Understanding Unsegmented User Utterances in Real-Time Spoken Dialogue Systems

Mikio Nakano, Noboru Miyazaki, Jun-ichi Hirasawa,
Kohji Dohsaka, Takeshi Kawabata*
NTT Laboratories
3-1 Morinosato-Wakamiya, Atsugi 243-0198, Japan
nakano@atom.brl.ntt.co.jp, nmiya@atom.brl.ntt.co.jp, jun@idea.brl.ntt.co.jp,
dohsaka@atom.brl.ntt.co.jp, kaw@nttspch.hil.ntt.co.jp

Abstract

This paper proposes a method for incrementally understanding user utterances whose semantic boundaries are not known and responding in real time even before boundaries are determined. It is an integrated parsing and discourse processing method that updates the partial result of understanding word by word, enabling responses based on the partial result. This method incrementally finds plausible sequences of utterances that play crucial roles in the task execution of dialogues, and utilizes beam search to deal with the ambiguity of boundaries as well as syntactic and semantic ambiguities. The results of a preliminary experiment demonstrate that this method understands user utterances better than an understanding method that assumes pauses to be semantic boundaries.

1 Introduction

Building a real-time, interactive spoken dialogue system has long been a dream of researchers, and the recent progress in hardware technology and speech and language processing technologies is making this dream a reality. It is still hard, however, for computers to understand unrestricted human utterances and respond appropriately to them. Considering the current level of speech recognition technology, system-initiative dialogue systems, which prohibit users from speaking unrestrictedly, are preferred (Walker et al., 1998). Nevertheless, we are still pursuing techniques for understanding unrestricted user utterances because, if the accuracy of understanding can be improved, systems that allow users to speak freely could be developed and these would be more useful than systems that do not.

Most previous spoken dialogue systems (e.g. systems by Allen et al. (1996), Zue et al. (1994) and Peckham (1993)) assume that the user makes one utterance unit in each speech interval, unless the push-to-talk method is used. Here, by *utterance unit* we mean a phrase from which a speech act representation is derived, and it corresponds to a *sentence* in written language. We also use *speech act* in this paper to mean a command that updates the hearer's belief state about the speaker's intention and the context of the dialogue. In this paper, a system using this assumption is called an *interval-based system.*

The above assumption no longer holds when no restrictions are placed on the way the user speaks. This is because utterance boundaries (i.e., semantic boundaries) do not always correspond to pauses and techniques based on other acoustic information are not perfect. Utterance boundaries thus cannot be identified prior to parsing, and so the timing of determining parsing results to update the belief state is unclear. On the other hand, responding to a user utterance in real time requires understanding it and updating the belief state in real time; thus, it is impossible to wait for subsequent inputs to determine boundaries.

Abandoning full parsing and adopting keyword-based or fragment-based understanding could prevent this problem. This would, however, sacrifice the accuracy of understanding because phrases across the pauses could not be syntactically analyzed. There is, therefore, a need for a method based on full parsing that enables real-time understanding of user utterances without boundary information.

This paper presents *incremental significant-utterance-sequence search* (ISSS), a method that

* Current address: NTT Laboratories, 1-1 Hikarino-oka, Yokosuka 239-0847, Japan

enables incremental understanding of user utterances word by word by finding plausible sequences of utterances that play crucial roles in the task execution of dialogues. The method utilizes beam search to deal with the ambiguity of boundaries as well as syntactic and semantic ambiguities. Since it outputs the partial result of understanding that is the most plausible whenever a word hypothesis is inputted, the response generation module can produce responses at any appropriate time. A comparison of an experimental spoken dialogue system using ISSS with an interval-based system shows that the method is effective.

2 Problem

A dilemma is addressed in this paper. First, it is difficult to identify utterance boundaries in spontaneous speech in real time using only pauses. Observation of human-human dialogues reveals that humans often put pauses in utterances and sometimes do not put pauses at utterance boundaries. The following human utterance shows where pauses might appear in an utterance.

