Corpus-based Discourse Understanding in Spoken Dialogue Systems

Ryuichiro Higashinaka and **Mikio Nakano** and **Kiyoaki Aikawa**[†]
NTT Communication Science Laboratories
Nippon Telegraph and Telephone Corporation
3-1 Morinosato Wakamiya
Atsugi, Kanagawa 243-0198, Japan
{rh,nakano}@atom.brl.ntt.co.jp, aik@idea.brl.ntt.co.jp

Abstract

This paper concerns the discourse understanding process in spoken dialogue systems. This process enables the system to understand user utterances based on the context of a dialogue. Since multiple candidates for the understanding result can be obtained for a user utterance due to the ambiguity of speech understanding, it is not appropriate to decide on a single understanding result after each user utterance. By holding multiple candidates for understanding results and resolving the ambiguity as the dialogue progresses, the discourse understanding accuracy can be improved. This paper proposes a method for resolving this ambiguity based on statistical information obtained from dialogue corpora. Unlike conventional methods that use hand-crafted rules, the proposed method enables easy design of the discourse understanding process. Experiment results have shown that a system that exploits the proposed method performs sufficiently and that holding multiple candidates for understanding results is effective.

[†]Currently with the School of Media Science, Tokyo University of Technology, 1404-1 Katakuracho, Hachioji, Tokyo 192-0982, Japan.

1 Introduction

For spoken dialogue systems to correctly understand user intentions to achieve certain tasks while conversing with users, the dialogue state has to be appropriately updated (Zue and Glass, 2000) after each user utterance. Here, a *dialogue state* means all the information that the system possesses concerning the dialogue. For example, a dialogue state includes intention recognition results after each user utterance, the user utterance history, the system utterance history, and so forth. Obtaining the user intention and the content of an utterance using only the single utterance is called *speech understanding*, and updating the dialogue state based on both the previous utterance and the current dialogue state is called *discourse understanding*. In general, the result of speech understanding can be ambiguous, because it is currently difficult to uniquely decide on a single speech recognition result out of the many recognition candidates available, and because the syntactic and semantic analysis process normally produce multiple hypotheses. The system, however, has to be able to uniquely determine the understanding result after each user utterance in order to respond to the user. The system therefore must be able to choose the appropriate speech understanding result by referring to the dialogue state.

Most conventional systems uniquely determine the result of the discourse understanding, i.e., the dialogue state, after each user utterance. However, multiple dialogue states are created from the current dialogue state and the speech understanding results corresponding to the user utterance, which leads to ambiguity. When this ambiguity is ignored, the discourse understanding accuracy is likely to decrease. Our idea for improving the discourse understanding accuracy is to make the system hold multiple dialogue states after a user utterance and use succeeding utterances to resolve the ambiguity among dialogue states. Although the concept of combining multiple dialogue states and speech understanding results has already been reported (Miyazaki et al., 2002), they use intuition-based hand-crafted rules for the disambiguation of dialogue states, which are costly and sometimes lead to inaccuracy. To resolve the ambiguity of dialogue states and reduce the cost of rule making, we propose using statistical information obtained from dialogue corpora, which comprise dialogues conducted between the system and users.

The next section briefly illustrates the basic architecture of a spoken dialogue system. Section 3 describes the problem to be solved in detail. Then after introducing related work, our approach is described with an example dialogue. After that, we describe the experiments we performed to verify our approach, and discuss the results. The last section summarizes the main points and mentions future work.

2 Discourse Understanding

Here, we describe the basic architecture of a spoken dialogue system (Figure 1). When receiving a user utterance, the system behaves as follows.

1. The speech recognizer receives a user utterance and outputs a speech recognition hypothesis.

2. The language understanding component receives the speech recognition hypothesis. The syntactic and semantic analysis is performed to convert it into a form called a *dialogue act*. Table 1 shows an example of a dialogue act. In the example, "refer-start-and-end-time" is called the *dialogue act type*, which briefly describes the meaning of a dialogue act, and "start=14:00" and "end=15:00" are add-on information.[1]

[1] In general, a dialogue act corresponds to one sentence. However, in dialogues where user utterances are unrestricted, smaller units, such as phrases, can be regarded as dialogue acts.

