Example-based Spoken Dialogue System using WOZ System Log

Hiroya MURAO *,**, **Nobuo KAWAGUCHI** **,† **Shigeki MATSUBARA** **,†
Yukiko YAMAGUCHI† **Yasuyoshi INAGAKI**‡
* Digital Systems Development Center,SANYO Electric Co., Ltd.,
Hirakata-shi, Osaka, 573-8534 Japan
** Center for Integrated Acoustic Information Research,Nagoya University,
† Information Technology Center, Nagoya University,
Furo-cho, Chikusa-ku, Nagoya-shi, 464-8603 Japan
‡ The Faculty of Information Science and Technology, Aichi Prefectural University,
Nagakute-cho, Aichi-gun, Aichi, 480-1198, Japan
murao@hr.hm.rd.sanyo.co.jp

Abstract

This paper proposes a new framework for a spoken dialogue system based on dialogue examples between human subjects and the Wizard of OZ (WOZ) system. Using this framework and a model of information retrieval dialogue, a spoken dialogue system for retrieving shop information while driving in a car has been designed. The system refers to the dialogue examples to find an example that is suitable for generating a query or a reply. The authors have also constructed a large-scale dialogue database using a WOZ system, which enables efficient collection of dialogue examples.

1 Introduction

Against the background of ever-increasing computing power, techniques for constructing spoken dialogue systems using large-scale speech and text corpora have become the target of much research (Levin et al., 1998; Young, 2002). In prior research, the authors have proposed a spoken-dialogue control technique using dialogue examples with the aim of performing flexible dialogue control during information-retrieval dialogue and of achieving speech understanding robust against speech recognition errors (Murao et al., 2001). This technique uses input speech data and supplementary information corresponding to input speech such as retrieval formulas (queries) to form ”examples” that decide system action. A system using this technique cannot run effectively, however, without a large volume of example data. Traditionally, though, collecting human-to-human dialogue data and manually providing such supplementary information for each instance of input speech has required considerable labor.

In this paper, we address this problem and propose a new technique for constructing an example-based dialogue system using, as example data, the dialogue performed between a human subject and a pseudo-spoken-dialogue system based on the Wizard of OZ (WOZ) scheme. We also describe a specific spoken dialogue system for information retrieval that we constructed using this technique.

2 Dialogue Processing Based on Examples

We first provide an overview of example-based dialogue processing that we previously proposed (Murao et al., 2001).

2.1 Model of information retrieval dialogue

Given a scenario in which a human operator searches an information database and returns information to a user, dialog between the operator and user can be modeled as shown in Fig. 1. The elements of this model are described below.

1. **Request** The user tells the operator the contents of an inquiry and demands reference.

2. **Retrieval** The operator receiving the user’s request generates a query after referencing domain knowledge and current dialogue context

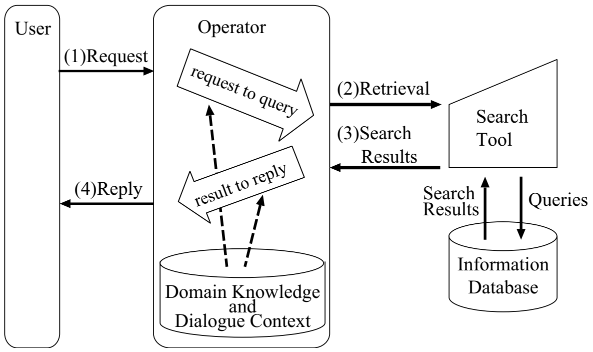


Figure 1: Information flow of information retrieval dialogue

and then processes the query indirectly by manipulating a search tool such as an ordinary computer.

3. **Search results** The search tool generates search results.

4. **Reply** The operator returns a reply to the user based on search results and dialogue context.

Setting up information flow in this way allows us to view operator behavior in the following way. Specifically, the operator in Fig. 1 makes two decisions in the process of advancing dialog.

Decision 1: Generate a query after listening to user speech

Decision 2: Generate a reply after receiving search results

Here, an experienced operator would use more than just the superficial information obtained from user speech. To generate a query or reply that best suits the user's need at that time, the operator would also make use of domain knowledge, dialogue context, and the search results themselves. In other words, this kind of dialogue processing can be viewed as a mapping operation from input information such as user speech and domain knowledge to output information such as a query. With this in mind, we considered whether a "decision" to guide such dialogue could be automatically performed by referring to actual examples of behavior manifested by an experienced human operator. In short, we decided to store a large volume of dialogue examples, i.e., mapping information, and to determine output information for certain input information on the basis of mapping information stored in similar dialogue examples.

