EXAMPLE-BASED QUERY GENERATION FOR SPONTANEOUS SPEECH

Hiroya Murao[1,2], *Nobuo Kawaguchi*[1,3], *Shigeki Matsubara*[1,4] *and Yasuyoshi Inagaki*[5]

[1]Center for Integrated Acoustic Information Research, Nagoya University
[2]Hypermedia Research Center, Sanyo Electric Co. Ltd
[3]Computation Center, Nagoya University
[4]Faculty of Language and Culture, Nagoya University
[5]Department of Computational Science and Engineering, Nagoya University

[1,3,4,5]Furo-cho, Chikusa-ku, Nagoya, 464-8601, JAPAN
[2]1-8-13, Hashiridani, Hirakata, Osaka, 573-8534, JAPAN
murao@hr.hm.rd.sanyo.co.jp

ABSTRACT

This paper proposes a new query generation method that is based on examples of human-to-human dialogue. Along with modeling the information flow in dialogue, a system for information retrieval in car has been designed. The system refers to the dialogue corpus to find an example that is similar to input speech, and makes a query from the example. We also give the experimental results to show the effectiveness of this method.

1. INTRODUCTION

The models for spoken dialogue processing have been constructed using state-transition, frame and so on [1]. It is difficult for such the model to cover all the various phenomena in the spontaneously spoken dialogue. Recently, to overcome the difficulty, the models based on the dialogue corpus have been used for semantic analysis of spoken language or dialogue strategies optimization. It has been shown that such models are effective for the spontaneous speech understanding[2][3][4].

In this paper, we propose a framework to construct a information retrieval dialogue system using a dialogue corpus. In this framework, the utterances stored in the dialogue corpus are used as examples. And the actions of the system are determined by those examples. Since the aim of the user in the information retrieval dialogue is to create a query corresponding to user's requests, we can say that the process creating a query is nothing but a mapping operation from the input utterance to the query. That is, we think that using the pair of input utterance and output query as the example, the query corresponding to user's input can be generated. For the purpose of the implementation and the evaluation of a robust spoken dialogue system whose task is shop information retrieval in a car, we are currently collecting the data of spontaneously spoken dialogue in a moving car environment[5]. Using this data, the examples database is constructed and the dialogue system is designed.

In the following sections, we look at the informational flow in a information retrieval dialogue to model the dialogue, then propose the query and reply generation method based on the dialogue examples. And we describe the design of the prototype system based on the technique, and report the evaluation of the system.

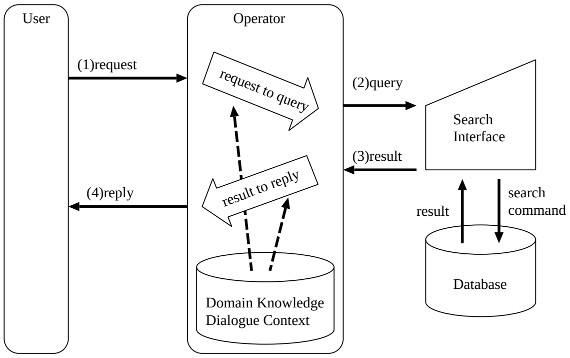


Fig. 1. Information flow of information retrieval dialogue

2. EXAMPLE-BASED DIALOGUE

2.1. Dialogue Model

Before considering a human-to-machine dialogue, let us try to model a human-to-human dialogue. Fig.1 shows the informational flow in the information retrieval dialogue between a user and a human operator.

1. **Request** Receiving the user's request, the operator generates a database query according to the current dialogue context.
2. **Search** The operator performs the search based on the query.
3. **Search results** A search result is generated.
4. **Response** The operator responds to the user according to the dialogue context and the search result.

As Fig.1 shows, the operator does the following two decisions:

1. Generating a search query from the user's utterance.
2. Responding to the user on the basis of the search result.

The skilled operator is considered to use a domain knowledge, dialogue context, past experience, etc., to make a "decision" to find out what to do for a user's request.

The operator uses not only the surface-information which appears in user’s utterance but also the various information such as domain knowledge or dialogue context to perform the operation suitable for the user’s purpose. However, it is difficult to make rules completely for such a “decision”. So we can say that it is effective to make such a “decision” with reference to the examples which the skilled human operator performed.

