DIALOG MANAGEMENT FOR SPEECH OUTPUT FROM CONCEPT REPRESENTATION

Yoichi Yamashita and Riichiro Mizoguchi

The Institute of Scientific and Industrial Research, Osaka University
8-1, Mihogaoka, Ibaraki-shi, Osaka, 567 Japan
yama@ei.sanken.osaka-u.ac.jp

ABSTRACT

Building a general speech interface independent of Intelligent Performance Systems (IPS) is an important technique for efficient construction of spoken dialog systems. This paper describes dialog management for speech output in MASCOTS-II which is a dialog manager in such an interface system. MASCOTS-II carries out dialog management based on two mechanisms; SR-plan analyzes the dialog structure of utterances and TPN (Topic Packet Network) models the topic transition in dialog. In speech output of this framework, output messages from IPS are described with concept representation which is based on the case structure and the phrase patterns. MASCOTS-II receives concept representation from IPS and modifies it according to the dialog context. Modified concept representation is converted into synthetic speech by SOCS, a concept-to-speech conversion system in the interface. MASCOTS-II extracts emphatic words to add prominence so that the user pays his/her attention to important information in the IPS utterance. It also has a mechanism for restoring missing words in IPS utterance to notify the recognition result to the user. The concept representation is rewritten in terms of phrase pattern named custom template, if it is possible.

I. INTRODUCTION

There arises a strong demand to make use of speech for communication with various Intelligent Performance Systems (IPS), such as data base, guidance system, and so on, in place of keyboards and visual display terminals. Three key components, speech recognition, speech synthesis, and dialog management are indispensable to the human-to-machine communication through spoken language. An general speech interface system including these three components makes it easy to construct an IPS which communicates with its users through speech. If the interface is designed dependently on an IPS, it must be reconstructed for every IPS. As the phoneme recognition technique and the speech synthesizer are, of course, independent of the IPS, there are many procedures common to the dialog management as well as speech recognition and the speech synthesis. To avoid the wasteful reconstruction, the generality of the speech interface system is a crucial issue. It means that its architecture should not be dependent on the IPS.

In the speech interface, Dialog Manager (DM) plays many important roles both for speech input to IPS and output from IPS. The dialog processing for speech input includes the understanding of dialog structure, expectation of the user utterance, restoration of ellipsis, and so on. For speech output, DM must determine the expression appropriate to the dialog context and extract some features useful to speech synthesis. In addition, the speech recognition errors by the system and the misunderstanding of the synthetic speech by the user may be inevitable in the spoken dialog. These dialog processings should be carried out not by IPS but by DM because they are commonly required to most dialogs. Then, it is very important to consider the dialog processing separately from the IPS and to discuss the generality of the DM, that is, the independency of IPS. This paper describes the dialog manager in a general speech interface system focussing on the dialog processing for speech output.

II. Dialog Manager in a General Speech Interface

2.1 Outline of Speech Output

From the viewpoint mentioned above, the authors have designed a framework of an interface for speech input and output, shown in Fig.1, which is independent of the IPS. This interface system is composed of three major components; the speech understanding component, the speech synthesis component and the dialog manager, which are named SPURT-I (SPeech Understanding system with Rule-based and Topic-directed architecture)[1], SOCS (Speech Output from Case Structure representation)[2], and MASCOTS-II (MAnagement System for COnversation using Twin-stacks and Sr-plan), respectively. The previous MASCOTS was developed only for the speech understanding, particularly the language processing subsystem in SPURT-I[3]. We extend MASCOTS to cover dialog processing for speech output from IPS, keeping its generality. The new version of MASCOTS is called MASCOTS-II.