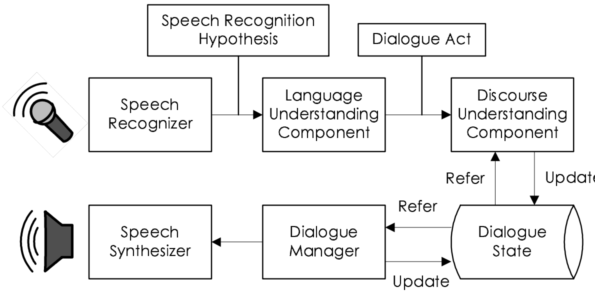

Figure 1: Architecture of a spoken dialogue system.

3. The discourse understanding component receives the dialogue act, refers to the current dialogue state, and updates the dialogue state.

4. The dialogue manager receives the current dialogue state, decides the next utterance, and outputs the next words to speak. The dialogue state is updated at the same time so that it contains the content of system utterances.

5. The speech synthesizer receives the output of the dialogue manager and responds to the user by speech.

This paper deals with the discourse understanding component. Since we are resolving the ambiguity of speech understanding from the discourse point of view and not within the speech understanding candidates, we assume that a dialogue state is uniquely determined given a dialogue state and the next dialogue act, which means that a dialogue act is a command to change a dialogue state. We also assume that the relationship between the dialogue act and the way to update the dialogue state can be easily described without expertise in dialogue system research. We found that these assumptions are reasonable from our experience in system development. Note also that this paper does not separately deal with reference resolution; we assume that it is performed by a command. A speech understanding result is considered to be equal to a dialogue act in this article.

In this paper, we consider *frames* as representations of dialogue states. To represent dialogue states, plans have often been used (Allen and Perrault, 1980; Carberry, 1990). Traditionally, plan-based discourse understanding methods have been implemented mostly in keyboard-based dialogue systems,

User Utterance	“from two p.m. to three p.m.”
Dialogue Act	[act-type=refer-start-and-end-time, start=14:00, end=15:00]

Table 1: A user utterance and the corresponding dialogue act.

although there are some recent attempts to apply them to spoken dialogue systems as well (Allen et al., 2001; Rich et al., 2001); however, considering the current performance of speech recognizers and the limitations in task domains, we believe frame-based discourse understanding and dialogue management are sufficient (Chu-Carroll, 2000; Seneff, 2002; Bobrow et al., 1977).

3 Problem

Most conventional spoken dialogue systems uniquely determine the dialogue state after a user utterance. Normally, however, there are multiple candidates for the result of speech understanding, which leads to the creation of multiple dialogue state candidates. We believe that there are cases where it is better to hold more than one dialogue state and resolve the ambiguity as the dialogue progresses rather than to decide on a single dialogue state after each user utterance.

As an example, consider a piece of dialogue in which the user utterance “from two p.m.” has been misrecognized as “uh two p.m.” (Figure 2). Figure 3 shows the description of the example dialogue in detail including the system’s inner states, such as dialogue acts corresponding to the speech recognition hypotheses[2] and the intention recognition results.[3] After receiving the speech recognition hypothesis “uh two p.m.,” the system cannot tell whether the user utterance corresponds to a dialogue act specifying the start time or the end time (da1,da2). Therefore, the system tries to obtain further information about the time. In this case, the system utters a backchannel to prompt the next user utterance to resolve the ambiguity from the discourse.[4] At this stage, the system holds two dialogue states having different intention recognition results (ds1,ds2). The next utterance, “to three p.m.,” is one that uniquely corresponds to a dialogue act specifying the end time (da3), and thus updates the two current dialogue states. As a result, two dialogue states still remain (ds3,ds4). If the system can tell that the previous dialogue act was about the start time at this moment, it can understand the user intention correctly. The correct understanding result, ds3, is derived from the combination of ds1 and da3, where ds1 is induced by ds0 and da1. As shown here, holding multiple understanding results can be better than just deciding on the best speech understanding hypothesis and discarding other possibilities.

