Confirmation Strategy for Document Retrieval Systems with Spoken Dialog Interface

Teruhisa Misu, Kazunori Komatani, Tatsuya Kawahara

Graduate School of Informatics, Kyoto University
Kyoto 606-8501, Japan

Abstract

Adequate confirmation is indispensable in spoken dialog systems to eliminate misunderstandings caused by speech recognition errors. Spoken language also inherently includes redundant expressions such as disfluency and out-of-domain phrases, which do not contribute to task achievement. It is easy to define a set of keywords to be confirmed for conventional database query tasks, but not straightforward in general document retrieval tasks. In this paper, we propose two statistical measures for identifying portions to be confirmed. A *relevance score* (RS) represents matching degree with the document set. A *significance score* (SS) detects portions that consequently affect the retrieval results. With these measures, the system can generate confirmation prior to and posterior to the retrieval, respectively. The strategy is implemented and evaluated with retrieval from software support knowledge base of 40K entries. It is shown that the proposed strategy using the two measures is more efficient than using the conventional confidence measure.

1. Introduction

The target of spoken dialog systems is being extended from simple databases such as flight information [1] to general documents [2, 3] including manuals [4] and newspaper articles. It is indispensable for information retrieval systems to interpret user utterances. In conventional database query tasks, the user's intention is interpreted by extracting predefined keywords from the utterance, because the database structure and query commands are well-defined. Confirmation will be made if such keywords cannot be identified. On the other hand, in general document retrieval tasks such as queries to operation manuals or Web pages, the target of retrieval is natural language text, and it is necessary to match the whole speech recognition result as a sentence, against a set of documents. There are two problems in this case.

1. **Errors in automatic speech recognition (ASR)**

   Errors are inevitable in large vocabulary continuous speech recognition. If keywords are predefined, the system can focus on them using confidence measures [5, 6] to handle possible errors. However, it is not feasible to define such keywords in document retrieval tasks.

2. **Redundancy in spoken language expression**

   In spontaneous speech, user utterances may include redundant expressions such as disfluency and irrelevant phrases. That means every portion of the user utterance is not important for information retrieval, but might be even harmful.

To solve these problems, we need a framework to detect necessary portions for task achievement of document matching and retrieval.

In this paper, we propose an efficient confirmation strategy based on two statistical measures computed for phrase units. One is a relevance score with the target document set, which is computed with a document language model and used for making confirmation prior to the retrieval. The other is a significance score in the document matching, which is computed after the retrieval using N-best results and used for prompting the user for post-selection if necessary.

2. Text Retrieval System for Large-scale Knowledge Base

Our task involves text retrieval from a large-scale knowledge base. As the target domain, we adopt a software support knowledge base provided by Microsoft Corporation. The knowledge base consists of the following three kinds: glossary, frequently asked questions (FAQ), and a database of support articles. The specification is shown in Table 1, and there are about 40K entries in total.

Table 1: Document set (Knowledge base)

Text collection	# documents	# text size (byte)
glossary	4,707	1,400,000
FAQ	11,306	12,000,000
DB of support articles	23,323	44,000,000

Dialog Navigator[7] has been developed at University of Tokyo as a document retrieval system for this knowledge base. The system accepts a typed-text input from users and outputs a result of the retrieval. The system

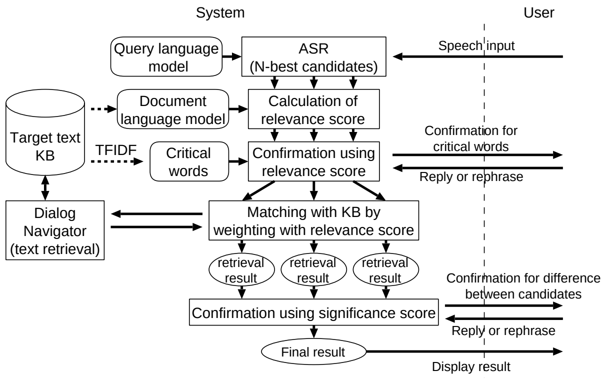
System
User
Query language model
ASR (N-best candidates)
Speech input
Target text KB
Document language model
Calculation of relevance score
TFIDF
Critical words
Confirmation using relevance score
Confirmation for critical words
Reply or rephrase
Dialog Navigator (text retrieval)
Matching with KB by weighting with relevance score
retrieval result
retrieval result
retrieval result
Confirmation for difference between candidates
Confirmation using significance score
Reply or rephrase
Final result
Display result