In speech output from IPS, the representation scheme which is used to describe output messages is an crucial issue. IPS should decide only what to say and commit the determination of the surface spoken language expression to

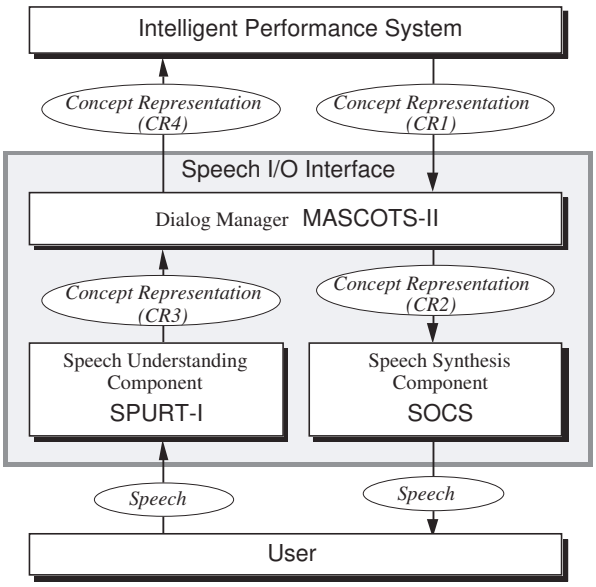

Fig.1 A block diagram of speech interface.

```
s1( hajimeru([$object(lesson1),$persuasive]) ).
      ('start')
  lesson1( $modify( chiri, benkyo ) ).
              ('geography')  ('lesson')
s2( $because(soviet1,seiikusuru([$actor(kome ),$negative])) ).
                          ('grow')              ('rice')
  soviet1( $is( soren, $modify( $p_prominent(samui), kuni ) ).
         ('the Soviet union')                          ('cold') ('country')
s3( $why(seiikusuru([$actor(kome),$place(iran)]),[$p_interrogative]) ).
              ('grow')             ('rice')          ('Iran')
```

Fig.2 Examples of concept representations.

the speech interface system. DM adjusts the messages according to the dialog context and the speech synthesis component extracts some features necessary for prosody generation from message descriptions. If the messages are expressed in the natural language text, such procedures in the interface system become difficult. Therefore, the representation form with appropriate abstraction level is very important for describing messages for speech output. The authors defines the representation based on the case structure and the phrase patterns as concept representation, considering the interaction between IPS and the speech synthesis component[3]. Fig.2 shows examples of concept representation. IPS generates a message to the user using concept representation, and MASCOTS-II modifies it according to the dialog context. Then, SOCS converts modified concept representation into synthetic speech. As shown in Fig.1, both the message generated by IPS and the input to SOCS are described with the same concept representation scheme. The contents of them may be slightly different, because MASCOTS-II modifies the concept representation according to the dialog context. Two concept representations are referred to as CR1 and CR2, respectively. In the proposed framework, the user utterance analyzed is also expressed by using the concept representation and referred to as CR3.

2.2 Dialog management Model

DM must understand dialog structure and identify the relation among utterances in the dialog to carry out various dialog procedures for speech input and output. Several methods have been proposed for dialog understanding[4]. The basic mechanism of MASCOTS-II for the dialog understanding was based on SR-plan and TPN.

2.2.1 SR-plan[3]

Although there are various types of dialogs in the real world, human-to-machine communication through speech is a goal-oriented dialog. The goal-oriented dialogs consists of the utterance pairs whose components are the stimulus (requirement, question, order and so on) and the response to it. To grasp characteristics of the dialog structure, we introduce the concept of SR-plan, which is abbreviation of Stimulus-Response plan. SR-plan is a plan which represents a interaction of stimulus and response. To expect the user utterance form, we classified SR-plans into 17 categories according to the type of interaction. There are two type of SR-plan: one is the user SR-plan which are for dealing with the interactions constructed by user stimuli and IPS responses, another is the system SR-plan for IPS stimuli and user responses. Two stacks, the system and user stacks, manage SR-plan activated by the utterance of IPS and the user, respectively.