2.2 Generation of queries and replies based on examples

2.2.1 Structure of example data

The two "decisions" performed during the time of information retrieval dialogue between the user and operator can be expressed as a mapping between the following input and output information.

- Input/output information in the decision for generating a query:

  Input User speech and dialogue context

  Output Query

- Input/output information in the decision for generating a reply:

  Input User speech, dialogue context, and search results

  Output Reply

It is therefore sufficient to save those items that cover such input and output information. Specifically, a large number of example data can be collected using the following information as elements to construct an example database.

1. Text of user speech

2. Query

3. Reply text

4. Search results

5. Dialogue context (past speech, grounding information, conversational objects , etc.)

The following describes the procedure for generating a query or reply with respect to input speech by referencing an example database.

2.2.2 Query generation process

From among the examples in the example database, the system extracts the one most similar to the input speech and the dialogue context at that time. It then adjusts the query in that example to fit the input speech and generates a new query.

2.2.3 Reply generation process

The system performs a search based on the generated query and receives search results. It then extracts the most similar example from the example database with respect to input speech, the dialogue context at that time, and the search results. Finally, the system adjusts the reply in that example to fit the current conditions and generates a new reply.

2.3 Problem points

Operating a dialogue system based on dialogue examples requires the construction of an example database as described above. Constructing a large-scale example database, moreover, requires a large volume of dialogue text in which supplementary information such as queries and search results has been provided with respect to input speech.

Up to now, we have been constructing an example database by first collecting human-to-human dialogue and converting speech to text and then assigning queries, search results, and the like to each instance of input speech. This, however, is a laborious process. In addition, example data constructed on the basis of human-to-human dialogue data may have features different from those of human-to-dialogue-system dialogue data. In other words, we cannot call the above approach an optimal method for constructing example data.

3 Construction of an Example Database using the WOZ System

We propose the Wizard of OZ (WOZ) system as one means of efficiently collecting dialogue data that includes supplementary information attached to speech. Carrying on a dialogue using WOZ makes it possible to collect the information needed for constructing an example database while collecting speech data.

3.1 WOZ system

When carrying on a dialogue using the WOZ system, the user feels that he or she is talking to a completely mechanical system despite the fact that a human being is actually being used for some of the elements making up the dialogue system. Collecting dialogue data by WOZ should therefore result in data that is closer to dialogue that would occur between a human and a machine.

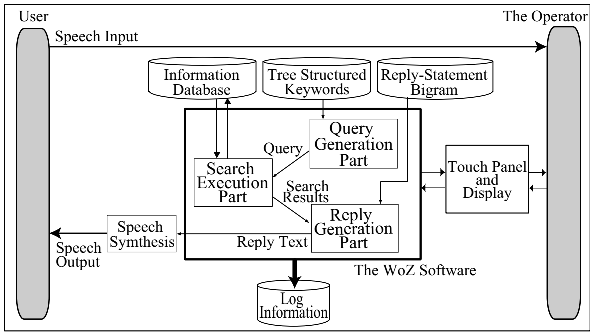


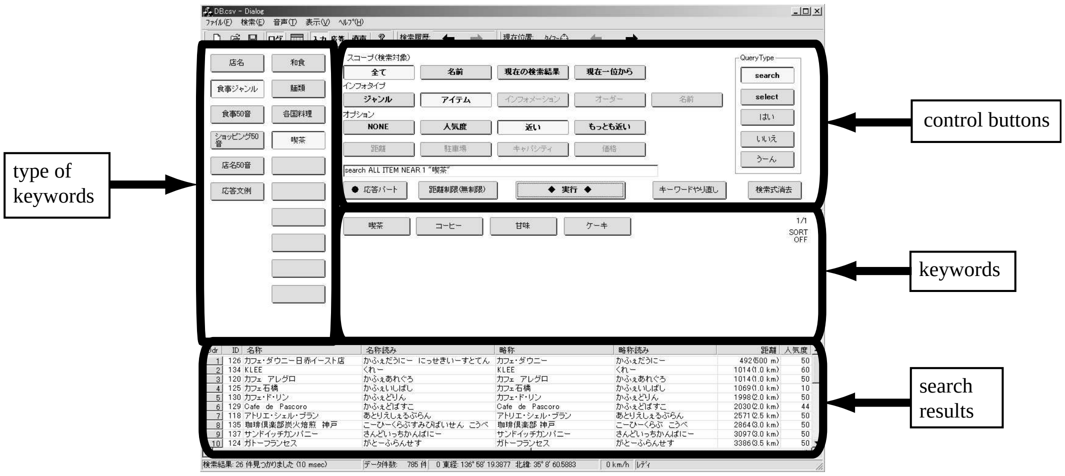
Figure 2: Configuration of Wizard of OZ system

Collecting spoken dialogue data using the WOZ system has actually been performed a number of times in the past (MADCOW, 1992; Bertenstam et al., 1995; Life et al., 1996; Eskenazi et al., 1999; San-Segundo et al., 2001; Lemmela and Boda, 2002; Yoma et al., 2002). The objective of those studies, however, was to collect, analyze, and evaluate dialogue data between people and artificial objects, and in many cases, only one of the artificial-object's functions was taken over by a human, for example, the speech recognition function.

