Construction and Optimization of a Question and Answer Database for a Real-environment Speech-oriented Guidance System

Shota TAKEUCHI, Tobias CINCAREK, Hiromichi KAWANAMI, Hiroshi SARUWATARI, Kiyohiro SHIKANO
Graduate School of Information Science, Nara Institute of Science and Technology
8916-5, Takayama-cho, Ikoma-shi, Nara, Japan, 630-0192
Phone: +81-743-72-5287 Fax: +81-743-72-5289
Email: {shota-t, cincar-t, kawanami, sawatari, shikano}@is.naist.jp
URL: http://spalab.naist.jp

Abstract
In this paper, we describe a question and answer database (QADB), which is useful for constructing a spoken dialog system. For building a real-environment speech-oriented dialog system, real user data are required. In order to collect these data, we have been employing a spoken guidance system since five years. A database of transcriptions and appropriate responses, etc. has been constructed for 2 years of the collected utterances. Furthermore we introduce a QADB optimization method to exclude inappropriate data. Using this method, response accuracy improves by 1.6% to 75.7% absolute and by 2.9% to 59.8% about adults and children, respectively.

Keywords: Speech Dialog System, Example-based Response Generation, Optimization, Leave-one-out Cross-validation

1 Introduction
A spoken dialog system is a natural user interface for human-machine communication using speech. This kind of system employs speech recognition and dialog control technology to reply to user utterances. Both technologies can be developing by studying spoken dialog system. There are various kinds of spoken dialog system, for example, call-routing service [1], weather information guidance [2] and multi-modal interface [3].

In a real environment, e.g. a public hall, many users will talk to a spoken dialog system using their own natural language expressions. Moreover, utterances may be affected with background noise and reverberation. Therefore techniques to respond to various utterance expressions properly and to recognize speech in various situations accurately are necessary for a real environment spoken dialog system. In this paper, we investigate the former techniques for flexible dialog processing of various utterance expressions.

For building a reliable speech dialog system, dialog control techniques, for example, semantic understanding of input sentences and estimation of omitted words using dialog history, are necessary, but state-of-the-art techniques are not robust enough against speech recognition errors. Furthermore, for employing spoken dialog systems in a real environment, a speaker-independent model is used for speech recognition. A large amount of speech data is required to achieve high accuracy. Additionally, employing a dialog corpus can enhance the dialog control technology. However, there are few corpora of spontaneous utterances collected in a real environment. [1, 3, 4, 9]

In order to collect spontaneous user utterances for constructing a dialog corpus, we developed two speech-oriented guidance systems “Takemaru-kun” [5] and “Kita-chan.” They are installed at a public community center and a railway station, respectively. The systems employ a one question to one answer strategy for responding to user utterances. Using an animated agent, the purpose of these systems is mainly guidance of the facilities and to give the user the impression that the system has a personality.

Takemaru-kun has been operated since November 2002. A two-year database of utterance labels such as transcriptions and appropriate system responses has been constructed by listening to each utterance. More accurate recognition models can be built by using this database. Moreover, various methods for speech recognition in a real environment have been developed, e.g. speech recognition with age group distinction [5], noise rejection using GMM [6], unsupervised training of acoustic models [7], and recognition of preschool children speech [8].

The deployed speech-oriented guidance system uses example-based response generation in the dialog control module to cope with various utterances and expressions. This method retrieves the response corresponding to the example question most similar to the recognition result [9]. This method can generate a response easier and faster than analyzing the semantics of automatic recognition results. Furthermore, it is sufficient for the guidance task which requires an immediate
response.

Example-based response generation relies on a QADB (question and answer database) to generate appropriate responses. Errors in the QADB may cause an improper system response. Their inclusion is hard to suppress perfectly because the utterance database may contain label mistakes of human annotators.

Consequently, a method to construct an optimal QADB by excluding improper pairs in order to improve the response performance has been developed [10]. In this paper, we describe and evaluate the QADB optimization method.

2 Speech-oriented Guidance System “Takemaru-kun”

Takemaru-kun is a speech-oriented guidance system which is installed at the public community center. Each response is using text-to-speech, agent animation and either floor maps or web page as visual information. Guidance is conducted by presenting both a voice-based and a visual response, and it looks like the agent itself is answering by showing agent animation.