2.2.2 TPN (Topic Packet Network)

Topic transition in a dialog can be viewed as a sort of hierarchy. In other words, the dialog begins with a rough topic and goes down to more specific ones, and often comes back to the previous topic explicitly and implicitly. In addition, a topic can be specialized into limited descendants. We believe that the relation between a topic and its descendant topics is often independent of individual dialogs. We define this set of the descendant as a TP (Topic Packet). Each TP has a super topic. When a topic moves to the specific one as dialog goes on, another descendant TP whose super topic is the previous topic is invoked and one of the topics in it is activated. And, the descendant TP is dynamically linked to the previous topic in the ancestor TP. If topic comes back to the previous super topic, the descendant TP is detached. These manipulation of links for TPs results in a dynamic changing network which is called TPN (Topic Packet Network) and TPN gives a model for the topic transition in dialogs.

In general, the identification of topic is an important problem for the dialog understanding, the expectation of user utterance for speech input, and the determination of expressions appropriate to the dialog context for speech output. Various types of knowledge, such as meaning of words, syntax, background domain, and so on, are necessary to the topic identification. Unless the dialog manager has knowledge on topic transition, that is a very tough task for the dialog manager. MASCOTS-II introduces TPN in order to model topic transition and to identify the current topic. Each topic description in TP includes the keywords and the system templates which are used for identification of the topic activated by the user and IPS utterance, respectively.

III. FUNCTIONS OF DIALOG MANAGER FOR SPEECH OUTPUT

The speech interface system should take on the job of dealing with the preceding utterances in dialog to determine the surface sentence and to generate the prosodic parameters of synthetic speech. Thus, IPS does not need to take account of the dialog context for message description and only have to decide what to say. MASCOTS-II carries out several dialog procedures for speech output as follows.

3.1 Extraction of Emphatic Words

In dialog, it is frequently observed that some words in sentences are emphatically presented in the sense of prosody and/or verbal expression. If those words are not emphasized when the sentence is uttered isolatedly, that is, without dialog context, the emphasis of them is considered as being intrinsically caused by the dialog context. The words emphasized dependently on dialog-context are classified into the following three categories:

(1) words which are replaced with other words appeared in the preceding utterance,
(2) words which is selected among semantically similar words, and
(3) words of the major point in an answer.

These dialog-dependent emphases should be indicated in the concept representation not by IPS but by DM of the interface system. Therefore, DM should have mechanisms for extraction of words to be emphasized. To this end, MASCOTS-II makes use of the dialog history and SR-plan.

(1) Emphases based on the replacement of words

When there is something wrong in the user utterance, IPS may correct it by replacing a wrong word with a correct one in the IPS utterance. In general, the user should pay his/her attention to the words which was replaced with other words in the preceding utterance. MASCOTS-II decides such words should be emphasized by comparing the IPS utterance with the last user utterance. Fig.3 (a) shows an example of this procedure. In this figure, u, s and s' corresponds to the utterance in CR3, CR1 and CR2, respectively. Comparing the IPS utterance s with the user utterance u, MASCOTS-II can easily find out that '*ikebukuro-iki*' was replaced by '*shinjuku-iki*', and '*shinjuku-iki*' was

U: *Ikebukuro-iki no chikatetsu ni noru no desuka?*
('May I take a subway for Ikebukuro?')
u: *noru*([$obj($modify(*ikebukuro-iki*,*chikatetsu*)),$interrogative])
'take' 'for Ikebukuro' 'subway'
s: *noru*([$obj($modify(*shinjuku-iki*,*chikatetsu*)),$imperative])
'for Shinjuku'
s': *noru*([$obj($modify($p_prominent(*shinjuku-iki*),*chikatetsu*)),$imperative])
S: ***Shinjuku-iki*** *no chikatetsu ni notte kudasai.*
('Please take a subway for Shinjuku.')
(a)

U: *Osaka to Kyoto no dochira ga Tokyo ni chikai desuka?*
('Which is nearer to Tokyo, Osaka or Kyoto?')
u: $is($or(Osaka,Kyoto), *chikai*, [$subject(Tokyo),$interrogative])
'near'
s: $is(Kyoto, *chikai*, [$subject(Tokyo)])
s': $is($p_prominent(Kyoto), *chikai*, [$subject(Tokyo)])
S: ***Kyoto*** *ga Tokyo ni chikai desu.*
('Kyoto is nearer to Tokyo.')
(b)