Our study, however, goes further than the above. In particular, we create special software (called WOZ software) that allows a human being to perform the functions of interpreting user speech, generating queries and executing searches, and generating replies. We then propose a framework that enables the operator (wizard) to carry on a dialogue with the user while operating this WOZ software so that obtained data can be used later to perform direct control of a dialogue system. Specifically, we configure a pseudo-spoken-dialogue system (WOZ) consisting of WOZ software and an operator, hold information retrieval dialogue between this system and human subjects, and save the queries ,search results and reply statements generated at this time as log information. We then use this log information and text-converted speech to construct an example database that can be used for dialogue control.

3.2 System configuration

Figure 2 shows the entire configuration of the WOZ system that we constructed. In this configuration,

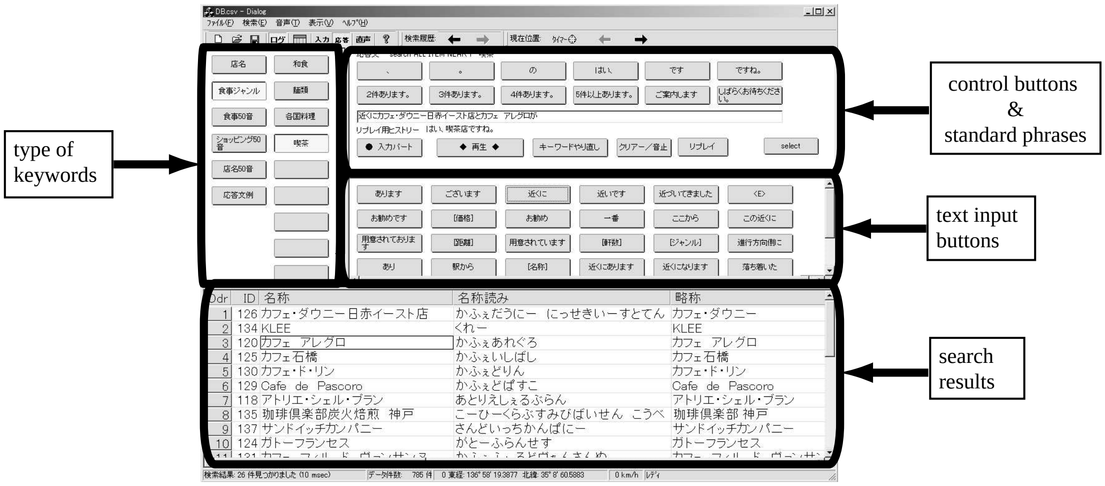


search ALL ITEM NEAR 1 "喫茶"

Odr	ID	名称	名称読み	略称	略称読み	距離	人気度
1	126	カフェ・ダウニー日赤イースト店	かふぇだうにー にっせきいーすとてん	カフェ・ダウニー	かふぇだうにー	492(500 m)	50
2	134	KLEE	くれー	KLEE	くれー	1014(1.0 km)	60
3	120	カフェ アレグロ	かふぇあれぐろ	カフェ アレグロ	かふぇあれぐろ	1014(1.0 km)	50
4	125	カフェ石橋	かふぇいしばし	カフェ石橋	かふぇいしばし	1069(1.0 km)	10
5	130	カフェ・ド・リン	かふぇどりん	カフェ・ド・リン	かふぇどりん	1998(2.0 km)	50
6	129	Cafe de Pascoro	かふぇどぱすこ	Cafe de Pascoro	かふぇどぱすこ	2030(2.0 km)	44
7	118	アトリエ・シェル・ブラン	あとりえしぇるぶらん	アトリエ・シェル・ブラン	あとりえしぇるぶらん	2571(2.5 km)	50
8	135	珈琲倶楽部炭火焙煎 神戸	こーひーくらぶすみびばいせん こうべ	珈琲倶楽部 神戸	こーひーくらぶ こうべ	2864(3.0 km)	50
9	137	サンドイッチカンパニー	さんどいっちかんぱにー	サンドイッチカンパニー	さんどいっちかんぱにー	3097(3.0 km)	50
10	124	ガトーフランセス	がとーふらんせす	ガトーフランセス	がとーふらんせす	3386(3.5 km)	50

