Efficient Confirmation Strategy for Large-scale Text Retrieval Systems with Spoken Dialogue Interface

Kazunori Komatani Teruhisa Misu Tatsuya Kawahara Hiroshi G. Okuno
Graduate School of Informatics
Kyoto University
Yoshida-Hommachi, Sakyo, Kyoto 606-8501, Japan
{komatani,kawahara,okuno}@i.kyoto-u.ac.jp

Abstract

Adequate confirmation for keywords is indispensable in spoken dialogue systems to eliminate misunderstandings caused by speech recognition errors. Spoken language also inherently includes out-of-domain phrases and redundant expressions such as disfluency, which do not contribute to task achievement. It is necessary to appropriately make confirmation for important portions. However, a set of keywords necessary to achieve the tasks cannot be predefined in retrieval for a large-scale knowledge base unlike conventional database query tasks. In this paper, we describe two statistical measures for identifying portions to be confirmed. A *relevance score* represents the matching degree with the target knowledge base. A *significance score* detects portions that consequently affect the retrieval results. These measures are defined based on information that is automatically derived from the target knowledge base. An experimental evaluation shows that our method improved the success rate of retrieval by generating confirmation more efficiently than using a conventional confidence measure.

1 Introduction

Information retrieval systems with spoken language have been studied (Harabagiu et al., 2002; Hori et al., 2003). They require both automatic speech recognition (ASR) and information retrieval (IR) technologies. As a straight manifestation to create these systems, we can think of using ASR results as an input for IR systems that retrieve a text knowledge base (KB). However, two problems occur in the characteristics of speech.

1. Speech recognition errors
2. Redundancy included in spoken language expressions

One is an ASR error, which is basically inevitable in speech communications. Therefore, an adequate confirmation is indispensable in spoken dialogue systems to eliminate the misunderstandings caused by ASR errors.

If keywords to be confirmed are defined, the system can confirm them using confidence measures (Komatani and Kawahara, 2000; Hazen et al., 2000) to manage the errors. In conventional tasks for spoken dialogue systems in which their target of retrieval was well-defined, such as the relational database, keywords that are important to achieve the tasks correspond to items in the relational database. Most spoken dialogue systems that have been developed, such as airline information systems (Levin et al., 2000; Potamianos et al., 2000; San-Segundo et al., 2000) and train information systems (Allen et al., 1996; Sturm et al., 1999; Lamel et al., 1999), are categorized here. However, it is not feasible to define such keywords in retrieval for operation manuals (Komatani et al., 2002) or WWW pages, where the target of retrieval is not organized and is written as natural language text.

Another problem is that a user's utterances may include redundant expressions or out-of-domain phrases. A speech interface has been said to have the advantage of ease of input. This means that redundant expressions, such as disfluency and irrelevant phrases, are easily input. These do not directly contribute to task achievement and might even be harmful. ASR results that may include such redundant portions are not adequate for an input of IR systems.

A novel method is described in this paper that automatically detects necessary portions for task achievement from the ASR results of a user's utterances; that is, the system determines if each part of the ASR results is necessary for the retrieval. We introduce two measures for each portion of the results. One is a *relevance score* (RS) with the target document

[HOWTO] Use Speech Recognition in Windows XP

The information in this article applies to:

- Microsoft Windows XP Professional
- Microsoft Windows XP Home Edition

Summary: This article describes how to use speech recognition in Windows XP. If you installed speech recognition with Microsoft Office XP, or if you purchased a new computer that has Office XP installed, you can use speech recognition in all Office programs as well as other programs for which it is enabled.

Detail information: Speech recognition enables the operating system to convert spoken words to written text. An internal driver, called a speech recognition engine, recognizes words and converts them to text. The speech recognition engine ...

Figure 1: Example of one article in database

set. The score is computed with a document language model and is used for making confirmation prior to the retrieval. The other is a *significance score* (SS) in the document matching. It is computed after the retrieval using N-best results and is used for prompting the user for post-selection if necessary. Information necessary to define these two measures, such as a document language model and retrieval results for N-best candidates of the ASR, can be automatically derived from the target knowledge base. Therefore, the system can detect the portions necessary for the retrieval and make the confirmation efficiently without defining the keywords manually.