U: *Chikatetsu wo doko de oriru no desuka?*
('Where shall I get off the subway?')
u: *oriru*([$obj(*chikatetsu*),$place(*doko*),$interrogative])
'get off' 'where'
s: *oritu*([$obj(*chikatetsu*),$place(Hibiya),$imperative])
s': *oriru*([$obj(*chikatetsu*),$place($p_prominent(Hibiya)),$imperative])
S: ***Hibiya*** *de chitaketsu wo orite kudasai.*
('Please get off the subway at Hibiya.')
(c)

Fig.3 Extraction of emphatic words.

marked as an emphatic word. That results in attaching a prosody operator '$p_prominent' to the IPS utterance, and s is changed into s'. SOCS converts s' into synthetic speech with prosodical emphasis for '*shinjuku-iki*' according to the '$p_prominent' operator.

(2) Emphases based on semantically similar words

Semantically similar words often appear in the parallel relationship in the utterance, and they are typically described by the '$and' and '$or' templates in the concept representation. If the user utterance contains '$and' or '$or' templates and the next IPS utterance refers to one of the words in that templates in the concept representation, then MASCOTS-II judges that IPS selects a word among semantically similar words and decides it should be emphasized. MASCOTS-II easily carries out this procedure by comparing two utterances in the concept representation. Fig.3 (b) shows an example of this case. The user utterance contains 'Osaka' and 'Kyoto' which are placed in parallel relation by the '$or' template, and IPS refers only to 'Kyoto'. MASCOTS-II extracts 'Kyoto' and attaches '$p_prominent' to s.

(3) Emphases based on the major point in an answer.

In general, the IPS's answer to the user WH-question contains the information the user wants. MASCOTS-II extracts such information, that is, the major point of IPS answer as the emphatic words. MASCOTS-II can easily know which user utterance the IPS responds to and what type of requirements the user does, using SR-plans mentioned above. MASCOTS-II utilizes the user SR-plan <UP-ASK-COMPONENT> which is one of user SR-plans to ask objects of action. To identify the major point of the IPS answer. In this SR-plan, some keywords are associated with the case and indicate the information the user wants, shown in Fig.4. Top of keywords in this figure says that '$actor' is the major point of the IPS utterance if '$actor(*dare*)' is found in the user utterance. Here, '*dare*' is a keyword for the major point identification. Fig.3 (c) is an example of an emphatic word which is extracted in this manner. The user utterance (u) activates SR-plan <UP-ASK-COMPONENT>, MASCOTS-II finds the keyword '$place(*doko*)' and knows that the case information '$place' in the concept representation

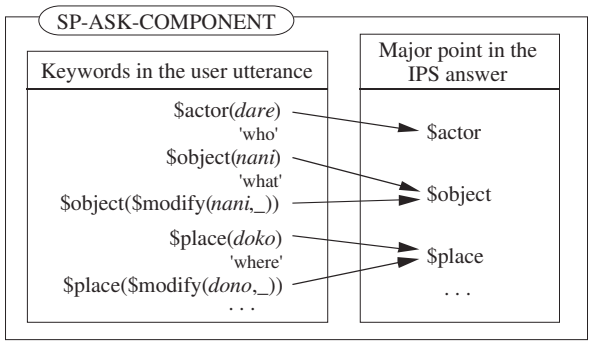

Fig.4 The major point of the system answer.

is the major point of the next IPS answer. After receiving the IPS message (s), MASCOTS-II judges that it is the response to u using activated SR-plan. Then, '$place(Hibiya)' is marked as to be emphasized and s is modified into s'.