> I'd like to make a reservation for a conference room ⟨*pause*⟩ for, uh ⟨*pause*⟩ this afternoon ⟨*pause*⟩ at about ⟨*pause*⟩ say ⟨*pause*⟩ 2 or 3 o'clock ⟨*pause*⟩ for ⟨*pause*⟩ 15 people

As far as Japanese is concerned, several studies have pointed out that speech intervals in dialogues are not always well-formed substrings (Seligman et al., 1997; Takezawa and Morimoto, 1997).

On the other hand, since parsing results cannot be obtained unless the end of the utterance is identified, making real-time responses is impossible without boundary information. For example, consider the utterance "I'd like to book Meeting Room 1 on Wednesday". It is expected that the system should infer the user wants to reserve the room on 'Wednesday this week' if this utterance was made on Monday. In real conversations, however, there is no guarantee that 'Wednesday' is the final word of the utterance. It might be followed by the phrase 'next week', in which case the system made a mistake in inferring the user's intention and must backtrack and re-understand. Thus, it is not possible to determine the interpretation unless the utterance boundary is identified. This problem is more serious in head-final languages such as Japanese because function words that represent negation come after content words. Since there is no explicit clue indicating an utterance boundary in unrestricted user utterances, the system cannot make an interpretation and thus cannot respond appropriately. Waiting for a long pause enables an interpretation, but prevents response in real time. We therefore need a way to reconcile real-time understanding and analysis without boundary clues.

3 Previous Work

Several techniques have been proposed to segment user utterances prior to parsing. They use intonation (Wang and Hirschberg, 1992; Traum and Heeman, 1997; Heeman and Allen, 1997) and probabilistic language models (Stolcke et al., 1998; Ramaswamy and Kleindienst, 1998; Cettolo and Falavigna, 1998). Since these methods are not perfect, the resulting segments do not always correspond to utterances and might not be parsable because of speech recognition errors. In addition, since the algorithms of the probabilistic methods are not designed to work in an incremental way, they cannot be used in real-time analysis in a straightforward way.

Some methods use keyword detection (Rose, 1995; Hatazaki et al., 1994; Seto et al., 1994) and key-phrase detection (Aust et al., 1995; Kawahara et al., 1996) to understand speech mainly because the speech recognition score is not high enough. The lack of the full use of syntax in these approaches, however, means user utterances might be misunderstood even if the speech recognition gave the correct answer. Zechner and Waibel (1998) and Worm (1998) proposed understanding utterances by combining partial parses. Their methods, however, cannot syntactically analyze phrases across pauses since they use speech intervals as input units. Although Lavie et al. (1997) proposed a segmentation method that combines segmentation prior to parsing and segmentation during parsing, but it suffers from the same problem.

In the parser proposed by Core and Schubert (1997), utterances interrupted by the other dialogue participant are analyzed based on meta-rules. It is unclear, however, how this parser can be incorpo-

rated into a real-time dialogue system; it seems that it cannot output analysis results without boundary clues.

4 Incremental Significant-Utterance-Sequence Search Method

4.1 Overview

The above problem can be solved by incremental understanding, which means obtaining the most plausible interpretation of user utterances every time a word hypothesis is inputted from the speech recognizer. For incremental understanding, we propose *incremental significant-utterance-sequence search* (ISSS), which is an integrated parsing and discourse processing method. ISSS holds multiple possible belief states and updates those belief states when a word hypothesis is inputted. The response generation module produces responses based on the most likely belief state. The timing of responses is determined according to the content of the belief states and acoustic clues such as pauses.

In this paper, to simplify the discussion, we assume the speech recognizer incrementally outputs elements of the recognized word sequence. Needless to say, this is impossible because the most likely word sequence cannot be found in the midst of the recognition; only networks of word hypotheses can be outputted. Our method for incremental processing, however, can be easily generalized to deal with incremental network input, and our experimental system utilizes the generalized method.