検索結果: 26 件見つかりました (10 msec) | データ件数: 785 件 | 0 東経: 136° 58' 19.3877 北緯: 35° 8' 60.5883 | 0 km/h | レディ

Figure 3: An example of display of Wizard of OZ system (1): Query generation part



近くにカフェ・ダウニー日赤イースト店とカフェ アレグロが

リプレイ用ヒストリー はい、喫茶店ですね。

Odr	ID	名称	名称読み	略称
1	126	カフェ・ダウニー日赤イースト店	かふぇだうにー にっせきいーすとてん	カフェ・ダウニー
2	134	KLEE	くれー	KLEE
3	120	カフェ アレグロ	かふぇあれぐろ	カフェ アレグロ
4	125	カフェ石橋	かふぇいしばし	カフェ石橋
5	130	カフェ・ド・リン	かふぇどりん	カフェ・ド・リン
6	129	Cafe de Pascoro	かふぇどぱすこ	Cafe de Pascoro
7	118	アトリエ・シェル・ブラン	あとりえしぇるぶらん	アトリエ・シェル・ブラン
8	135	珈琲倶楽部炭火焙煎 神戸	こーひーくらぶすみびばいせん こうべ	珈琲倶楽部 神戸
9	137	サンドイッチカンパニー	さんどいっちかんぱにー	サンドイッチカンパニー
10	124	ガトーフランセス	がとーふらんせす	ガトーフランセス

検索結果: 26 件見つかりました (10 msec) | データ件数: 785 件 | 0 東経: 136° 58' 19.3877 北緯: 35° 8' 60.5883 | 0 km/h | レディ

Figure 4: An example of display of Wizard of OZ system (2): Reply generation part

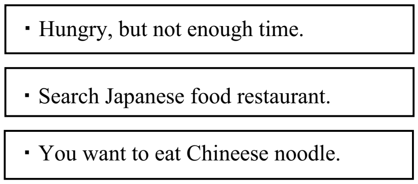
the WOZ software, which was created using the C++ language, runs on a personal computer under Windows2000. It consists of a screen for generating queries (query part) and a screen for generating replies (reply part). Figures 3 and 4 show sample screens of these parts. This GUI adopts a touch-panel system to facilitate operations — an operator only has to touch a button on one of these screens to generate a query, search an information database, generate a reply, or output synthesized speech.

WOZ software must feature high operability to achieve natural dialogue between the WOZ system and a human user. When designing WOZ software on the basis of a human-to-human dialogue corpus that we previously collected, we used the following techniques to enable the system to operate in real time while carrying on a dialogue with the user.

First, the query part arranges keywords in a tree structure by search type so that appropriate keywords can be selected at a touch to generate a query and retrieve information quickly [1] . Search results are displayed at the bottom of the screen in list form.

Second, the reply part displays text-input buttons for generating replies and a list of search results. The text-input buttons correspond to words, phrases, and short standard sentences, and pushing them in

[1] Queries that deal with context in regard to input speech are currently not defined for the sake of simplicity in software operation.



・Hungry, but not enough time.

・Search Japanese food restaurant.

・You want to eat Chineese noodle.

Figure 5: Examples of prompting panels

an appropriate order generates a reply in text form. The arrangement of these text-input buttons on the screen is based on connection frequency between text elements (reply-statement bigram) as previously determined from the human-to-human dialogue corpus mentioned above. In other words, each text-input button represents a text entry having the highest frequency of following the immediately previous text entry to the left, which makes for quick generation of a reply. Furthermore, to enable quick input, the section of the screen displaying the search results has been designed so that the name portion of each result can be touched directly and automatically included in the reply. The generated reply in text form is finally output in voice form via the speech synthesis section of the system.

Switching back and forth between the query and reply parts can be performed as needed using a switch button. The reply part also includes buttons for instantly generating words and short phrases of confirmation and encouragement (e.g., ”yes,” ”I see”) while the user is speaking to create as natural a dialogue as possible.

3.3 Collecting dialogue data by the WOZ system

We targeted shop-information retrieval while driving a car as an information-retrieval application based on spoken dialogue, and collected dialogue data between the WOZ system and human subjects (Kawaguchi et al., 2002). This data was collected within an automobile driven by subjects each of whom acted as a user searching for information. A personal computer running the WOZ software was placed in the automobile with the ”wizard” sitting in the back seat. All spoken dialogue was recorded using another personal computer.

Data collection was performed according to the following procedure for a duration of about five minutes per subject.