Figure 1: System flow

interprets an input sentence by taking a syntactic dependency and synonymous expression into consideration for matching it with the knowledge base [7].

In this work, we adopt Dialog Navigator as a back-end system and construct a spoken dialog interface. We focus on a confirmation strategy to interpret user utterances robustly, by taking into account the problems that are characteristic of spoken language as previously described.

3. Confirmation Strategy using Relevance Score and Significance Score

Making confirmation for every portion is tedious, even with a reliable confidence measure, because every erroneous portion does not necessarily affect the retrieval result. We therefore consider the influence of recognition errors for retrieval, and control generation of confirmation.

Since Dialog Navigator outputs a dozen of retrieved candidates, as in Web search engines, slight modification of the query sentence including ASR errors is tolerable as long as the major retrieved entries remain unchanged. Therefore, we make use of N-best results of ASR for the query, and test if a significant difference is caused among N-best sets of retrieved candidates. If there actually is, we then make a confirmation on the portion that makes the difference. This is regarded as a posterior confirmation. On the other hand, if there is a very critical error in the ASR result, such as those in the product name in software support, the following retrieval would make no sense. Therefore, we also introduce a confirmation prior to the retrieval for critical words.

The system flow including the confirmation is summarized below.

1. Calculate a relevance score for each phrase of the ASR result.
2. Make confirmation for critical words having a low relevance score.
3. Retrieve from the knowledge base for each of the N-best results of ASR.
4. Calculate significance scores, and generate confirmation based on them.
5. Output the retrieval results.

The flow is also shown in Figure 1, and explained in the following subsections in detail.

3.1. Definition of Relevance Score

We first define a relevance score that measures the potential degree of matching with the document set. For the purpose, we introduce a document language model, which is different from that used in ASR. Then, we measure perplexity of input portions, phrase by phrase, with this language model.

The perplexity for a portion including ASR errors usually increases because such a word sequence is contextually less frequent. The perplexity for out-of-domain phrases also tends to be large because they scarcely appear in the knowledge base. We then define a relevance score by converting the perplexity (PP) using the following function. The score ranges between 0 and 1.

$$RS = \frac{1}{1 + \exp(\alpha * (\log PP - \beta))}$$

Here, α and β are constants and empirically set to 2.0 and 11.0. An example of calculating the relevance score is shown in Figure 2. In this sample, a portion, "*Atarashiku katta* (= that I recently bought)", which appears in the beginning of the sentence does not contribute to the retrieval. A portion at the end of the sentence was incorrectly recognized because it was articulated weakly. The perplexity for these portions gets larger as a result, and the relevance score is correspondingly very small.

3.2. Confirmation for Critial Words using Relevance Score

Critical words should be confirmed before the retrieval, because the retrieval result would be severely damaged if they are not correctly recognized. We define a set of critical words using *tf·idf* values, which are derived from the target knowledge base. As a result, we selected 35 words, for example, 'set up', 'printer' and '(Microsoft) Office'.

We use the relevance score to determine whether we should make a confirmation for the critical words. If an critical word is contained in a phrase whose relevance score is lower than a threshold θ, a confirmation is made. Users can either confirm or discard, or correct the phrase, before passing it to the matching module.

User utterance:
"*Atarashiku katta XP no pasokon de fax kinou wo tsukau niha doushitara iidesu ka?*"

(Please tell me how to use the facsimile function in the personal computer with Windows XP that I recently bought.)

Speech recognition result:
"*Atarashiku katta XP no pasokon de fax kinou wo tsukau ni sono e ikou?*"

[The underlined part was incorrectly recognized.]