4.2 Significant-Utterance Sequence

A *significant utterance* (SU) in the user's speech is a phrase that plays a crucial role in performing the task in the dialogue. An SU may be a full sentence or a subsentential phrase such as a noun phrase or a verb phrase. Each SU has a speech act that can be considered a command to update the belief state. SU is defined as a syntactic category by the grammar for linguistic processing, which includes semantic inference rules.

Any phrases that can change the belief state should be defined as SUs. Two kinds of SUs can be considered; domain-related ones that express the user's intention about the task of the dialogue and dialogue-related ones that express the user's attitude with respect to the progress of the dialogue such as confirmation and denial. Considering a meeting room reservation system, examples of domain-related SUs are "I need to book Room 2 on Wednesday", "I need to book Room 2", and "Room 2" and dialogue-related ones are "yes", "no", and "Okay".

User utterances are understood by finding a sequence of SUs and updating the belief state based on the sequence. The utterances in the sequence do not overlap. In addition, they do not have to be adjacent to each other, which leads to robustness against speech recognition errors as in fragment-based understanding (Zechner and Waibel, 1998; Worm, 1998).

The belief state can be computed at any point in time if a significant-utterance sequence for user utterances up to that point in time is given. The belief state holds not only the user's intention but also the history of system utterances, so that all discourse information is stored in it.

Consider, for example, the following user speech in a meeting room reservation dialogue.

> I need to, uh, book Room 2, and it's on Wednesday.

The most likely significant-utterance sequence consists of "I need to, uh, book Room 2" and "it's on Wednesday". From the speech act representation of these utterances, the system can infer the user wants to book Room 2 on Wednesday.

4.3 Finding Significant-Utterance Sequences

SUs are identified in the process of understanding. Unlike ordinary parsers, the understanding module does not try to determine whether the whole input forms an SU or not, but instead determines where SUs are. Although this can be considered a kind of partial parsing technique (McDonald, 1992; Lavie, 1996; Abney, 1996), the SUs obtained by ISSS are not always subsentential phrases; they are sometimes full sentences.

For one discourse, multiple significant-utterance sequences can be considered. "Wednesday next week" above illustrates this well. Let us assume that the parser finds two SUs, "Wednesday" and "Wednesday next week". Then three significant-utterance sequences are possible: one consisting of "Wednesday", one consisting of "Wednesday next

week", and one consisting of no SUs. The second sequence is obviously the most likely at this point, but it is not possible to choose only one sequence and discard the others in the midst of a dialogue. We therefore adopt beam search. Priorities are assigned to the possible sequences, and those with low priorities are neglected during the search.

4.4 ISSS Algorithm

The ISSS algorithm is based on shift-reduce parsing. The basic data structure is *context*, which represents search information and is a triplet of the following data.

stack: A push-down stack used in a shift-reduce parser.

belief state: A set of the system's beliefs about the user's intention with respect to the task of the dialogue and dialogue history.

priority: A number assigned to the context.

Accordingly, the algorithm is as follows.

(I) Create a context in which the stack and the belief state are empty and the priority is zero.

(II) For each input word, perform the following process.

1. Obtain the lexical feature structure for the word and push it to the stacks of all existing contexts.
2. For each context, apply rules as in a shift-reduce parser. When a shift-reduce conflict or a reduce-reduce conflict occur, the context is duplicated and different operations are performed on them. When a reduce operation is performed, increase the priority of the context by the priority assigned to the rule used for the reduce operation.
3. For each context, if the top of the stack is an SU, empty the stack and update the belief state according to the content of the SU. Increase the priority by the square of the length (i.e., the number of words) of this SU.

(I) SU [day: ?x] → NP [sort: day, sem: ?x] (priority: 1)
(II) NP[sort: day] → NP [sort: day] NP [sort: week] (priority: 2)

Figure 1: Rules used in the example.

4. Discard contexts with low priority so that the number of remaining contexts will be the *beam width* or less.

Since this algorithm is based on beam search, it works in real time if Step (II) is completed quickly enough, which is the case in our experimental system.