Table 1: Collected WOZ data

Number of sessions	Speech length (min.)		Speech Units	
	User	WOZ	User	WOZ
487	499	791	13,828	12,487

- A prompting panel such as shown in Fig. 5 is presented to the subject.
- The subject converses freely with WOZ based on the prompting panel shown.

The wizard operates the WOZ system while listening to the subject, that is, the wizard performs an appropriate search and returns a reply using speech synthesis [2] .

Table 1 shows the scale of collected data.

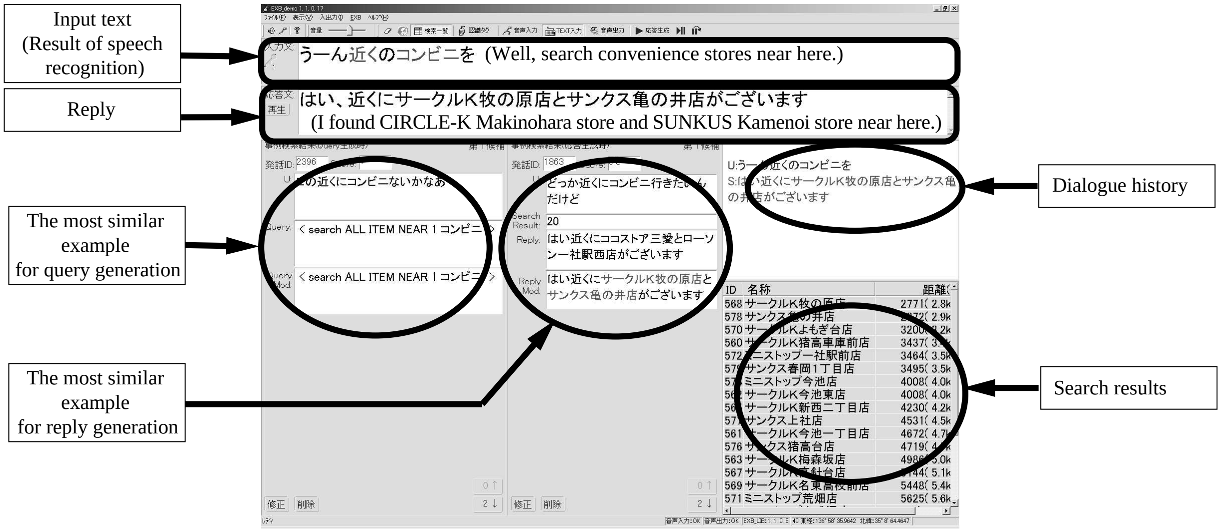
3.4 Constructing an example database using WOZ log information

WOZ software was designed to output detailed log information. This information consists mainly of the following items. All log information is recorded with time stamps.

- Speaker ID (input by the wizard when initiating a dialogue)
- Query generated for the input speech in question
- Search results returned for the generated query (number of hits and shop IDs)
- Text of reply generated by the operator (wizard)

A saved WOZ log can be used to efficiently construct an example database by the following procedure. To begin with, a written record of user speech is made based on the voice recording of spoken dialog with time information added. Next, based on

[2] The wizard generates queries, performs searches, and generates replies to the extent possible for speech to which defined queries can be applied. If a query cannot be generated, the wizard will not keep trying and will generate only an appropriate response.


Input text
(Result of speech recognition)
Reply
The most similar example for query generation
The most similar example for reply generation
Dialogue history
Search results
入力文
応答文
再生
事例検索結果(Query生成時)
第1候補
事例検索結果(応答生成時)
第1候補
修正
削除
0 ↑
2 ↓

うーん近くのコンビニを (Well, search convenience stores near here.)

はい、近くにサークルK牧の原店とサンクス亀の井店がございます
(I found CIRCLE-K Makinohara store and SUNKUS Kamenoi store near here.)

発話ID: 2396

U: この近くにコンビニないかなあ

Query: < search ALL ITEM NEAR 1 コンビニ >

Query Mod: < search ALL ITEM NEAR 1 コンビニ >

発話ID: 1863

U: どっか近くにコンビニ行きたいんだけど

Search Result: 20

Reply: はい近くにココストア三愛とローソン一社駅西店がございます

Reply Mod: はい近くにサークルK牧の原店とサンクス亀の井店がございます

U:うーん近くのコンビニを
S:はい近くにサークルK牧の原店とサンクス亀の井店がございます

ID	名称	距離(
568	サークルK牧の原店	2771(2.8k
578	サンクス亀の井店	2872(2.9k
570	サークルKよもぎ台店	3200(3.2k
560	サークルK猪高車庫前店	3437(3.4k
572	ミニストップ一社駅前店	3464(3.5k
579	サンクス春岡1丁目店	3495(3.5k
57[illegible]	ミニストップ今池店	4008(4.0k
56[illegible]	サークルK今池東店	4008(4.0k
56[illegible]	サークルK新西二丁目店	4230(4.2k
57[illegible]	サンクス上社店	4531(4.5k
561	サークルK今池一丁目店	4672(4.7k
576	サンクス猪高台店	4719(4.7k
563	サークルK梅森坂店	4986(5.0k
567	サークルK高針台店	5144(5.1k
569	サークルK名東高校前店	5448(5.4k
571	ミニストップ荒畑店	5625(5.6k