3.2 Restoration of Missing Words in the Utterance

In general, there are inevitable recognition errors for the user input speech when the communication between human and machine is carried out through speech. The correct communication needs the mechanism which informs the user of the recognition results of his/her speech in an implicit or explicit way. Otherwise, the user may receive the unexpected wrong information from the IPS. For example, imagine the following dialog.

User: *Byooin wa doko desu ka*?
('Where is the hospital?')
System: *Kono saki no koosaten no migigawa desu.*
('The right side of the next intersection.')

In this dialog, if '*byooin* (hospital)' is incorrectly recognized as '*biyooin* (barber)', the user is told by IPS the wrong place and is not conscious of that. To avoid this situation, the notification of the recognition result to the user is necessary.

There are two types of notification; explicit and implicit one. When recognition score of a word is lower than the threshold, MASCOTS-II generates an explicit notification utterance in order to verify or ask the word in the user utterance. For example, if the score of the first word is low in the above dialog, MASCOTS-II says

Biyooin desu ka? ('Barber?').

When the score of a word which is the second best candidate is close to that of the best one though the score of the top one is high, an implicit notification is carried out. In the above dialog, if the second '*byooin*' gives a close score of the top '*biyooin*', MASCOTS says

Biyooin wa Kono saki no koosaten no migigawa desu.
('The barber is on the right side of the next intersection.')

Of course, judgment that the notification should be carried out or not, is dependent on the speech recognition performance and it is another crucial issue to be discussed. Anyway, the dialog manager should have a mechanism of such notification.

In this section, the restoration of words for the implicit notification is described. The implicit notification is achieved by the repeatedly presenting words which appeared in the last user utterance. If the IPS utterance includes all such words, the restoration is not necessary. However, for designing a general DM, special attention should be paid for that IPS may omit some words. Therefore, MASCOTS-II has mechanisms for restoring words, which are divided into two types according to the IPS utterance. First, when the IPS utterance is an answer to the last user question, this

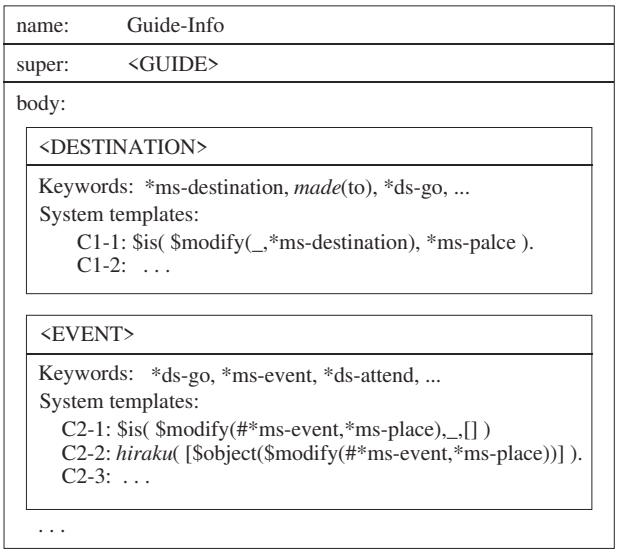

Fig.5 The system templates in TP.

U: *onsei-kenkyuukai e ikitai no desu ga.*
('I'd like to go to onsei-kenkyuukai.')
u: *iku*([$obj(*onsei-kenkyuukai*), $wish]).
'go'
s: $is(*kaijo*, *doko*, [$interrogative]).
'place' 'where'
s': $is($modify($focus(*onsei-kenkyuukai*),*kaijo*), *doko*, [$interrogative]).
S: *onsei-kenkyuukai no kaijo wa doko desu ka?*
('Could you tell me where the onsei-kenkyuukai is held?')

Fig.6 The restoration of missing words in the IPS

is an easy task for MASCOTS-II. Because the user utterance is described with the concept representation, MASCOTS-II easily finds what case information is missing in the IPS utterance of CR1. MASCOTS-II can restore missing words into the IPS utterance of CR2 just only comparing the concept representations of two utterances after the identification of the utterance pair using SR-plan.