The priorities for contexts are determined using a general heuristics based on the length of SUs and the kind of rules used. Contexts with longer SUs are preferred. The reason we do not use the length of an SU, but its square instead, is that the system should avoid regarding an SU as consisting of several short SUs. Although this heuristics seems rather simple, we have found it works well in our experimental systems.

Although some additional techniques, such as discarding redundant contexts and multiplying a weight w ($w > 1$) to the priority of each context after the Step 4, are effective, details are not discussed here for lack of space.

4.5 Response Generation

The contexts created by the utterance understanding module can also be accessed by the response generation module so that it can produce responses based on the belief state in the context with the highest priority at a point in time. We do not discuss the timing of the responses here, but, generally speaking, a reasonable strategy is to respond when the user pauses. In Japanese dialogue systems, producing a backchannel is effective when the user's intention is not clear at that point in time, but determining the content of responses in a real-time spoken dialogue system is also beyond the scope of this paper.

4.6 A Simple Example

Here we explain ISSS using a simple example. Consider again "Wednesday next week". To simplify the explanation, we assume the noun phrase

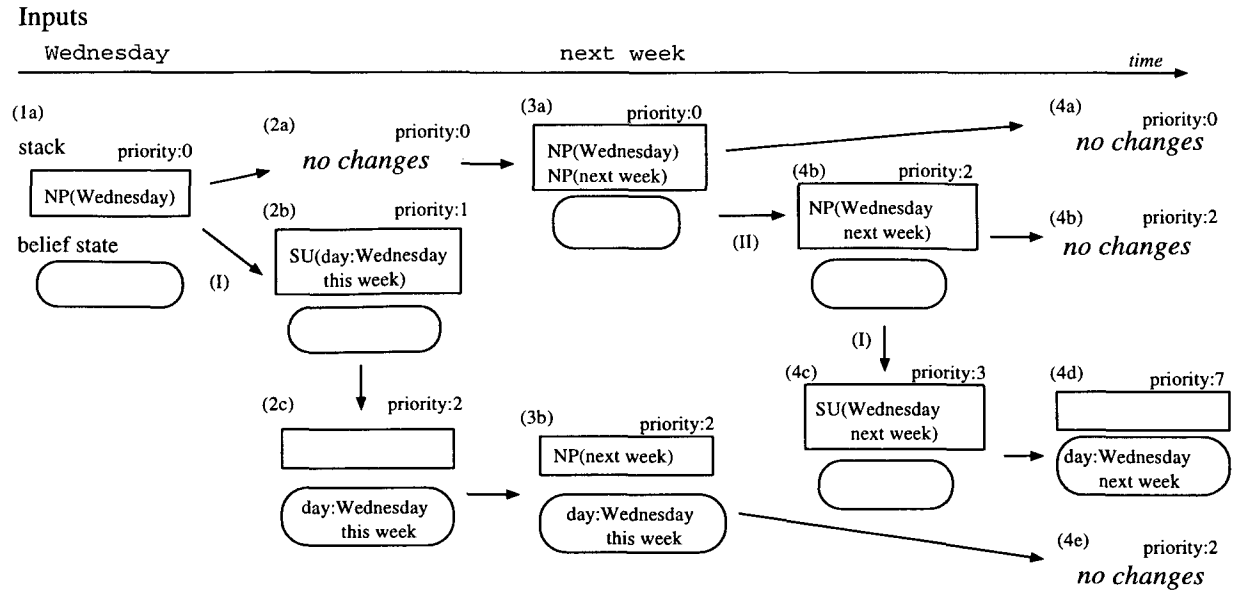

Figure 2: Execution of ISSS.

'next week' is one word. The speech recognizer incrementally sends to the understanding module the word hypotheses 'Wednesday' and 'next week'. The rules used in this example are shown in Figure 1. They are unification-based rules. Not all features and semantic constraints are shown. In this example, nouns and noun phrases are not distinguished. The ISSS execution is shown in Figure 2.