2 Text Retrieval System for Large-scale Knowledge Base

Our task involves text retrieval for a large-scale knowledge base. As the target domain, we adopted a software support knowledge base provided by the Microsoft Corporation. The knowledge base consists of the following three components: glossary, frequently asked questions (FAQ), and a database of support articles. Figure 1 is an example of the database. The knowledge base is very large-scale, as shown in Table 1.

The Dialog Navigator (Kiyota et al., 2002) was developed in the University of Tokyo as a text retrieval system for this knowledge base. The system accepts a typed-text input as questions from users and outputs a result of the retrieval. The system interprets input sentences taking a syntactic dependency and synonymous expression into consideration for matching it with the knowledge base. The system can also navigate for the user when he/she makes vague questions based on scenarios (dialog card) that were described manually beforehand. Hundreds of the dialog cards have been prepared to ask questions back to the users. If a user question matches its input part, the system generates a question based on its description.

Table 1: Document set (Knowledge base)

Text collection	# of texts	# of characters
Glossary	4,707	700,000
FAQ	11,306	6,000,000
Support articles	23,323	22,000,000

We adopted the Dialog Navigator as a backend system and constructed a text retrieval system with a spoken dialogue interface. We then investigated a confirmation strategy to interpret the user's 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 confirmations for every portion that has the possibility to be an ASR error is tedious. This is because every erroneous portion does not necessarily affect the retrieval results. We therefore take the influence of recognition errors for retrieval into consideration, and control generation of confirmation.

We make use of N-best results of the 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 a critical error occurs 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. Recognize a user's utterance.

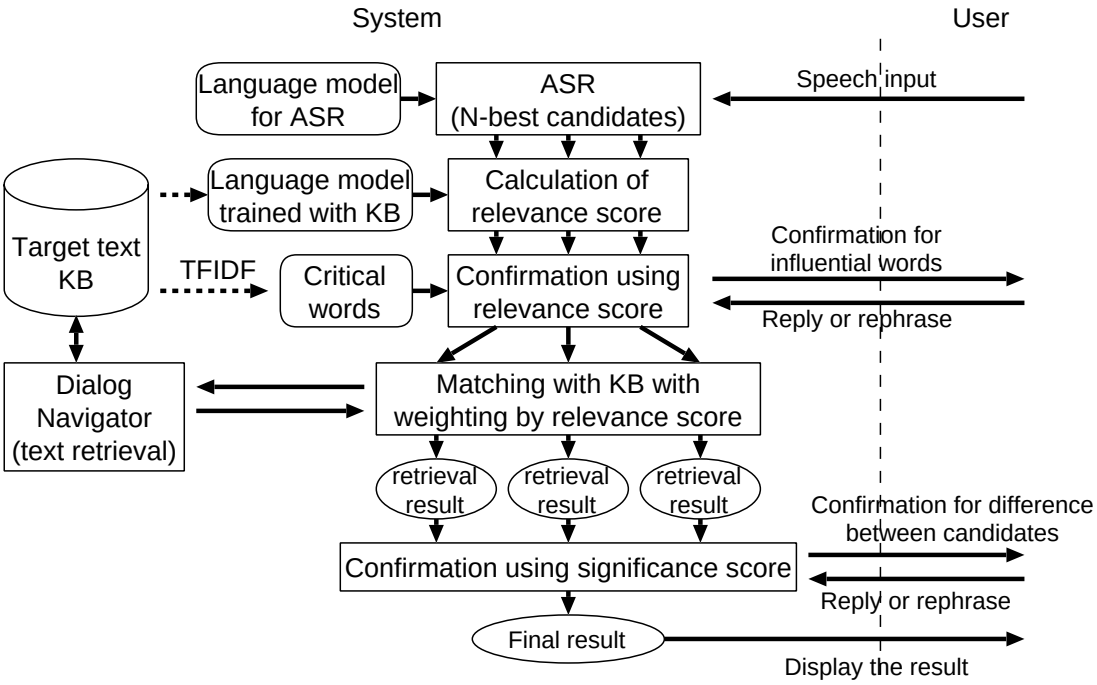

Figure 2: System flow

2. Calculate a relevance score for each phrase of ASR results.
3. Make a confirmation for critical words with a low relevance score.
4. Retrieve the knowledge base using the Dialog Navigator for N-best candidates of the ASR.
5. Calculate significance scores and generate a confirmation based on them.
6. Show the retrieval results to the user.