The system consists of a microphone, one set of loud speakers and two displays on a desk. The computer hardware for performing speech dialog processing is connected to the Internet through a gateway. Using this configuration, the state of the system can be supervised. All inputs including speech and noise are recorded and saved with a timestamp. Takemaru-kun does not respond to noise thanks to noise rejection [6].

There are 309 distinct responses for Takemaru-kun which can be classified into eight categories:

(a) Guidance at the public community center (center guidance, 64 topics)
(b) Guidance for the local area and sightseeing (local guidance, 86 topics)
(c) Time, weather, news (news, 8 topics)
(d) Access to important web sites (web-site, 21 topics)
(e) Profile of Takemaru-kun (profile, 52 topics)
(f) Greeting from Takemaru-kun (greeting, 7 topics)
(g) Answer of utterances except guidance request (chat, 68 topics)
(h) “Sorry, I don’t know” response. (out-of-domain, 3 topics)

Takemaru-kun has been employed since November 2002 to collect spontaneous utterances of real users continuously. System operation has never aborted, and the system’s contents have been updated several times.

3 Takemaru-kun Speech Database

In the following, the contents of the Takemaru-kun speech database are described.

3.1 Method to Arranging the Database

The recorded speech data were transcribed and labeled by several human annotators. All data from November 2002 to October 2004 are completely labeled and transcribed. Even noise data are classified as "noise."

The Collection of speech has been continued also after October 2004. Unlabeled speech data for additional 3 years is also available. These speech data have been employed for research on acoustic modeling using unsupervised training, e.g. selective training [7]. Moreover, the system stores the log of speech recognition results since the start of Takemaru-kun. Currently, the log is used for making statistics of the collected speech data. It is not used to for the database construction, but the log information is useful for research on spoken dialog system.

3.2 Contents of Takemaru-kun Speech Database

All input data are annotated with the following labels:

(a) Date: The name of the speech file. Also acts as identifiers of the collected data.
(b) Gender: Speakers’ gender classified by listening. Three classes of female, male, uncertain.
(c) Age group: Speakers’ age classified by listening. Six classes of infant (about 0-5-years-old), lower school child (about 6-10-years-old), higher school child (about 11-15-years-old), adult, elder, uncertain.
(d) Validity: Whether a speech is utterance to the system, is intelligible, and has meaning. Data with background speech or howling are
classified as invalid.

(e) Transcription (Hiragana-Kanji): Transcription in Japanese Hiragana-Kanji writing. About noise or unnecessary speech, the information is noted.

(f) Transcription (Katakana): Transcription with Japanese Katakana writing.

(g) Pronunciations (Katakana): Pronunciation in Japanese Katakana writing. Differing from katakana Transcription w.r.t. on long vowels.

(h) Appropriate system response: Most appropriate response to user input. Appending the response number of Takemaru-kun system. Represents indirectly the meaning of the user utterance - system response pair.

The annotators have not witnessed the speech collection, so that classifying the gender or age groups of some speech e.g. preschool children was not possible. There were many annotators. Each annotators processed distinct portions of the data. The response label is limited to only one system response, while some data had more than one response label.

Special tags are assigned to noise and invalid speech inputs. Only valid utterances (valid speech) are labeled with an appropriate response.

3.3 Database Statistics

Statistics for the Takemaru-kun speech database are shown in Table 1. One-third of the inputs are noise. From this result, the necessity of noise rejection for a spoken dialog system is clear.

When considering the relative share of age groups among the valid speech data, the most are from children [Table 2]. This result suggests that a deployed system needs more efforts to respond to children inputs properly. It is suggested that the reason for this distribution of the public community center is the system interface with an animated agent. This results in a majority of children inputs.

The classification of valid speech inputs into the eight response categories is shown in Figure 2. There are more utterances to talk with “Takemaru-kun” than guidance requests, which are assumed to be the domain of the system. Considering only children data, greetings to Takemaru-kun are relatively less than frequently for other age groups.

4 Example-based Response Generation

"Example-based response generation" is named after the strategy to select a fixed response for each user input using a set of example questions. The advantage is that semantic analysis is not necessary.

4.1 Proposed Approach

The user expects a useful spoken dialog system which gives appropriate and immediate responses. For this reason, precise and simple response generation is needed for the system.