音声入力:OK 音声出力:OK EXB_LIB:1, 1, 0, 5 40 東経:136° 58' 35.9642 北緯:35° 8' 64.4647

Figure 6: A view of example-based dialogue system

Table 2: Configuration of constructed example database

Number of sessions	Number of examples
243	1,206

the time information in the log output by WOZ software, a correspondence is established between user speech and queries and between search results and replies.

We constructed an example database using a portion of dialogue data collected in the above manner. Table 2 summarizes the data used for this purpose.

Query and search-result correspondences were established for about 20% of all user speech excluding speech outside of the task in question and speech outside of query specifications.

4 Spoken Dialogue System using Dialogue Examples

We here describe a dialogue system that runs using the example database that we constructed (see (Murao et al., 2001) for details). The task is to search for shop information while inside an automobile. This system was implemented using the C++ language under Windows2000. Figure 6 shows a screen shot of this example-based dialogue system.

4.1 System configuration

The following describes the components of this system with reference to Fig. 7.

Dialogue example database (DEDB): Consists of data constructed from dialogue text and log information output from WOZ software. Dialogue text is subjected to morphological analysis [3], and words essential to advancing the dialogue (e.g., shop name, facility name, food name) are assigned word class tags based on classes given to these words beforehand according to meaning.

Word Class Database (WCDB): Consists of words essential to the task in question and classes given to them according to meaning. Word classes are determined empirically based on dialogue within the dialogue corpus.

Shop Information Database (SIDB): Consists of a collection of information on about 800 restaurants and shops in Nagoya, the same as that used in the WOZ system.

Speech Recognition: Uses "Japanese Dictation Toolkit(Kawahara et al., 2000)". The language model was created from the previously collected human-to-human dialogue corpus.

[3] Using ChaSen morphological-analysis software for the Japanese language (Asahara and Matsumoto, 2000).

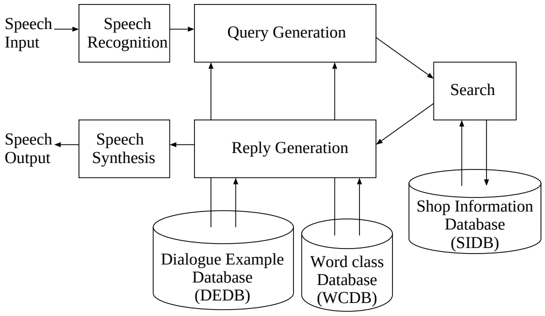


Figure 7: Configuration of example-based dialogue system

Query Generation: Extracts from the DEDB the example closest to current input speech and conditions, modifies the query in that example according to current conditions, and outputs the result.

Search execution: Accesses the SIDB using the generated query and obtains search results.

Reply Generation: Extracts from the DEDB the example closest to input speech and search results, modifies the reply in that example according to current conditions, and outputs the result.

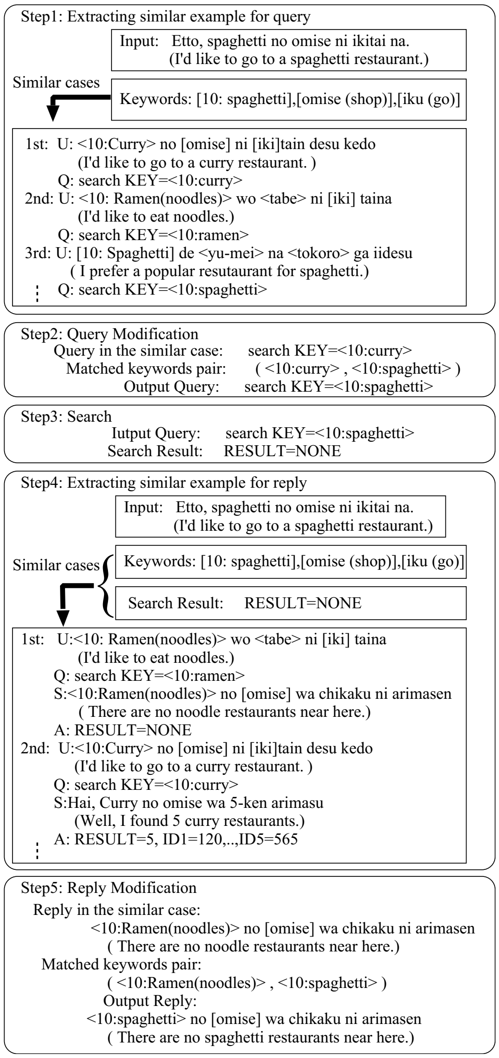
Speech Synthesis: Outputs replies in voice form using a Japanese TTS (Text To Speech) software “EleganTalk Ver. 2.1” by Sanyo Electric Co., Ltd. .