This flow is also shown in Figure 2 and explained in the following subsections in detail.

3.1 Definition of Relevance Score

We use test-set perplexity for each portion of the ASR results as one of the criteria in determining whether the portion is influential or not for the retrieval. The language model to calculate the perplexity was trained only with the target knowledge base. It is different from that used in the ASR.

The perplexity is defined as an exponential of entropy per word, and it represents the average number of the next words when we observe a word sequence. The perplexity can be denoted as the following equation because we assume an ergodicity on language and use a trigram as a language model.

$$\log PP = -\frac{1}{n}\sum_{k}\log P(w_k|w_{k-1}, w_{k-2})$$

This perplexity PP represents the degree that a given word sequence, $w_1 w_2 \cdots w_n$, matches the knowledge base with which the language model $P(w_n|w_{n-1}, w_{n-2})$ was trained. If the perplexity is small, it indicates the sequence appears frequently in the knowledge base. On the contrary, the perplexity for a portion including the ASR errors increases because it is contextually less frequent. The perplexity for out-of-domain phrases similarly increases because they scarcely appear in the knowledge base. It enables us to detect a portion that is not influential for retrieval or those portions that include ASR errors. Here, a phrase, called *bunsetsu*[1] in Japanese, is adopted as a portion for which the perplexity is calculated. We use a syntactic parser KNP (Kurohashi and Nagao, 1994) to divide the ASR results into the phrases[2].

[1] *Bunsetsu* is a commonly used linguistic unit in Japanese, which consists of one or more content words and zero or more functional words that follow.

[2] As the parser was designed for written language, the division often fails for portions including ASR errors. The division error, however, does not affect the whole system's performance because the perplexity for the erroneous portions increases, indicating they are irrelevant.

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 here.]

Division into phrases (*bunsetsu*):
"*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 3: Example of calculating perplexity (PP) and relevance score (RS)

We then calculate the perplexity for the phrases (*bunsetsu*) to which the preceding and following words are attached. We then define the relevance score by applying a sigmoid-like transform to the perplexity using the following equation. Thus, the score ranges between 0 and 1 by the transform and can be used as a weight for each *bunsetsu*.

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

Here, α and β are constants and are empirically set to 2.0 and 11.0. An example of calculating the relevance score is shown in Figure 3. In this sample, a portion, "*Atarashiku katta* (= that I bought recently)", that appeared in the beginning of the utterance does not contribute to any retrieval. A portion at the end of the sentence was incorrectly recognized because it may have been pronounced weakly. The perplexity for these portions increases as a result, and the relevance score correspondingly decreases.

3.2 Confirmation for Critical Words using Relevance Score

Critical words should be confirmed before the retrieval. This is because a retrieval result will be severely damaged if the portions are not correctly recognized. We define a set of words that are potentially critical using *tf·idf* values, which are often used in information retrieval. They can be derived from the target knowledge base automatically. We regard a word with the maximum *tf·idf* values in each document as being its representative, and the words that are representative in more documents are regarded as being more important. When the amount of documents represented by the more important words exceeds 10% out of the whole number of documents, we define a set of the words as being critical. As a result, 35 words were selected as potentially critical ones in the knowledge base, such as 'set up', 'printer', and '(Microsoft) Office'.

We use the relevance score to determine whether we should make a confirmation for the critical words. If a critical word is contained in a phrase whose relevance score is lower than threshold θ, the system makes a confirmation. We set threshold θ through the preliminary experiment. The confirmation is done by presenting the recognition results to the user. Users can either confirm or discard or correct the phrase before passing it to the following matching module.