There are many possible module configurations of a spoken dialog system. A standard spoken dialog system has modules for speech recognition, language understanding, dialog control, and response generation.

In a one-question-to-one-answer dialog system, however, it is not required to perform in-depth language understanding and dialog control since it is not necessary to deal with the dialog history. Thus the dialog system described in this paper generates a response using the speech recognition result directly.

Although this approach does not use a sophisticated dialog model, the QADB (pairs of example questions and responses) can be interpreted as simple dialog model.

4.2 Principle

The example-based response generation approach can cope with various user expressions by providing a large amount of user questions for each
system response. If the system has stored an appropriate response for an input, it can present the appropriate response if the same input occurs. This principle is called example-based response generation.

When using this method, many QA pairs (pair of example question and proper response) have to be collected to construct the QADB for system implementation. On employment, it retrieves the example question from the database, which is the most similar to the user input. The response corresponding to the retrieved example is presented as system output. This method performs only similarity calculation of each example for the input and retrieves the example with the highest similarity; therefore it is suitable for a real environment spoken dialog system.

The example-based response generation method for the current Takemaru-kun system has the following characteristics:

- Perform speech recognition to obtain plain text; generate a response using this sentence.
- Calculate the similarity using symmetric matching of word units.
- Nearest neighbour search: Select the QA pair of highest similarity.
- Extract QA pairs from the utterance database.

In Japanese, words of a sentence are not separated explicitly by spaces. Therefore a pre-processing with a morphological analyzer [11] is carried out before matching the example question with the recognition result.

4.3 QADB Construction

When constructing the QADB from examples of real user utterances, the system can properly retrieve an example for an utterance which a system developer would not have thought of. Moreover by searching not only for exact but also similar examples, it can select an example for an input which does not appear in the QADB.

Construction of the QADB is straight-forward: Extraction of the utterance transcription and the appropriate system response for each utterance from the Takemaru database. Duplicate pairs are excluded.

4.4 Algorithm Details

The method for example-based response generation in this paper consists of a scoring and a retrieval step. The former calculates the similarity of the input with each example. The latter determines the most similar example.

4.4.1 Scoring

In this section we describe the similarity calculation. The similarity is the normalized number of matching words in the input and the example question. This method does not use vector space model e.g. TF*IDF. In preliminary experiments we could show that a vector space model, e.g. based on TF*IDF weights, is inferior to the proposed method.

For calculating similarity, words which occur in the input and the example question simultaneously are counted. The word position is not considered.

The similarity is calculated by dividing the number of matched words by the maximum of words in the input and the example.

We define the set $I$ and $E$ of words corresponding to the input and the example question, the number of simultaneously occurring words $c(I, E)$, and the denominator for normalization $N(I, E)$.

$$I : \text{set of input words (recognition result)},$$
$$E : \text{set of words in the example question},$$
$$|I| : \text{number of words in set I},$$

The similarity $s(I, E)$ is defined as

$$s(I, E) = \frac{c(I, E)}{N(I, E)} = \frac{|I \cap E|}{\max(|I|, |E|)}$$

The terms $c(I, E)$ and $N(I, E)$ are symmetric; therefore this similarity measure is symmetric. This measure takes a high value when there is a large number of matching words and the length of the example is similar to the input. The measure takes values in the $[0, 1]$ range when no word occurs twice. The range constraint is kept by distinguishing duplicate words.

4.4.2 Retrieval

The example with the highest similarity is determined after calculating the similarity of the input for each example with the previously described method.

4.5 Response Accuracy

For performance evaluation, the response accuracy is used. This measure is the rate of correct system responses for the test data. Correctness is defined as follows: a system response of an input is correct when the system response is equal to an appropriate response of the input.

Response accuracy can also be interpreted as the user satisfaction rate.

5 QADB Optimization

5.1 Purpose

A problem specific to spoken dialog system is that response performance may decrease due to speech
recognition errors. This differs from dialog systems which use text input. Moreover, the example-based response generation method has some problems; inappropriate QA pairs or pairs with similar example questions and different responses may cause an inappropriate system response.

It is difficult to assign appropriate responses for the human annotators, who are not completely familiar with the topic domains of system responses. Therefore a QADB may include inappropriate data when constructing it from an utterance database which has been prepared by different annotators.