When 'Wednesday' is inputted, its lexical feature structure is created and pushed to the stack. Since Rule (I) can be applied to this stack, (2b) in Figure 2 is created. The top of the stack in (2b) is an SU, thus (2c) is created, whose belief state contains the user's intention of meeting room reservation on Wednesday this week. We assume that 'Wednesday' means Wednesday this week by default if this utterance was made on Monday, and this is described in the additional conditions in Rule (I). After 'next week' is inputted, NP is pushed to the stacks of all contexts, resulting in (3a) and (3b). Then Rule (II) is applied to (3a), making (4b). Rule (I) can be applied to (4b), and then (4c) is created and is turned into (4d), which has the highest priority.

Before 'next week' is inputted, the interpretation that the user wants to book a room on Wednesday this week has the highest priority, and then after that, the interpretation that the user wants to book a room on Wednesday next week has the highest priority. Thus, by this method, the most plausible interpretation can be obtained in an incremental way.

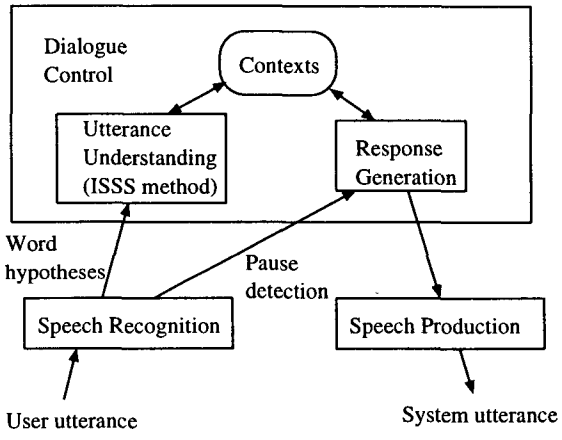

Figure 3: Architecture of the experimental systems.

5 Implementation

Using ISSS, we have developed several experimental Japanese spoken dialogue systems, including a meeting room reservation system.

The architecture of the systems is shown in Figure 3. The speech recognizer uses HMM-based continuous speech recognition directed by a regular

grammar (Noda et al., 1998). This grammar is weak enough to capture spontaneously spoken utterances, which sometimes include fillers and self-repairs, and allows each speech interval to be an arbitrary number of arbitrary *bunsetsu* phrases.[1] The grammar contains less than one hundred words for each task; we reduced the vocabulary size so that the speech recognizer could output results in real time. The speech recognizer incrementally outputs word hypotheses as soon as they are found in the best-scored path in the forward search (Hirasawa et al., 1998; Görz et al., 1996). Since each word hypothesis is accompanied by the pointer to its preceding word, the understanding module can reconstruct word sequences. The newest word hypothesis determines the word sequence that is acoustically most likely at a point in time.[2]

The utterance understanding module works based on ISSS and uses a domain-dependent unification grammar with a context-free backbone that is based on bunsetsu phrases. This grammar is more restrictive than the grammar for speech recognition, but covers phenomena peculiar to spoken language such as particle omission and self-repairs. A belief state is represented by a frame (Bobrow et al., 1977); thus, a speech act representation is a command for changing the slot value of a frame. Although a more sophisticated model would be required for the system to engage in a complicated dialogue, frame representations are sufficient for our tasks. The response generation module is invoked when the user pauses, and plans responses based on the belief state of the context with the highest priority. The response strategy is similar to that of previous frame-based dialogue systems (Bobrow et al., 1977). The speech production module outputs speech according to orders from the response generation module.