2.2. Example-based Query and Reply Generation

As Fig.1 shows, to design an example-based dialogue system, it is required to fix the process of query and reply generation, and the form of examples. In “example-based dialogue”, which we propose in this paper, they are described as follows:

- **Construction of the examples database** The dialogues between the users and the operator are collected, with the operations performed at that time. The two actions to generate a query and a reply can be determined with the following information:

  Info A: For the decision of query generation

  1. user’s utterance
  2. context of dialogue

  Info B: For the decision of reply generation

  1. user’s utterance
  2. context of dialogue
  3. search result

  Therefore, the examples database should have 5 kinds of information: 1)user’s utterances, 2)search queries, 3)operator’s utterances, 4)results of the search, and 5)context information (past requests, past replies, past search results).

- **Query Generation Process** (“request to query” arrow in Fig.1) For a user’s request, the most similar example in the examples database is picked up concerning Info A. Then the query in the example is corrected so that it may be suited for the present situation. And a search is performed by the query.

- **Reply Generation Process** (“result to reply” arrow in Fig.1) For the search result, the most similar example in the examples database is picked up concerning Info B. Then the reply statement in the example is corrected so that it may be suited for the present situation.

3. IN-CAR SHOP INFORMATION SYSTEM

We have implemented the prototype system based on our proposed idea. As the first step of the development, we targeted an operation for the context independent utterances.

3.1. System Configuration

The configuration of the system is shown in Fig.2.

- **Dialogue Examples Database(DEDB)** The dialogue examples database has been constructed on the CIAIR-HCC (CIAIR spoken language dialogue corpus)[5]. For each utterance for a user’s request, a search query corresponding to the utterance is recorded. A search query consists of keywords to search the SIDB. And for each utterance for the operator’s reply, the ID numbers of search results are recorded. The text is analyzed morphologically. Special words (store name, food name and so on) are classified semantically and assigned the word class tags in advance.

  Fig.3 shows a sample of the DEDB.

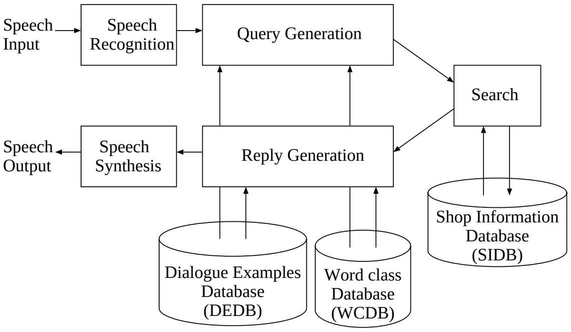
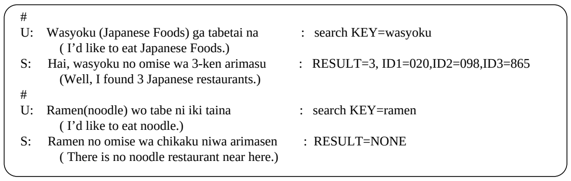


Fig. 2. System configuration



```
#
U:   Wasyoku (Japanese Foods) ga tabetai na          :  search KEY=wasyoku
         ( I’d like to eat Japanese Foods.)
S:   Hai, wasyoku no omise wa 3-ken arimasu          :  RESULT=3, ID1=020,ID2=098,ID3=865
         (Well, I found 3 Japanese restaurants.)
#
U:   Ramen(noodle) wo tabe ni iki taina              :  search KEY=ramen
         ( I’d like to eat noodle.)
S:   Ramen no omise wa chikaku niwa arimasen         :  RESULT=NONE
         ( There is no noodle restaurant near here.)
```

Fig. 3. Dialogue examples database (a part)

- **Word Class Database(WCDB)** This database consists of the important words classified semantically. We classified them based on a dialogue corpus experientially. The current number of classes is 43.

- **Shop Information Database(SIDB)** The restaurants, shops, gas stations, etc. in Nagoya city are registered. It is composed of about 800 places.

- **Speech Recognition** Japanese dictation toolkit[6], is used for Japanese speech recognition. The N-gram language model is created from the transcription of the dialogue speech.