As solution to these problems, it is proposed to exclude unsuitable QA pairs when constructing the QADB. This method is called QADB optimization [10]. The goal of QADB optimization is to improve the response accuracy of the system. Additionally, it is expected that the QADB becomes more compact.

5.2 Principle

The main purpose of QADB optimization is the detection of useless QA pairs.

If the response accuracy of utterances improves by excluding a QA pair from a QADB, the QA pair is considered as useless.

For QADB optimization, it is important to distinguish the role of utterances from that of QA pairs, although they are of the same origin. The utterance plays the role of an evaluator for the QA pairs.

The optimization method is based on a so-called greedy algorithm to reduce the calculation amount; only one QA pair is excluded temporarily. The effect of excluding more than one pair is not considered. The method also uses leave-one-out cross-validation in order to use the whole training data for evaluation.

5.3 Algorithm

Uselessness of a QA pair w.r.t. a validation utterance is determined by using a temporary QADB constructed from the training data excluding the validation utterance. If the number of correct responses increases, the currently considered QA pair is considered as useless (increment a uselessness counter for QA pair) otherwise it is considered as useful (decrement counter for QA pair).

After this evaluation for all data, the degree of uselessness is accumulated for all QA pairs. QA pairs with a uselessness counter greater than zero are removed. This QADB optimization process is iterated several times. QA pairs removed once are not added again to the QADB later. The algorithm terminates if it is no longer possible to remove QA pairs.

5.4 Experimental Results

We evaluate the response accuracy improvement for the test utterance data and discuss the effect of excluding inappropriate system responses by QADB optimization.

5.4.1 Conditions

The data used in experiments are shown in Table 3. Only valid utterances are employed as training data. Test and training data are disjoint. Experiments with adult and child data are conducted separately. All utterances are recognized by a large vocabulary continuous speech recognizer using an N-gram language model and a context-dependent acoustic model [12]. The recognition accuracy is shown in Table 4.

5.4.2 Results

The experimental results are shown in Figure 4. The solid line shows the response accuracy and the dotted line shows the distinct number of QA pairs in the optimized QADB.

The response accuracy in case of speech input improves with optimization by 1.6% to 75.7% absolute for adults and by 2.9% to 59.8% for children. Additionally, convergence of the QADB size can be observed. The result for children data shows that the response accuracy improves with one and two iterations of the QADB optimization algorithm although speech recognition is low.

---

1 A variation of cross-validation. Split training data to one sample and the rest and evaluate the model with the remaining for the sample. Accumulate the evaluation results for all data to determine the final evaluation result.
6 Conclusions
We have employed a speech-oriented guidance system and collected spontaneous speech data for five years in order to enhance speech recognition and dialog control technology. A speech database is constructed from these utterances of two years. We described a QADB optimization method using this database; The method constructs a QADB by excluding inappropriate data effectively. The response accuracy for speech input improves with the optimization by 1.6% to 75.7% absolute for adults and by 2.9% to 59.8% for children.

7 Acknowledgement
This work is partly supported by MEXT leading project "e-society".

8 References

Table 3 Statistics of data with experiments

<table>
<thead>
<tr>
<th></th>
<th>period</th>
<th>Aug. 2003</th>
</tr>
</thead>
<tbody>
<tr>
<td>test data</td>
<td># of data</td>
<td>adult 1053</td>
</tr>
<tr>
<td></td>
<td></td>
<td>child 6543</td>
</tr>
<tr>
<td>training</td>
<td># of data</td>
<td>adult 19383</td>
</tr>
<tr>
<td>data</td>
<td></td>
<td>child 79346</td>
</tr>
<tr>
<td></td>
<td>distinct</td>
<td>adult 6345</td>
</tr>
<tr>
<td></td>
<td>QA pairs</td>
<td>child 30669</td>
</tr>
</tbody>
</table>

evaluation comparing system response with single proper response of test data

Table 4 Recognition accuracy of test data

<table>
<thead>
<tr>
<th></th>
<th>word correct rate</th>
<th>word accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>adult</td>
<td>90.33%</td>
<td>85.10%</td>
</tr>
<tr>
<td>child</td>
<td>74.76%</td>
<td>67.18%</td>
</tr>
</tbody>
</table>

Fig. 4. Response accuracy improvement by optimization.