Division into phrases:
"*Atarashiku / katta / XP no / pasokon de / fax kinou wo / tsukau ni / sono / e / ikou?*"

Calculation of perplexity:

phrases (their context)	PP	RS
(<S>) *Atarashiku* (*katta*)	499.57	0.86
(*atarashiku*) *katta* (*XP*)	2079.83	0.47
(*katta*) *XP no* (*pasokon*)	105.64	0.99
(*no*) *pasokon de* (*FAX*)	185.92	0.95
(*de*) *FAX kinou wo* (*tsukau*)	236.23	0.89
(*wo*) *tsukau ni* (*sono*)	98.40	0.99
(*ni*) *sono* (*e*)	1378.72	0.62
(*sono*) *e* (*ikou*)	144.58	0.96
(*e*) *ikou* (</S>)	27150.00	0.00

<S>, </S> denote the beginning and end of a sentence.

Figure 2: Example of calculating perplexity (PP) and relevance score (RS)

3.3. Weighted Matching using Relevance Score

A phrase with a low relevance score is likely to be an ASR error or a portion that does not contribute to the retrieval, even if it contains content words. We therefore use the relevance score RS as a weight for phrases during the matching with the knowledge base. This is expected to reduce the damage to the retrieval by ASR errors and redundant expressions, and generate more appropriate retrieval results.

3.4. Significance Score using Retrieval Results

A significance score is defined by using plural retrieval results corresponding to the N-best candidates of ASR. Ambiguous portions during the ASR appear as the differences between the N-best candidates. The significance score represents the degree to which the portion is actually influential to the retrieval by observing the difference of the retrieval results.

The procedure to calculate a significance score requires detection of different words between the N-best candidates. By obtaining the retrieval result for each candidate, we then define a significance score SS as the difference between the retrieval results of n-th and m-th candidates as follows.

$$SS(n,m) = 1 - \frac{|res(n) \cap res(m)|^2}{|res(n)||res(m)|}$$

Here, $res(n)$ denotes a set of retrieved documents for the n-th candidate, and $|res(n)|$ denotes the number of elements in the set. That is, the significance score decreases if the two retrieval results have a large common portion.

3.5. Confirmation using Significance Score

A posterior confirmation is made based on the significance score. If the score is higher than a threshold, the system makes a confirmation by presenting the difference in the N-best list of ASR to users. Otherwise, the system just presents the retrieval result of the first candidate without making confirmation. Here, we set $N = 3$ and the threshold for the score to 0.5. In the confirmation phase, if the user selects from the list, then the system displays the corresponding retrieval result. If the user judges all candidates as inappropriate, the system rejects the current results and prompts him/her to utter the query again.

4. Experimental Evaluation

We implemented and evaluated our method as a front-end of Dialog Navigator. The ASR system consists of Julius[8] for SAPI[1] and a trigram language model trained with a query corpus as well as texts of knowledge base.

We collected the test data by 30 subjects who had not used our system. Each subject was requested to retrieve support information for 14 tasks, which consisted of 11 prepared scenarios (query sentences are not given) and 3 spontaneous queries. Subjects were allowed to utter a query sentence again up to three times per task if a relevant retrieval result was not obtained. We obtained 651 utterances for 420 tasks in total. The average word accuracy of ASR was 76.8%.

4.1. Evaluation of Success Rate of Retrieval

First, we evaluated with a success rate of retrieval for the collected speech data. We regard a retrieval as successful when the retrieval results contain a correct answer for the user's initial query. We compared following cases.

1. Transcription: A correct transcription of user utterances, which was made manually, was used as an input to Dialog Navigator.