Figure 4 shows the transcription of an example dialogue of a reservation system that was recorded in the experiment explained below. As an example of SUs across pauses, "*gozen-jûji kara gozen-jûichiji made* (from 10 a.m. to 11 a.m.)" in U5 and U7

S1:	donoyôna goyôken de shôka (May I help you?)	5.69–7.19
U2:	kaigishitsu no yoyaku o onegaishimasu (I'd like to book a meeting room.) [*hai sôdesu gogoyoji made* (*That's right, to 4 p.m.*)]	7.79–9.66
S3:	hai (uh-huh)	10.06–10.32
U4:	e konshû no suiyôbi (Well, Wednesday this week) [*iie konshû no suiyôbi* (*No, Wednesday this week*)]	11.75–13.40
S5:	hai (uh-huh)	14.04–14.31
U5:	gozen-jûji kara (from 10 a.m.) [*gozen-jûji kara* (*from 10 a.m.*)]	15.13–16.30
S6:	hai (uh-huh)	17.15–17.42
U7:	gozen-jûichiji made (to 11 a.m.) [*gozen-jûichiji made* (*to 11 a.m.*)]	18.00–19.46
S8:	hai (uh-huh)	19.83–20.09
U9:	daisan- (three) [*daisan-kaigishitu* (*Meeting Room 3*)]	20.54–21.09
S10:	hai (uh-huh)	21.92–22.19
U11:	daisan-kaigishitu o onegaishimasu (I'd like to book Meeting Room 3) [failure]	21.52–23.59
S12:	hai (uh-huh)	24.05–24.32
U13:	yoyaku o onegaishimasu (Please book it) [*jûniji* (*12 o'clock*)]	25.26–26.52
S14:	hai (uh-huh)	27.09–27.36
U15:	yoyaku shitekudasai (Please book it) [*yoyaku shitekudasai* (*Please book it*)]	31.72–32.65
S16:	konshû no suiyôbi gozen-jûji kara gozen-jûichiji made daisan-kaigishitu toyûkotode yoroshî-deshôka (Wednesday this week, from 10 a.m. to 11 a.m., meeting room 3, OK?)	33.62–39.04
U17:	hai (yes) [*hai* (*yes*)]	40.85–41.10
S18:	kashikomarimashitâ (All right)	41.95–43.00

Figure 4: Example dialogue.
S means a system utterance and U a user utterance. Recognition results are enclosed in square brackets. The figures in the rightmost column are the start and end times (in seconds) of utterances.

was recognized. Although the SU "*jûniji yoyaku shitekudasai* (12 o'clock, please book it)" in U13 and U15 was syntactically recognized, the system could not interpret it well enough to change the frame because of grammar limitations. The reason why the user hesitated to utter U15 is that S14 was not what the user had expected.

We conducted a preliminary experiment to investigate how ISSS improves the performance of spoken dialogue systems. Two systems were com-

[1] A *bunsetsu* phrase is a phrase that consists of one content word and a number (possibly zero) of function words.

[2] A method for utilizing word sequences other than the most likely one and integrating acoustic scores and ISSS priorities remains as future work.

pared: one that uses ISSS (system A), and one that requires each speech interval to be an SU (an interval-based system, system B). In system B, when a speech interval was not an SU, the frame was not changed. The dialogue task was a meeting room reservation. Both systems used the same speech recognizer and the same grammar. There were ten subjects and each carried out a task on the two systems, resulting in twenty dialogues. The subjects were using the systems for the first time. They carried out one practice task with system B beforehand. This experiment was conducted in a computer terminal room where the machine noise was somewhat adverse to speech recognition. A meaningful discussion on the success rate of utterance segmentation is not possible because of the recognition errors due to the small coverage of the recognition grammar.[3]

All subjects successfully completed the task with system A in an average of 42.5 seconds, and six subjects did so with system B in an average of 55.0 seconds. Four subjects could not complete the task in 90 seconds with system B. Five subjects completed the task with system A 1.4 to 2.2 times quicker than with system B and one subject completed it with system B one second quicker than with system A. A statistical hypothesis test showed that times taken to carry out the task with system A are significantly shorter than those with system B ($Z = 3.77$, $p < .0001$).[4] The order in which the subjects used the systems had no significant effect. In addition, user impressions of system A were generally better than those of system B. Although there were some utterances that the system misunderstood because of grammar limitations, excluding the data for the three subjects who had made those utterances did not change the statistical results.