S1	:	what time would you like to reserve a meeting room?
U1	:	from two p.m. [uh two p.m.]
S2	:	uh-huh
U2	:	to three p.m. [to three p.m.]
S3	:	from two p.m. to three p.m.?
U3	:	yes [yes]

Figure 2: Example dialogue.
(S means a system utterance and U a user utterance. Recognition results are enclosed in square brackets.)

In this paper, we consider a discourse understanding component that deals with multiple dialogue states. Such a component must choose the best combination of a dialogue state and a dialogue act out of all possibilities. An appropriate scoring method for the dialogue states is therefore required.

4 Related Work

Nakano et al. (1999) proposed a method that holds multiple dialogue states ordered by priority to deal with the problem that some utterances convey meaning over several speech intervals and that the understanding result cannot be determined at each interval end. Miyazaki et al. (2002) proposed a method combining Nakano et al.’s (1999) method and n-best recognition hypotheses, and reported improvement in discourse understanding accuracy. They used a metric similar to the concept error rate for the evalu-

[2] In this example, for convenience of explanation, the n-best speech recognition input is not considered.

[3] An intention recognition result is one of the elements of a dialogue state.

[4] A yes/no question may be an appropriate choice as well.

[System utterance (S1)]
"What time would you like to reserve a meeting room?"
[Dialogue act] [act-type=ask-time]
[Intention recognition result candidates]
1. [room=nil, start=nil, end=nil] (ds0)

↓

[User utterance (U1)]
"From two p.m."
[Speech recognition hypotheses]
1. "uh two p.m."
[Dialogue act candidates]
1. [act-type=refer-start-time,time=14:00] (da1)
2. [act-type=refer-end-time,time=15:00] (da2)
[Intention recognition result candidates]
1. [room=nil, start=14:00, end=nil]
   (ds1, induced from ds0 and da1)
2. [room=nil, start=nil, end=14:00]
   (ds2, induced from ds0 and da2)

↓

[System utterance (S2)] "uh-huh"
[Dialogue act] [act-type=backchannel]

↓

[User utterance (U2)]
"To three p.m."
[Speech recognition hypotheses]
1. "to three p.m."
[Dialogue act candidates]
1. [act-type=refer-end-time, time=15:00] (da3)
[Intention recognition result candidates]
1. [room=nil, start=14:00, end=15:00]
   (ds3, induced from ds1 and da3)
2. [room=nil, start=nil, end=15:00]
   (ds4, induced from ds2 and da3)

↓

[System utterance (S3)]
"from two p.m. to three p.m.?"
[Dialogue act]
[act-type=confirm-time,start=14:00, end=15:00]

↓

[User utterance (U3)] "yes"
[Speech recognition hypotheses]
1. "yes"
[Dialogue act candidates]
1. [act-type=acknowledge]
[Intention recognition result candidates]
1. [room=nil, start=14:00, end=15:00]
2. [room=nil, start=nil, end=15:00]

Figure 3: Detailed description of the understanding of the example dialogue.

ation of discourse accuracy, comparing reference dialogue states with hypothesis dialogue states. Both these methods employ hand-crafted rules to score the dialogue states to decide the best dialogue state. Creating such rules requires expert knowledge, and is also time consuming.

There are approaches that propose statistically estimating the dialogue act type from several previous dialogue act types using N-gram probability (Nagata and Morimoto, 1994; Reithinger and Maier, 1995). Although their approaches can be used for disambiguating user utterance using discourse information, they do not consider holding multiple dialogue states.

In the context of plan-based utterance understanding (Allen and Perrault, 1980; Carberry, 1990), when there is ambiguity in the understanding result of a user utterance, an interpretation best suited to the estimated plan should be selected. In addition, the system must choose the most plausible plans from multiple possible candidates. Although we do not adopt plan-based representation of dialogue states as noted before, this problem is close to what we are dealing with. Unfortunately, however, it seems that no systematic ways to score the candidates for disambiguation have been proposed.