- **Query Generation** The module extracts the example, which is the most similar to the input utterance, from the dialogue database. Then the query in the example is corrected so that it may be suited for the present situation.

- **Search** The search module accesses the SIDB and generates the search result.

- **Reply Generation** The module extracts the example, which is the most similar to the search result and the input utterance, from the dialogue database. Then the reply statement in the example is corrected so that it may be suited for the present situation.

- **Speech Synthesizer** The module synthesizes the sound of the reply statement.

3.2. The Procedure of Query and Reply Generation

We describe the behavior of the system in accordance with the example of Fig.4.

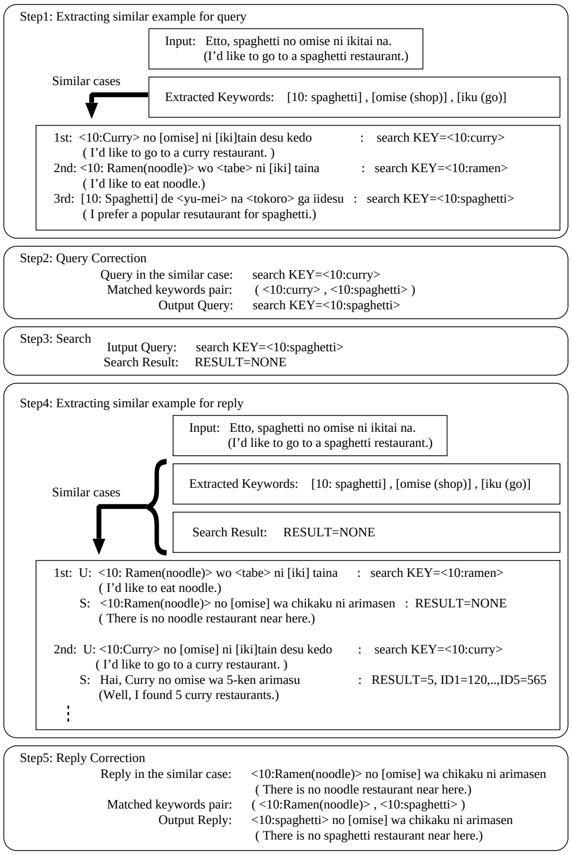

Step1: Extracting similar example for query
Input: Etto, spaghetti no omise ni ikitai na.
(I'd like to go to a spaghetti restaurant.)
Similar cases
Extracted Keywords: [10: spaghetti] , [omise (shop)] , [iku (go)]
1st: <10:Curry> no [omise] ni [iki]tain desu kedo : search KEY=<10:curry>
(I'd like to go to a curry restaurant.)
2nd: <10: Ramen(noodle)> wo <tabe> ni [iki] taina : search KEY=<10:ramen>
(I'd like to eat noodle.)
3rd: [10: Spaghetti] de <yu-mei> na <tokoro> ga iidesu : search KEY=<10:spaghetti>
(I prefer a popular resutaurant for spaghetti.)
Step2: Query Correction
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
Extracted Keywords: [10: spaghetti] , [omise (shop)] , [iku (go)]
Search Result: RESULT=NONE
1st: U: <10: Ramen(noodle)> wo <tabe> ni [iki] taina : search KEY=<10:ramen>
(I'd like to eat noodle.)
S: <10:Ramen(noodle)> no [omise] wa chikaku ni arimasen : RESULT=NONE
(There is no noodle restaurant near here.)
2nd: U: <10:Curry> no [omise] ni [iki]tain desu kedo : search KEY=<10:curry>
(I'd like to go to a curry restaurant.)
S: Hai, Curry no omise wa 5-ken arimasu : RESULT=5, ID1=120,..,ID5=565
(Well, I found 5 curry restaurants.)
⋮
Step5: Reply Correction
Reply in the similar case: <10:Ramen(noodle)> no [omise] wa chikaku ni arimasen
(There is no noodle restaurant near here.)
Matched keywords pair: (<10:Ramen(noodle)> , <10:spaghetti>)
Output Reply: <10:spaghetti> no [omise] wa chikaku ni arimasen
(There is no spaghetti restaurant near here.)