The reason it took longer to carry out the tasks with system B is that, compared to system A, the probability that it understood user utterances was much lower. This is because the recognition results of speech intervals do not always form one SU. About 67% of all recognition results of user speech intervals were SUs or fillers.[5]

Needless to say, these results depend on the recognition grammar, the grammar for understanding, the response strategy and other factors. It has been suggested, however, that assuming each speech interval to be an utterance unit could reduce system performance and that ISSS is effective.

6 Concluding Remarks

This paper proposed ISSS (incremental significant-utterance-sequence search), an integrated incremental parsing and discourse processing method that enables both the understanding of unsegmented user utterances and real-time responses. This paper also reported an experimental result which suggested that ISSS is effective. It is also worthwhile mentioning that using ISSS enables building spoken dialogue systems with less effort because it is possible to define significant utterances without considering where pauses might appear.

Acknowledgments

We would like to thank Dr. Ken'ichiro Ishii, Dr. Norihiro Hagita, and Dr. Kiyoaki Aikawa, and the members of the Dialogue Understanding Research Group for their helpful comments. We used the speech recognition engine REX developed by NTT Cyber Space Laboratories and would like to thank those who helped us use it. Thanks also go to the subjects of the experiment. Comments by the anonymous reviewers were of great help.

References

Steven Abney. 1996. Partial parsing via finite-state cascades. In *Proceedings of the ESSLLI '96 Robust Parsing Workshop*, pages 8–15.

James F. Allen, Bradford W. Miller, Eric K. Ringger, and Teresa Sikorski. 1996. A robust system for natural spoken dialogue. In *Proceedings of ACL-96*, pages 62–70.

Harald Aust, Martin Oerder, Frank Seide, and Volker Steinbiss. 1995. The Philips automatic train timetable information system. *Speech Communication*, 17:249–262.

[3] About 50% of user speech intervals were not covered by the recognition grammar due to the small vocabulary size of the recognition grammar. For the remaining 50% of the intervals, the word error rate of recognition was about 20%. The word error rate is defined as 100 * (*substitutions* + *deletions* + *insertions*) / (*correct* + *substitutions* + *deletions*) (Zechner and Waibel, 1998).

[4] In this test, we used a kind of censored mean which is computed by taking the mean of the logarithms of the ratios of the times only for the subjects that completed the tasks with both systems. The population distribution was estimated by the bootstrap method (Cohen, 1995).

[5] Note that 91% of user speech intervals were well-formed substrings (not necessary SUs).

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

Mauro Cettolo and Daniele Falavigna. 1998. Automatic detection of semantic boundaries based on acoustic and lexical knowledge. In *Proceedings of ICSLP-98*, pages 1551–1554.

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

Mark G. Core and Lenhart K. Schubert. 1997. Handling speech repairs and other disruptions through parser metarules. In *Working Notes of AAAI Spring Symposium on Computational Models for Mixed Initiative Interaction*, pages 23–29.

Günther Görz, Marcus Kesseler, Jörg Spilker, and Hans Weber. 1996. Research on architectures for integrated speech/language systems in Verbmobil. In *Proceedings of COLING-96*, pages 484–489.

Kaichiro Hatazaki, Farzad Ehsani, Jun Noguchi, and Takao Watanabe. 1994. Speech dialogue system based on simultaneous understanding. *Speech Communication*, 15:323–330.

Peter A. Heeman and James F. Allen. 1997. Intonational boundaries, speech repairs, and discourse markers: Modeling spoken dialog. In *Proceedings of ACL/EACL-97*.

Jun-ichi Hirasawa, Noboru Miyazaki, Mikio Nakano, and Takeshi Kawabata. 1998. Implementation of coordinative nodding behavior on spoken dialogue systems. In *Proceedings of ICSLP-98*, pages 2347–2350.

Tatsuya Kawahara, Chin-Hui Lee, and Biing-Hwang Juang. 1996. Key-phrase detection and verification for flexible speech understanding. In *Proceedings of ICSLP-96*, pages 861–864.