5 Approach

The discourse understanding method that we propose takes the same approach as Miyazaki et al. (2002). However, our method is different in that, when ordering the multiple dialogue states, the statistical information derived from the dialogue corpora is used. We propose using two kinds of statistical information:

1. the probability of a dialogue act type sequence, and

2. the collocation probability of a dialogue state and the next dialogue act.

5.1 Statistical Information

Probability of a dialogue act type sequence Based on the same idea as Nagata and Morimoto (1994) and Reithinger and Maier (1995), we use the probability of a dialogue act type sequence, namely, the N-gram probability of dialogue act types. System utterances and the transcription of user utterances are both converted to dialogue acts using a dialogue act conversion parser, then the N-gram probability of the dialogue act types is calculated.

#	explanation
1.	whether slots asked previously by the system are changed
2.	whether slots being confirmed are changed
3.	whether slots already confirmed are changed
4.	whether the dialogue act fills slots that do not have values
5.	whether the dialogue act tries changing slots that have values
6.	when 5 is true, whether slot values are not changed as a result
7.	whether the dialogue act updates the initial dialogue state [5]

Table 2: Seven binary attributes to classify collocation patterns of a dialogue state and the next dialogue act.

Collocation probability of a dialogue state and the next dialogue act From the dialogue corpora, dialogue states and the succeeding user utterances are extracted. Then, pairs comprising a dialogue state and a dialogue act are created after converting user utterances into dialogue acts. Contrary to the probability of sequential patterns of dialogue act types that represents a brief flow of a dialogue, this collocation information expresses a local detailed flow of a dialogue, such as dialogue state changes caused by the dialogue act. The simple bigram of dialogue states and dialogue acts is not sufficient due to the complexity of the data that a dialogue state possesses, which can cause data sparseness problems. Therefore, we classify the ways that dialogue states are changed by dialogue acts into 64 classes characterized by seven binary attributes (Table 2) and compute the occurrence probability of each class in the corpora. We assume that the understanding result of the user intention contained in a dialogue state is expressed as a frame, which is common in many systems (Bobrow et al., 1977). A frame is a bundle of slots that consist of attribute-value pairs concerning a certain domain.

[5]The first user utterance should be treated separately, because the system's initial utterance is an open question leading to an unrestricted utterance of a user.

5.2 Scoring of Dialogue Acts

Each speech recognition hypothesis is converted to a dialogue act or acts. When there are several dialogue acts corresponding to a speech recognition hypothesis, all possible dialogue acts are created as in Figure 3, where the utterance "uh two p.m." produces two dialogue act candidates. Each dialogue act is given a score using its linguistic and acoustic scores. The linguistic score represents the grammatical adequacy of a speech recognition hypothesis from which the dialogue act originates, and the acoustic score the acoustic reliability of a dialogue act. Sometimes, there is a case that a dialogue act has such a low acoustic or linguistic score and that it is better to ignore the act. We therefore create a dialogue act called *null act*, and add this *null act* to our list of dialogue acts. A *null act* is a dialogue act that does not change the dialogue state at all.

5.3 Scoring of Dialogue States

Since the dialogue state is uniquely updated by a dialogue act, if there are l dialogue acts derived from speech understanding and m dialogue states, $m \times l$ new dialogue states are created. In this case, we define the score of a dialogue state S_{t+1} as

$$S_{t+1} = S_t + \alpha \cdot s_{act} + \beta \cdot s_{ngram} + \gamma \cdot s_{col}$$

where S_t is the score of a dialogue state just before the update, s_{act} the score of a dialogue act, s_{ngram} the score concerning the probability of a dialogue act type sequence, s_{col} the score concerning the collocation probability of dialogue states and dialogue acts, and α, β, and γ are the weighting factors.

5.4 Ordering of Dialogue States

The newly created dialogue states are ordered based on the score. The dialogue state that has the best score is regarded as the most probable one, and the system responds to the user by referring to it. The maximum number of dialogue states is needed in order to drop low-score dialogue states and thereby perform the operation in real time. This dropping process can be considered as a beam search in view of the entire discourse process, thus we name the maximum number of dialogue states *the dialogue state beam width*.