Fig. 4. Example of query and reply generation

Step 1: Extracting similar example for query For a speech recognition result, it extracts the most similar example from the DEDB. Considering the speech recognition error, we should take account of the robustness for the similarity calculation between the input utterance and that in examples. So, a keyword matching method with the word class information is adopted. For a speech recognition result with a morphological analysis result, the keyword is extracted selecting independent words and assigned the word class tag to the special words by the information of the WCDB. And the similarity is calculated as follows. For each transcription of user's utterances in the DEDB, the number of matched words and the number of special words which belong to the same word class, are accumulated with the correspondent weight. And the utterance which marks the highest point is regarded as the most similar one.

Step 2: Query correction The query for the extracted example is corrected corresponding to the input utterance. The correction is performed by replacing the keywords in the reference query using word class information.

Step 3: Search It searches the SIDB using the corrected query and gets a result of the search.

Step 4: Extracting similar example for reply It extracts the most similar example from the DEDB, considering not only the similarity between the input utterance and that in examples but also the number of searched item in the search result and that in examples. For example, if there is no item in the search result, it matches only the examples which have no item in the search result.

Table 1. Experimental parameters

Examples data	537 utterances by 44 speakers (context independent)
Test data	89 utterances by 20 speakers (context independent)
Informational database	785 items
Word class database	43 classes

Table 2. Classification for query evaluation

Class 1	Correct
Class 2	Partially correct
Class 3	Wrong
Class 4	Query generation failure (No matched example, or failed to keyword correction)

Step 5: Reply correction The reply statement for the extracted example is corrected corresponding to the input utterance. The correction is performed by replacing the words in the reference reply statement using word class information. And then speech synthesis module is used to produce a reply speech.

4. EVALUATION

We have evaluated the query generation part of the method using the context independent utterances. At first, to reveal the fundamental performance of the query generation part, the experiment on the transcribed user's utterance is performed. After that, we will see the relation between error rate of spontaneous speech recognition and the query generation performance.

4.1. An experiment for transcribed text input

Table 1 shows the experimental conditions. The evaluation is performed based on the following procedure, changing the number of utterances used in the DEDB.

1. Input the utterance transcription of the test data into the query generation part, and generate a query.
2. Classify the obtained query into four classes subjectively. (see Table 2.)

Fig.5 shows the experimental result. In the case with 537 examples, the correct queries (Class 1+2 in Table 2) were generated for about 88 % of the test data. Moreover, we can also see that the rate of the correct answers are improved in accordance with the number of examples.

4.2. An experiment for speech input

The system is required to have high performance in driving car environment, so the robustness against errors of speech recognition becomes important. To examine the relation between the error rate of speech recognition and the query generation performance, an experiment using speech input was performed. To simplify the

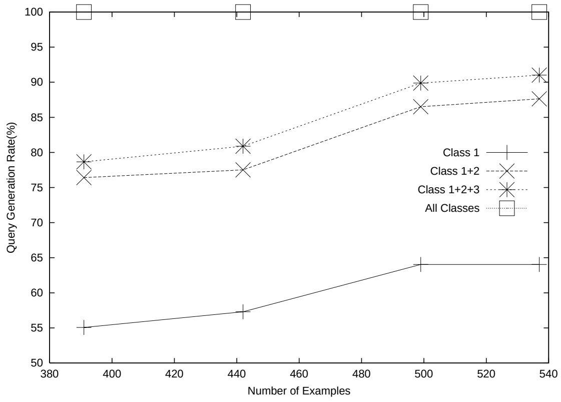


Fig. 5. The relation between the DEDB size and query generation rate (Transcribed text input)

Table 3. The main parameters of speech recognition module

Acoustic model	PTM triphone HMM, 3000 states, 64 mixtures,[6]
Language model (N-gram model)	CIAIR-HCC[5], 30,815 utterances by 106 speakers

issue, we used the reduced test data which contains only the utterances classified into the Class 1 and 2 in Table2 in the test of transcribed text input. The test data consists of 78 utterances. For these test data, the system can produce the correct query if the performance of speech recognition is sufficient.