Second, when the IPS utterance is a question, MASCOTS-II uses the TP to restore missing words. The system templates for the IPS utterance are associated with each topic in the TP to grasp topic change by IPS, mentioned in 2.2.2. In the system templates, information on focussed words is embedded in. Fig.5 shows an example of the TP. In this figure, C1-x and C2-x is patterns which matches with the IPS utterance when IPS changes the topic. Matching succeeds if the pattern is included in the IPS utterance. Atoms whose name begins with '#' are focussed information. It means that they are potentially omitted out of the IPS utterance. Even if focussed information is missing, matching between the pattern and the IPS utterance succeeds. When MASCOTS-II finds out that focussed information is missing and those words which match with its word category appeared in the last user utterance, MASCOTS-II restores focussed information. Fig.6 shows this operation. The IPS utterance s matches with templates C2-1 in consideration of missing of '*ms-event', and the topic change into <EVENT> is recognized by MASCOTS-II. MASCOTS-II searches a word which matches with '*ms-event' in u and finds out that it is '*onsei-kenkyuukai*'. Then, MASCOTS-II adds '*onsei-kenkyuukai*' along with the indicator '$focus' to the IPS utterance in order to notify the recognition results to the user. The concept representation

ACT([CASES,$place(*doko*),$interrogative]) =>
$where(ACT([CASES]))
dokode ... ka?
('Where ...?')

ACT([CASES,$interrogative,$persuasive]) =>
$offer(ACT([CASES]))
... to iunowa ikaga desu ka?
('How about ...?')

Fig.7 Examples of rewriting rules using custom templates

s results in s'.

3.3 Use of the Custom Templates

In SOCS, the custom templates are prepared for the frequently used expression. As they describe the prosodic pattern using Prosody Modification Functions (PMFs) and the generated sentence pattern, the use of them in the concept representations can synthesize more natural speech. IPS does not necessarily know what custom templates it can use when it generates output concept representation (CR1). MASCOTS-II rewrites the concept representation generated by IPS in terms of custom templates, if it is possible. For example, a concept representation,

hiraku([$object(*kaigi*),$place(*doko*),$interrogative])
'hold' 'conference' 'where'

can be rewritten into

$where(*hiraku*([$object(*kaigi*)]).

To do this, MASCOTS-II has the knowledge base of rewriting rules shown in Fig.7.

IV. CONCLUSIONS

Speech output using concept representation and dialog procedures of MASCOTS-II are described. MASCOTS-II is a dialog manager in the general speech interface system and deals with the procedures related to the dialog context by modifying the concept representation generated by IPS. In order to generate high quality synthetic speech adapted to dialog contexts, MASCOTS-II attaches extra indicators to the concept representation and SOCS only converts it into synthetic speech. For a dialog manager, understanding of dialog is an important issue. MASCOTS-II uses SR-plan for analyzing the dialog structure and TPN for topic transition. These mechanisms contribute sophisticated dialog management for speech input, such as expectation of next user utterance and extraction of user's intention[3]. MASCOTS-II including dialog management for speech input is under development for the dialog of information service on location and its evaluation will be a future work.

Reference

[1] Hori M., Tsujino K., Mizoguchi R. and Kakusho O.: "A Speech Understanding System SPURT-I: Performance Evaluation with a 1000-word Vocabulary," Proc. of third WESTPAC, Shanghai, China, pp.779-782 (1988).
[2] Yamashita Y. and Mizoguchi R.: "SOCS: A Speech Output System from Concept Representation," Proc. ICASSP92, Vol.2, 76.9, pp.69-72 (1992).
[3] Yamamoto T., Ohta Y., Yamashita Y. and Mizoguchi R.: "Dialog Management System MASCOTS in Speech Understanding System," Proc. ICSLP90, Kobe, pp.1301-1304 (1990).
[4] Litman D.J. and Allen J.F.: "A Plan Recognition Model for Subdialogues in Conversations," Cognitive Science, 11, pp.163-200 (1987).