6 Experiment

6.1 Extracting Statistical Information from Dialogue Corpus

Dialogue Corpus We analyzed a corpus of dialogues between naive users and a Japanese spoken dialogue system, which were collected in acoustically insulated booths. The task domain was meeting room reservation. Subjects were instructed to reserve a meeting room on a certain date from a certain time to a certain time. As a speech recognition engine, Julius3.1p1 (Lee et al., 2001) was used with its attached acoustic model. For the language model, we used a trigram trained from randomly generated texts of acceptable phrases. For system response, NTT's speech synthesis engine FinalFluet (Takano et al., 2001) was used. The system had a vocabulary of 168 words, each registered with a category and a semantic feature in its lexicon. The system used hand-crafted rules for discourse understanding. The corpus consists of 240 dialogues from 15 subjects (10 males and 5 females), each one performing 16 dialogues. Dialogues that took more than three minutes were regarded as failures. The task completion rate was 78.3% (188/240).

Extraction of Statistical Information From the transcription, we created a trigram of dialogue act types using the CMU-Cambridge Toolkit (Clarkson and Rosenfeld, 1997). Figure 3 shows an example of the trigram information starting from {*refer-start-time backchannel*}. The bigram information used for smoothing is also shown. The collocation probability was obtained from the recorded dialogue states and the transcription following them. Out of 64 possible patterns, we found 17 in the corpus as shown in Figure 4. Taking the case of the example dialogue in Figure 3, it happened that the sequence {*refer-start-time backchannel refer-end-time*} does not appear in the corpus; thus, the probability is calculated based on the bigram probability using the backoff weight, which is 0.006. The trigram probability for {*refer-end-time backchannel refer-end-time*} is 0.031.

The collocation probability of the sequence ds1 + da3 → ds3 fits collocation pattern 12, where a slot having no value was changed. The sequence ds2 + da3 → ds4 fits collocation pattern 17, where a slot having a value was changed to have a different value. The probabilities were 0.155 and 0.009, respectively. By the simple adding of the two probabilities in common logarithms in each case, ds3 has the probability score -3.015 and ds4 -3.549, suggesting that the sequence ds3 is the most probable discourse understanding result after U2.

dialogue act type sequence (trigram)	probability score
refer-start-time backchannel backchannel	-1.0852
refer-start-time backchannel ask-date	-2.0445
refer-start-time backchannel ask-start-time	-0.8633
refer-start-time backchannel request	-2.0445
refer-start-time backchannel refer-day	-1.7790
refer-start-time backchannel refer-month	-0.4009
refer-start-time backchannel refer-room	-0.8633
refer-start-time backchannel refer-start-time	-0.7172

dialogue act type sequence (bigram)	backoff weight	probability score
refer-start-time backchannel	-1.1337	-0.7928
refer-end-time backchannel	0.4570	-0.6450
backchannel refer-end-time	-0.5567	-1.0716

Table 3: An example of bigram and trigram of dialogue act types with their probability score in common logarithm.

#	collocation pattern	occurrence probability
1.	0 1 1 1 0 0 1	0.001
2.	0 1 1 0 0 1 0	0.053
3.	0 0 0 0 0 0 0	0.273
4.	1 0 0 0 1 0 0	0.001
5.	1 0 1 1 0 0 0	0.005
6.	0 0 1 1 0 0 0	0.036
7.	0 0 0 0 1 0 0	0.047
8.	0 1 1 0 1 0 0	0.041
9.	0 0 1 1 0 0 1	0.010
10.	0 0 1 0 0 1 0	0.016
11.	0 0 0 0 0 0 1	0.064
12.	0 0 0 1 0 0 0	0.155
13.	1 0 0 1 0 0 0	0.043
14.	0 0 1 0 1 0 0	0.061
15.	1 0 0 1 0 0 1	0.001
16.	0 0 0 1 0 0 1	0.186
17.	0 0 0 0 0 1 0	0.009

Table 4: The 17 collocation patterns and their occurrence probabilities. See Figure 2 for the detail of binary attributes. Attributes 1-7 are ordered from left to right.