The main conditions of speech recognition module is shown in Table3. We used "Japanese Dictation Toolkit 1999"[6] as speech recognizer. For our test data, word correct rate(WCR) is 62.17%. And keyword correct rate(KWCR), which is word correct rate for keywords to be extracted for similarity calculation, is 61.31%

The below is the procedure of the evaluation: For each of 78 test utterances, KWCR is calculated. And they are divided into 5 groups according to KWCR. The division rule is as follows:

$$\begin{array}{lrcccr}
\text{Group1:} & 0.00\% & \leq & \text{KWCR} & < & 1.00\% \\
\text{Group2:} & 1.00\% & \leq & \text{KWCR} & < & 33.00\% \\
\text{Group3:} & 33.00\% & \leq & \text{KWCR} & < & 67.00\% \\
\text{Group4:} & 67.00\% & \leq & \text{KWCR} & < & 100.00\% \\
\text{Group5:} & & & \text{KWCR} & = & 100.00\%
\end{array}$$

Then the query generation rate with 573 examples in the DEDB, is calculated for each groups. The total query generation rate for all 78 test utterances is 61.54%(Class 1) and 74.36%(Class 1+2).

The result is shown in Fig.6. Each data is plotted for x-axis in the value of the mean recognition rate of each 5 groups. From this data, we can see that, compared with degradation of the KWCR, the query generation rate is kept more highly. This exemplifies the high robustness of our method for errors of speech recognition.

5. CONCLUDING REMARKS

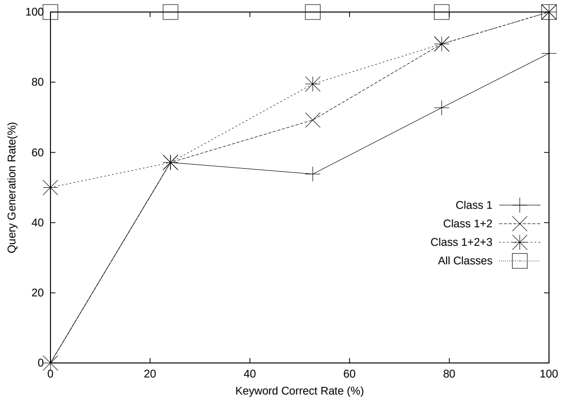


Fig. 6. The relation between word correct rate and query generation rate

In this paper, we have proposed the method of generating the query by using the practical human-to-human dialogues for information retrieval. The experimental results on the prototype system is as follows:

- For transcribed text input, it provides the correct query in about 88% rate.
- For the input of speech recognition result, it achieves relatively higher query generation rate compared with the keyword recognition rate.

These results have shown the method to be effective.

Acknowledgments: This work is partially supported by Grant-in-Aid for COE Research of the Ministry of Education, Science, Sports and Culture Japan. (No. 11CE2005)

6. REFERENCES

[1] D. Goddeau, H. Meng, Joe Polifroni, S. Seneff, and S. Busayapongchai, "A form-based dialogue manager for spoken language applications," in *Proceedings of ICSLP-96*, 1996, pp. 701–704.

[2] W. Minker, S. Bennacef, and J.L. Gauvain, "A stochastic case frame approach for natural language understanding," in *Proceedings of ICSLP-96*, 1996, pp. 1013–1016.

[3] M. Epstein, K. Papineni, S. Roukos, T. Ward, and S. Della Pietra, "Statistical natural language understanding using hidden clumpings," in *Proceedings of ICASSP-96*, 1996, pp. 176–179.

[4] E. Levin, R. Pieraccini, and W. Eckert, "A stochastic model of human-machine interaction for learning dialogue strategies," *IEEE Trans. on Speech and Audio Processing*, vol. 8, no. 1, pp. 11–23, Jan. 2000.

[5] N. Kawaguchi, S. Matsubara, H. Iwa, S. Kajita, K. Takeda, F. Itakura, and Y. Inagaki, "Construction of speech corpus in moving car environment," in *Proceedings of ICSLP-2000*, 2000, vol. 3, pp. 362–365.

[6] 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., "Free software toolkit for japanese large vocabulary continuous speech recognition," in *Proceedings of ICSLP-2000*, 2000, vol. 4, pp. 476–479.