4.2 Operation

The following describes system operation (see Fig. 8 for a specific operation example).

Step 1: Extracting similar example for query For a speech recognition result, the system extracts the most similar example from the DEDB. The robustness of the similarity calculation between the input utterance and the utterance in the DEDB should be considered against the speech recognition error. Therefore, a keyword matching method using the word class information is adopted. For a speech recognition result combined with a morphological analysis result, independent words and the important words to which the word class tags are assigned according to the information in the WCDB are regarded as the keywords, and their similarity is calculated as follows. For each transcription of a user’s utterances in the DEDB, the number of matched words and the number of important words which belong to the same word class are accumulated with the correspondent weight and the result is treated as the similarity. The utterance which marks the highest similarity is regarded as the most similar one.

Step2: Query Modification The query for the extracted example is modified with reference to


Step1: Extracting similar example for query
Input: Etto, spaghetti no omise ni ikitai na.
(I'd like to go to a spaghetti restaurant.)
Similar cases
Keywords: [10: spaghetti],[omise (shop)],[iku (go)]
1st: U: <10:Curry> no [omise] ni [iki]tain desu kedo
(I'd like to go to a curry restaurant.)
Q: search KEY=<10:curry>
2nd: U: <10: Ramen(noodles)> wo <tabe> ni [iki] taina
(I'd like to eat noodles.)
Q: search KEY=<10:ramen>
3rd: U: [10: Spaghetti] de <yu-mei> na <tokoro> ga iidesu
(I prefer a popular resutaurant for spaghetti.)
⋮ Q: search KEY=<10:spaghetti>

Step2: Query Modification
Query in the similar case: search KEY=<10:curry>
Matched keywords pair: (<10:curry> , <10:spaghetti>)
Output Query: search KEY=<10:spaghetti>

Step3: Search
Iutput Query: search KEY=<10:spaghetti>
Search Result: RESULT=NONE

Step4: Extracting similar example for reply
Input: Etto, spaghetti no omise ni ikitai na.
(I'd like to go to a spaghetti restaurant.)
Similar cases
Keywords: [10: spaghetti],[omise (shop)],[iku (go)]
Search Result: RESULT=NONE
1st: U:<10: Ramen(noodles)> wo <tabe> ni [iki] taina
(I'd like to eat noodles.)
Q: search KEY=<10:ramen>
S:<10:Ramen(noodles)> no [omise] wa chikaku ni arimasen
(There are no noodle restaurants near here.)
A: RESULT=NONE
2nd: U:<10:Curry> no [omise] ni [iki]tain desu kedo
(I'd like to go to a curry restaurant.)
Q: search KEY=<10:curry>
S:Hai, Curry no omise wa 5-ken arimasu
(Well, I found 5 curry restaurants.)
⋮ A: RESULT=5, ID1=120,..,ID5=565

Step5: Reply Modification
Reply in the similar case:
<10:Ramen(noodles)> no [omise] wa chikaku ni arimasen
(There are no noodle restaurants near here.)
Matched keywords pair:
(<10:Ramen(noodles)> , <10:spaghetti>)
Output Reply:
<10:spaghetti> no [omise] wa chikaku ni arimasen
(There are no spaghetti restaurants near here.)

Figure 8: Example of query and reply generation

the input utterance. The modification is performed by replacing the keywords in the reference query using word class information.

Step 3: Search The SIDB is searched by using the modified query and a search result is obtained.

Step 4: Extracting similar example for reply The system extracts the most similar example from the DEDB, by taking account of not only the similarity between the input utterance and the utterance in examples but also that between the number of items in the search result and that in the examples. Here, a total similarity score is computed by performing a weighted summation of two values: the utterance similarity score and the search-results similarity score obtained from the difference between the number of search results in an example and that obtained in Step 3. The search-results similarity score is computed as follows.

When the number of search results by modified query is 0: Give the highest score to examples in the example database with 0 number of search results and the lowest score to all other examples.