Alon Lavie, Donna Gates, Noah Coccaro, and Lori Levin. 1997. Input segmentation of spontaneous speech in JANUS: A speech-to-speech translation system. In Elisabeth Maier, Marion Mast, and Susann LuperFoy, editors, *Dialogue Processing in Spoken Language Systems*, pages 86–99. Springer-Verlag.

Alon Lavie. 1996. *GLR*: A Robust Grammar-Focused Parser for Spontaneously Spoken Language*. Ph.D. thesis, School of Computer Science, Carnegie Mellon University.

David D. McDonald. 1992. An efficient chart-based algorithm for partial-parsing of unrestricted texts. In *Proceedings of the Third Conference on Applied Natural Language Processing*, pages 193–200.

Yoshiaki Noda, Yoshikazu Yamaguchi, Tomokazu Yamada, Akihiro Imamura, Satoshi Takahashi, Tomoko Matsui, and Kiyoaki Aikawa. 1998. The development of speech recognition engine REX. In *Proceedings of the 1998 IEICE General Conference D-14-9*, page 220. (in Japanese).

Jeremy Peckham. 1993. A new generation of spoken language systems: Results and lessons from the SUNDIAL project. In *Proceedings of Eurospeech-93*, pages 33–40.

Ganesh N. Ramaswamy and Jan Kleindienst. 1998. Automatic identification of command boundaries in a conversational natural language user interface. In *Proceedings of ICSLP-98*, pages 401–404.

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

Marc Seligman, Junko Hosaka, and Harald Singer. 1997. "Pause units" and analysis of spontaneous Japanese dialogues: Preliminary studies. In Elisabeth Maier, Marion Mast, and Susann LuperFoy, editors, *Dialogue Processing in Spoken Language Systems*, pages 100–112. Springer-Verlag.

Shigenobu Seto, Hiroshi Kanazawa, Hideaki Shinchi, and Yoichi Takebayashi. 1994. Spontaneous speech dialogue system TOSBURG-II and its evaluation. *Speech Communication*, 15:341–353.

Andreas Stolcke, Elizabeth Shriberg, Rebecca Bates, Mari Ostendorf, Dilek Hakkani, Madelaine Plauche, Gökhan Tür, and Yu Lu. 1998. Automatic detection of sentence boundaries and disfluencies based on recognized words. In *Proceedings of ICSLP-98*, pages 2247–2250.

Toshiyuki Takezawa and Tsuyoshi Morimoto. 1997. Dialogue speech recognition method using syntactic rules based on subtrees and preterminal bigrams. *Systems and Computers in Japan*, 28(5):22–32.

David R. Traum and Peter A. Heeman. 1997. Utterance units in spoken dialogue. In Elisabeth Maier, Marion Mast, and Susann LuperFoy, editors, *Dialogue Processing in Spoken Language Systems*, pages 125–140. Springer-Verlag.

Marilyn A. Walker, Jeanne C. Fromer, and Shrikanth Narayanan. 1998. Learning optimal dialogue strategies: A case study of a spoken dialogue agent for email. In *Proceedings of COLING-ACL'98*.

Michelle Q. Wang and Julia Hirschberg. 1992. Automatic classification of intonational phrase boundaries. *Computer Speech and Language*, 6:175–196.

Karsten L. Worm. 1998. A model for robust processing of spontaneous speech by integrating viable fragments. In *Proceedings of COLING-ACL'98*, pages 1403–1407.

Klaus Zechner and Alex Waibel. 1998. Using chunk based partial parsing of spontaneous speech in unrestricted domains for reducing word error rate in speech recognition. In *Proceedings of COLING-ACL'98*, pages 1453–1459.

Victor Zue, Stephanie Seneff, Joseph Polifroni, Michael Phillips, Christine Pao, David Goodine, David Goddeau, and James Glass. 1994. PEGASUS: A spoken dialogue interface for on-line air travel planning. *Speech Communication*, 15:331–340.