6.2 Verification of our approach

To verify the effectiveness of the proposed approach, we built a Japanese spoken dialogue system in the meeting reservation domain that employs the proposed discourse understanding method and performed dialogue experiments.

The speech recognition engine was Julius3.3p1 (Lee et al., 2001) with its attached acoustic models. For the language model, we made a trigram from the transcription obtained from the corpora. The system had a vocabulary of 243. The recognition engine outputs 5-best recognition hypotheses. This time, values for s_{act}, s_{ngram}, s_{col} are the logarithm of the inverse number of n-best ranks,[6] the log likelihood of dialogue act type trigram probability, and the common logarithm of the collocation probability, respectively. For the experiment, weighting factors are all set to one ($\alpha = \beta = \gamma = 1$). The dialogue state beam width was 15. We collected 256 dialogues from 16 subjects (7 males and 9 females). The speech recognition accuracy (word error rate) was 65.18%. Dialogues that took more than five minutes were regarded as failures. The task completion rate was 88.3% (226/256).[7]

From all user speech intervals, the number of times that dialogue states below second place became first place was 120 (7.68%), showing a relative frequency of shuffling within the dialogue states.

6.3 Effectiveness of Holding Multiple Dialogue States

The main reason that we developed the proposed corpus-based discourse understanding method was that it is difficult to manually create rules to deal with multiple dialogue states. It is yet to be examined, however, whether holding multiple dialogue states is really effective for accurate discourse understanding.

To verify that holding multiple dialogue states is effective, we fixed the speech recognizer's output to 1-best, and studied the system performance changes when the dialogue state beam width was changed from 1 to 30. When the dialogue state beam width is too large, the computational cost becomes high and the system cannot respond in real time. We therefore selected 30 for empirical reasons.

The task domain and other settings were the same as in the previous experiment except for the dialogue state beam width changes. We collected 448 dialogues from 28 subjects (4 males and 24 females), each one performing 16 dialogues. Each subject was instructed to reserve the same meeting room twice, once with the 1-beam-width system and again with 30-beam-width system. The order of what room to reserve and what system to use was randomized. The speech recognition accuracy was 69.17%. Dialogues that took more than five minutes were regarded as failures. The task completion rates for the 1-beam-width system and the 30-beam-width system were 88.3% and 91.0%, and the average task completion times were 107.66 seconds and 95.86 seconds, respectively. A statistical hypothesis test showed that times taken to carry out a task with the 30-beam-width system are significantly shorter than those with the 1-beam-width system ($Z = -2.01$, $p < .05$). In this test, we used a kind of censored mean computed by taking the mean of the times only for subjects that completed the tasks with both systems. The population distribution was estimated by the bootstrap method (Cohen, 1995). It may be possible to evaluate the discourse understanding by comparing the best dialogue state with the reference dialogue state, and calculate a metric such as the CER (concept error rate) as Miyazaki et al. (2002) do; however it is not clear whether the discourse understanding can be evaluated this way, since it is not certain whether the CER correlates closely with the system's performance (Higashinaka et al., 2002). Therefore, this time, we used the task completion time and the task completion rate for comparison.

7 Discussion

Cost of creating the discourse understanding component The best task completion rate in the experiments was 91.0% (the case of 1-best recognition input and a 30 dialogue state beam width). This high rate suggests that the proposed approach is effective in reducing the cost of creating the discourse understanding component in that no hand-crafted rules are necessary. For statistical discourse understanding, an initial system, e.g., a system that employs the proposed approach with only s_{act} for scoring the dialogue states, is needed in order to create the dialogue corpus; however, once it has been made, the creation of the discourse understanding component

[6] In this experiment, only the acoustic score of a dialogue act was considered.

[7] It should be noted that due to the creation of an enormous number of dialogue states in discourse understanding, the proposed system takes a few seconds to respond after the user input.

requires no expert knowledge.

Effectiveness of holding multiple dialogue states The result of the examination of dialogue state beam width changes suggests that holding multiple dialogue states shortens the task completion time. As far as task-oriented spoken dialogue systems are concerned, holding multiple dialogue states contributes to the accuracy of discourse understanding.