When the number of search results by modified query is 1 or more: Give the highest score to examples in the example database with the same number of search results and an increasingly lower score as difference in the number of search results becomes larger (use heuristics).

For example, if not even one search result could be obtained by the modified query, examples in the example database with not even one search result constitute a match.

Step 5: Reply Modification The reply statement for the extracted example is modified with reference to the input utterance. The modification is performed by replacing the words in the reference reply statement by using word class information. Then a speech synthesis module is used to produce a reply speech.

4.3 Adding, modification, and deletion of example data

This system allows example data to be added, modified, and deleted. When a failed operation occurs while carrying on a dialogue, for example, buttons located at the bottom of the screen can be used to modify existing example data, add new examples, and delete unnecessary examples.

5 Conclusion

This paper has proposed an efficient technique for collecting example data using the Wizard of OZ (WOZ) system for the purpose of guiding spoken dialogue using dialogue examples. This technique has the following effects.

- Knowledge buried in the WOZ system log (conversions from input speech to query and reply, etc.) can be used as dialogue system knowledge.

- Because dialogue is collected using the WOZ system, the examples so collected are close to dialogue that would occur in an environment with an actual dialogue system. In other words, dialogue examples can be collected under conditions close to human-to-machine dialogue.

- The labor involved in recording speech necessary for construction of an example database can be reduced.

In future research, we plan to evaluate dialogue-processing performance and context processing using example databases constructed with the WOZ system.

References

Masayuki Asahara and Yuji Matsumoto. 2000. Extended models and tools for high-performance part-of-speech tagger. In *Proceedings of COLING 2000*, July.

J. Bertenstam, M. Blomberg, R. Carlson, K. Elenius, B. Granstrom, J. Gustafson, S. Hunnicutt, J. Hogberg, R. Lindell, L. Neovius, A. de Serpa-Leitao, L. Nord, and N. Strom. 1995. The waxholm application database. In *Proceedings of Eurospeech-95*, volume 1, pages 833–836.

Maxine Eskenazi, Alexander Rudnicky, Karin Gregory, Paul Constantinides Robert Brennan, Christina Bennett, and Jwan Allen. 1999. Data collection and processing in the carnegie mellon communicator. In *Proceedings of Eurospeech-99*, volume 6, pages 2695–2698.

Nobuo Kawaguchi, Shigeki Matsubara, Kazuya Takeda, and Fumitada Itakura. 2002. Multi-dimensional data acquisition for integrated acoustic information research. In *Proc. of 3rd International Language Resources and Evaluation Conference (LREC-2002)*, pages 2043–2046.

T. Kawahara, A. Lee, T. Kobayashi, K. Takeda, N. Minematsu, S. Sagayama, K. Itou, A. Ito, M. Yamamoto, A. Yamada, T. Utsuro, and K. Shikano. 2000. Free software toolkit for japanese large vocabulary continuous speech recognition. In *Proceedings of ICSLP-2000*, volume 4, pages 476–479.

Saija-Maaria Lemmela and Peter Pal Boda. 2002. Efficient combination of type-in and wizard-of-oz tests in speech interface development process. In *Proceedings of ICSLP-2002*, pages 1477–1480.

Esther Levin, Roberto Pieraccini, and Wieland Eckert. 1998. Using markov decision processes for learning dialogue strategies. In *Proceedings of ICASSP98*, volume 1, pages 201–204.

A. Life, I. Salter, J.N. Temem, F. Bernard, S. Rosset, S. Bennacef, and L. Lamel. 1996. Data collection for the mask kiosk: Woz vs prototype system. In *Proceedings of ICSLP-96*, pages 1672–1675.

MADCOW. 1992. Multi-site data collection for a spoken language corpus. In *DARPA Speech and Natural Language Workshop '92*.

Hiroya Murao, Nobuo Kawaguchi, Shigeki Matsubara, and Yasuyoshi Inagaki. 2001. Example-based query generation for spontaneous speech. In *Proceedings of 2001 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU2001)*.

R. San-Segundo, J.M. Montero, J.M. Gutierrez, A. Gallardo, J.D. Romeral, and J.M. Pardo. 2001. A telephone-based railway information system for spanish: Development of a methodology for spoken dialogue design. In *Proceedings of SIGdial-2001*, pages 140–148.

Nestor Becerra Yoma, Angela Cortes, Mauricio Hormazabal, and Enrique Lopez. 2002. Wizard of oz evaluation of a dialogue with communicator system in chile. In *Proceedings of ICSLP-2002*, pages 2701–2704.

Steve Young. 2002. Talking to machines (statistically speaking). In *Proceedings of ICSLP-2002*, pages 9–16.