuction:觀光導遊])
|           +— Noun([NameofFuction],[NameofFuction(觀光導遊)])
|              +— 觀光導遊
+— EndSymbol([endSent])
    +— 。
```

5. TRANSFORM of CASE FRAME FORM

Outputs of the language parser are case-frame forms. These are composed of four kinds of information, such as a Lead-word, a main verb word, a modality information and a case list, just as described in section 2. These are used directly as its input by the successor discourse analyzer to determine the topic/focus and act type of user's utterance.

In order to use the whole source of the language-independent part of MSDSKIT-1 without any change. It is necessary to translate the language parser's output case-frame form described by the target language into the corresponding case-frame form described by the pivot language. Inverse translation is also necessary before the output case-frame form of the dialog controller is conveyed to the sentence generator. Now, MSDSKIT-1, based on a set of transformation rules, three types of transformations are realized: the main verb word and the modality information, the case marker and its corresponding attribute values, the Lead word which specifies the "Yes/No/Then/…" message.

Based on above efforts, combining with a frame-system created for the succeeding discourse analyzer and the dialogue manager which can be run language-independently, the whole spoken dialog system is implemented now by our research group. More details about the dialog controller are given by [3,9].

6. CONCLUSION and FUTURE WORK

This paper described an approach to constructing a multilingual-supporting dialog system which has been applied in MSDSKIT-1. In such an approach, a case-frame form converter, is very similar to the EnConverter and the DeConverter in UNL in principle, is embedded into a conventional dialog system between the speech interface which is language-dependent and the dialog controller which is language-independent. MSDSKIT-1, currently, operates in two languages, Chinese and Japanese. In future, we will try to improve the templates for the sentence generator, improve the language-independent part in order to increase performance of MSDSKIT-1.

REFERENCES

[1] Price, P., "Evalutaion of Spoken Language system: The ATIS Domain", Proc. of EuroSpeech'93, 1993, Berlin, Germany

[2] ZHANG Xin, C. HUANG, ZHAO S.,HUANG T., "Spoken Language Understanding in Spoken Dialogue system for Travel Information accessing", Proc. of ISCSLP'98, pp.294-298, 1998, Singapore

[3] Niimi Y., Takinaga N., Nishimoto T., "Dialogue Management in a Spoken Dialogue System, SDSKIT-3", Proc. of SPECOM'98, pp.91-96, 1998, Russia

[4] http://www.unl.ias.unu.edu/MissionUNLP.html,

[5] http://www.cse.iitb.ernet.in:8000/~pb/UNL.htm

[6] http://www.zaobao.com.sg/special/newspapers/2000/pages7/dzkjb311200.html (in Chinese), OR. http://www.kyoto-np.co.jp/kp/topics/2000oct/12/07.html (in Japanese)

[7] Yunbiao XU, Masahiro ARAKI, Yasihisa NIIMI, "A Multilingual Spoken Dialog System", Proc. Of ISCSLP2000, pp.175-178, China

[8] Y. Niimi, N. Takinaga, T.Nishimoto, "Extraction of the Dialog Act and the Topic from Utterances in a Spoken Dialog System", Proc. of ICSLP'98, pp.2079-2082, 1998, Australia

[9] Yasuhisa Niimi, Tomoki OKU, Takuya Nishimoto and Masahiro ARAKI, "A task-independent dialogue controller based on the extended frame-driven method", Proc. of ICSLP'2000, Volum I, pp.114-117, 2000, China

[10] Gerald Gazdar, Chris Mellish, "Natural Language Processing in Prolog – An Introduction to Computational Linguistics", Addison-Wesley Publishing Company, ISBN 0-201-18053-7