8 Summary and Future Work

We proposed a new discourse understanding method that orders multiple dialogue states created from multiple dialogue states and the succeeding speech understanding results based on statistical information obtained from dialogue corpora. The results of the experiments show that our approach is effective in reducing the cost of creating the discourse understanding component, and the advantage of keeping multiple dialogue states was also shown.

There still remain several issues that we need to explore. These include the use of statistical information other than the probability of a dialogue act type sequence and the collocation probability of dialogue states and dialogue acts, the optimization of weighting factors α, β, γ, other default parameters that we used in the experiments, and more experiments in larger domains. Despite these issues, the present results have shown that our approach is promising.

Acknowledgements

We thank Dr. Hiroshi Murase and all members of the Dialogue Understanding Research Group for useful discussions. Thanks also go to the anonymous reviewers for their helpful comments.

References

James F. Allen and C. Raymond Perrault. 1980. Analyzing intention in utterances. *Artif. Intel.*, 15:143–178.

James Allen, George Ferguson, and Amanda Stent. 2001. An architecture for more realistic conversational systems. In *Proc. IUI*, pages 1–8.

Daniel G. Bobrow, Ronald M. Kaplan, Martin Kay, Donald A. Norman, Henry Thompson, and Terry Winograd. 1977. GUS, a frame driven dialog system. *Artif. Intel.*, 8:155–173.

Sandra Carberry. 1990. *Plan Recognition in Natural Language Dialogue*. MIT Press, Cambridge, Mass.

Junnifer Chu-Carroll. 2000. MIMIC: An adaptive mixed initiative spoken dialogue system for information queries. In *Proc. 6th Applied NLP*, pages 97–104.

P.R. Clarkson and R. Rosenfeld. 1997. Statistical language modeling using the CMU-Cambridge toolkit. In *Proc. Eurospeech*, pages 2707–2710.

Paul R. Cohen. 1995. *Empirical Methods for Artificial Intelligence*. MIT Press.

Ryuichiro Higashinaka, Noboru Miyazaki, Mikio Nakano, and Kiyoaki Aikawa. 2002. A method for evaluating incremental utterance understanding in spoken dialogue systems. In *Proc. ICSLP*, pages 829–832.

Akinobu Lee, Tatsuya Kawahara, and Kiyohiro Shikano. 2001. Julius – an open source real-time large vocabulary recognition engine. In *Proc. Eurospeech*, pages 1691–1694.

Noboru Miyazaki, Mikio Nakano, and Kiyoaki Aikawa. 2002. Robust speech understanding using incremental understanding with n-best recognition hypotheses. In *SIG-SLP-40, Information Processing Society of Japan.*, pages 121–126. (in Japanese).

Masaaki Nagata and Tsuyoshi Morimoto. 1994. First steps toward statistical modeling of dialogue to predict the speech act type of the next utterance. *Speech Communication*, 15:193–203.

Mikio Nakano, Noboru Miyazaki, Jun-ichi Hirasawa, Kohji Dohsaka, and Takeshi Kawabata. 1999. Understanding unsegmented user utterances in real-time spoken dialogue systems. In *Proc. 37th ACL*, pages 200–207.

Norbert Reithinger and Elisabeth Maier. 1995. Utilizing statistical dialogue act processing in Verbmobil. In *Proc. 33th ACL*, pages 116–121.

Charles Rich, Candace Sidner, and Neal Lesh. 2001. COLLAGEN: Applying collaborative discourse theory. *AI Magazine*, 22(4):15–25.

Stephanie Seneff. 2002. Response planning and generation in the MERCURY flight reservation system. *Computer Speech and Language*, 16(3–4):283–312.

Satoshi Takano, Kimihito Tanaka, Hideyuki Mizuno, Masanobu Abe, and ShiN'ya Nakajima. 2001. A Japanese TTS system based on multi-form units and a speech modification algorithm with harmonics reconstruction. *IEEE Transactions on Speech and Processing*, 9(1):3–10.

Victor W. Zue and James R. Glass. 2000. Conversational interfaces: Advances and challenges. *Proceedings of IEEE*, 88(8):1166–1180.