2. ASR result: The first candidate of ASR was used as an input (baseline).

3. Proposed method: Using the relevance and significance scores, the proposed confirmation strategy was adopted.

[1] http://julius.sourceforge.jp/sapi/

Table 2: Success rates of retrieval

# utterances	Transcription	ASR result	Proposed method
651	520 (79.9%)	421 (64.7%)	457 (70.2%)

Table 3: Comparison with method using confidence measure (CM)

	Proposed method	CM ($\theta_1 = 0.4$)	CM ($\theta_1 = 0.6$)	CM ($\theta_1 = 0.8$)
# confirmation	221	77	254	484
# success (success rate)	457 (70.2%)	427 (65.6%)	435 (66.8%)	445 (68.4%)

Table 2 lists the success rates for three cases. The proposed method attained a better rate than the case where the first candidate of ASR was used. Improvement of 36 cases (5.5%) was obtained by the proposed method, including 30 by the confirmation, and 14 by weighting during the matching using a relevance score, though the retrieval failed eight times as side effects of the weighting. Thus, the proposed confirmation strategy is effective in improving the task achievement.

4.2. Evaluation of Efficiency of Confirmation

We also evaluated in terms of the number of generated confirmations. The proposed method generated 221 confirmations. This means that confirmations were generated once every three utterances on the average. The 221 confirmations consisted of 66 prior to the retrieval using the relevance score and 155 posterior to the retrieval using the significance score.

We compared the proposed method with a conventional method, which used a confidence measure based on the N-best candidates of ASR [5]. In this method, the system generated confirmation only for content words having a confidence measure lower than θ_1. The threshold to generate confirmation (θ_1) was set to 0.4, 0.6 and 0.8.

The number of confirmations and retrieval successes are shown in Table 3. The proposed method achieved a higher success rate with a less number of confirmations (less than half) compared with the case of $\theta_1 = 0.8$ in the conventional method. Thus, the proposed confirmation strategy is more efficient.

5. Conclusion

We addressed an efficient confirmation strategy for document retrieval tasks. It consists of a prior confirmation for critical words and a posterior confirmation using the N-best list. We have introduced two measures of relevance score and significant score for respective confirmation phases. An experimental evaluation in the retrieval of software support documents shows that the proposed method generates confirmation more efficiently for better task achievement compared with the method using the conventional confidence measure of ASR. The proposed method is not dependent on the software support task, and expected to be applicable to general document retrieval tasks.

Acknowledgment: The authors are grateful to Prof. Kurohashi and Mr. Kiyota at University of Tokyo and Ms. Kido at Microsoft Corporation for their helpful advice.

6. References

[1] E. Levin, S. Narayanan, R. Pieraccini, K. Biatov, E. Bocchieri, G. Di Fabbrizio, W. Eckert, S. Lee, A. Pokrovsky, M. Rahim, P. Ruscitti, and M. Walker. The AT&T-DARPA communicator mixed-initiative spoken dialogue system. In *Proc. ICSLP*, 2000.

[2] S. Harabagiu, D. Moldovan, and J. Picone. Open-domain voice-activated question answering. In *Proc. COLING*, pages 502–508, 2002.

[3] C. Hori, T. Hori, H. Isozaki, E. Maeda, S. Katagiri, and S. Furui. Deriving disambiguous queries in a spoken interactive ODQA system. In *Proc. IEEE-ICASSP*, 2003.

[4] K. Komatani, T. Kawahara, R. Ito, and H. G. Okuno. Efficient dialogue strategy to find users' intended items from information query results. In *Proc. COLING*, pages 481–487, 2002.

[5] K. Komatani and T. Kawahara. Flexible mixed-initiative dialogue management using concept-level confidence measures of speech recognizer output. In *Proc. COLING*, pages 467–473, 2000.

[6] T. J. Hazen, T. Burianek, J. Polifroni, and S. Seneff. Integrating recognition confidence scoring with language understanding and dialogue modeling. In *Proc. ICSLP*, 2000.

[7] Y. Kiyota, S. Kurohashi, and F. Kido. "Dialog Navigator": A question answering system based on large text knowledge base. In *Proc. COLING*, pages 460–466, 2002.

[8] A. Lee, T. Kawahara, and K. Shikano. Julius – an open source real-time large vocabulary recognition engine. In *Proc. EUROSPEECH*, pages 1691–1694, 2001.