3.3 Weighted Matching using Relevance Score

A phrase that has a low relevance score is likely to be an ASR error or a portion that does not contribute to retrieval. We therefore use the relevance score RS as a weight for phrases during the matching with the knowledge base. This relieves damage to the retrieval by ASR errors or redundant expressions.

3.4 Significance Score using Retrieval Results

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

The significance score is calculated for portions that are different among N-best candidates. We define the significance score $SS(n, m)$ as the difference between the retrieval results of

n-th and m-th candidates. The value is defined by the equation,

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

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

Figure 4 has an example of calculating the significance score. In this sample, the portions of "*suuzi* (numerals)" and "*suushiki* (numeral expressions)" differ between the first and second candidates of the ASR. As the retrieval results for each candidate, 14 and 15 items are obtained, respectively. The number of common items between the two retrieval results is 8. Then, the significance score for the portion is 0.70 by the above equation.

3.5 Confirmation using Significance Score

The confirmation is also made for the portions detected by the significance score. If the score is higher than a threshold, the system makes the confirmation by presenting the difference to users[3]. Here, we set the number of N-best candidates of the ASR to 3, and the threshold for the score is set to 0.5.

In the confirmation phrase, if a user selects from the list, the system displays the corresponding retrieval results. If the score is lower than the threshold, the system does not make the confirmation and presents retrieval results of the first candidate of the ASR. If a user judges all candidates as inappropriate, the system rejects the current candidates 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 front-end works on a Web browser, Internet Explorer 6.0. Julius (Lee et al., 2001) for SAPI[4] was used as a speech recognizer on PCs. The system presents a confirmation to users on the display. He or she replies to the confirmation by selecting choices with the mouse.

[3]Confirmation will be generated practically if one of the significance scores between the first candidate and others exceeds the threshold.

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

[#1 candidate of ASR]

"*WORD2002 de suuzi wo nyuryoku suru houhou wo oshiete kudasai.*" (Please tell me the way to input numerals in WORD 2002.)

Retrieval results (# of the results was 14.)

1. Input the present date and time in Word
2. WORD: Add a space between Japanese and alphanumeric characters
3. WORD: Check the form of inputted characters
4. WORD: Input a handwritten signature
5. WORD: Put watermark characters into the background of a character
6. ...

[#2 candidate of ASR]

"*WORD2002 de suushiki wo nyuryoku suru houhou wo oshiete kudasai.*" (Please tell me the way to input numerical expressions in WORD 2002.)

Retrieval results (# of the results was 15.)

1. Insert numerical expressions in Word
2. Input the present date and time in Word
3. Input numerical expressions in Spreadsheet
4. Input numerical expressions in PowerPoint
5. Input numerical expressions in Excel
6. ...

Significance score

$SS(1,2) = 1 - \frac{8^2}{14 \times 15} = 0.70$

(# of common items in the retrieval results was 8.)

Figure 4: Example of calculating significance score

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 the sentence again up to 3 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 the ASR was 76.8%.

4.1 Evaluation of Success Rate of Retrieval

We calculated the success rates of retrieval for the collected speech data. We regarded the retrieval as having succeeded when the retrieval results contained an answer for the user's initial question. We set three experimental conditions:

Table 2: Comparison of success rate of retrieval with method using only ASR results

# utterances	Transcription	ASR results	Our method
651	520 (79.9%)	421 (64.7%)	457 (70.2%)

1. Transcription: A correct transcription of user utterances, which was made manually, was used as an input to the Dialog Navigator. This condition corresponds to a case of 100% ASR accuracy, indicating an utmost performance obtained by improvements in the ASR and our dialogue strategy.
2. ASR results: The first candidate of the ASR was used as an input (baseline).
3. Our method: The N-best candidates of the ASR were used as an input, and confirmation was generated based on our method using both the relevance and significance scores. It was assumed that the users responded appropriately to the generated confirmation.

Table 2 lists the success rate. The rate when the transcription was used as the input was 79.9%. The remaining errors included those caused by irrelevant user utterances and those in the text retrieval system. Our method attained a better success rate than the condition where the first candidate of the ASR was used. Improvement of 36 cases (5.5%) was obtained by our method, including 30 by the confirmations and 14 by weighting during the matching using the relevance score, though the retrieval failed eight times as side effects of the weighting.

We further investigated the results shown in Table 2. Table 3 lists the relations between the success rate of the retrieval and the accuracy of the ASR per utterance. The improvement rate out of the number of utterances was rather high between 40% and 60%. This means that our method was effective not only for utterances with high ASR accuracy but also for those with around 50% accuracy. That is, an appropriate confirmation was generated even for utterances whose ASR accuracy was not very high.

4.2 Evaluation of Confirmation Efficiency

We also evaluated our method from the number of generated confirmations. Our 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 our method with a conventional one, which used a confidence measure (CM) based on N-best candidates of the ASR (Komatani and Kawahara, 2000)[5]. In this method, the system generated confirmation only for content words with a confidence measure lower than θ_1. The thresholds to generate confirmation (θ_1) were set to 0.4, 0.6, and 0.8. If a content word that was confirmed was rejected by the user, the retrieval was executed after removing a phrase that included it.

The number of confirmations and retrieval successes are shown in Table 4. Our 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 generated confirmations based on the two scores were more efficient.

The confidence measure used in the conventional method only reflects the acoustic and linguistic likelihood of the ASR results. Our method, however, reflects the domain knowledge because the two scores are derived by either a language model trained with the target knowledge base or by retrieval results for the N-best candidates. The domain knowledge can be introduced without any manual deliberation. The experimental results show that the scores are appropriate to determine whether a confirmation should be generated or not.

5 Conclusion

We described an appropriate confirmation strategy for large-scale text retrieval systems with a spoken dialogue interface. We introduced two measures, relevance score and significance score, for ASR results. The measures are useful to control confirmation efficiently for portions including either ASR errors or redundant expressions. The portions to be confirmed are determined

[5]We used a word-level CM only because defining semantic categories for content words is required to calculate the concept-level CM. Because the semantic category corresponded to items in a relational database, we cannot use the concept-level CM in this task.

Table 3: Success rate of retrieval per ASR accuracy

ASR accuracy (%)	# utterances	ASR results	Our method	# improvement
−40	37	9	11	2 (5.4%)
−60	73	33	42	9 (12.3%)
−80	194	116	129	13 (6.7%)
−100	347	263	275	12 (3.5%)
Total	651	421	457	36 (5.5%)

Table 4: Comparison with method using confidence measure (CM)

	Our 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%)

using information that is automatically derived from the target knowledge base, such as a statistical language model, *tf·idf* values, and retrieval results. An experimental evaluation shows that our method can efficiently generate confirmations for better task achievement compared with that using a conventional confidence measure of the ASR. Our method is not dependent on the software support task, and expected to be applicable to general text retrieval tasks.

6 Acknowledgments

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

References

J. F. Allen, B. W. Miller, E. K. Ringger, and T. Sikorski. 1996. A robust system for natural spoken dialogue. In *Proc. of the 34th Annual Meeting of the ACL*, pages 62–70.

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

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

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

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

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

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

S. Kurohashi and M. Nagao. 1994. A syntactic analysis method of long Japanese sentences based on the detection of conjunctive structures. *Computational Linguistics*, 20(4):507–534.

L. F. Lamel, S. Rosset, J-L. S. Gauvain, and S. K. Bennacef. 1999. The LIMSI ARISE system for train travel information. In *Proc. IEEE-ICASSP*.

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

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. 2000. The AT&T-DARPA communicator mixed-initiative spoken dialogue system. In *Proc. ICSLP*.

A. Potamianos, E. Ammicht, and H.-K. J. Kuo. 2000. Dialogue management in the Bell labs communicator system. In *Proc. ICSLP*.

R. San-Segundo, B. Pellom, W. Ward, and J. Pardo. 2000. Confidence measures for dialogue management in the CU communicator system. In *Proc. IEEE-ICASSP*.

J. Sturm, E. Os, and L. Boves. 1999. Issues in spoken dialogue systems: Experiences with the Dutch ARISE system. In *Proc. ESCA workshop on Interactive Dialogue in Multi-Modal Systems*.