Flexible Mixed-Initiative Dialogue Management using Concept-Level Confidence Measures of Speech Recognizer Output

Kazunori Komatani and **Tatsuya Kawahara**
Graduate School of Informatics, Kyoto University
Kyoto 606-8501, Japan
{komatani, kawahara}@kuis.kyoto-u.ac.jp

Abstract

We present a method to realize flexible mixed-initiative dialogue, in which the system can make effective confirmation and guidance using concept-level confidence measures (CMs) derived from speech recognizer output in order to handle speech recognition errors. We define two concept-level CMs, which are on content-words and on semantic-attributes, using 10-best outputs of the speech recognizer and parsing with phrase-level grammars. Content-word CM is useful for selecting plausible interpretations. Less confident interpretations are given to confirmation process. The strategy improved the interpretation accuracy by 11.5%. Moreover, the semantic-attribute CM is used to estimate user's intention and generates system-initiative guidances even when successful interpretation is not obtained.

1 Introduction

In a spoken dialogue system, it frequently occurs that the system incorrectly recognizes user utterances and the user makes expressions the system has not expected. These problems are essentially inevitable in handling the natural language by computers, even if vocabulary and grammar of the system are tuned. This lack of robustness is one of the reason why spoken dialogue systems have not been widely deployed.

In order to realize a robust spoken dialogue system, it is inevitable to handle speech recognition errors. To suppress recognition errors, system-initiative dialogue is effective. But it can be adopted only in a simple task. For instance, the form-filling task can be realized by a simple strategy where the system asks a user the slot values in a fixed order. In such a system-initiated interaction, the recognizer easily narrows down the vocabulary of the next user's utterance, thus the recognition gets easier.

On the other hand, in more complicated task such as information retrieval, the vocabulary of the next utterance cannot be limited on all occasions, because the user should be able to input the values in various orders based on his preference. Therefore, without imposing a rigid template upon the user, the system must behave appropriately even when speech recognizer output contains some errors.

Obviously, making confirmation is effective to avoid misunderstandings caused by speech recognition errors. However, when confirmations are made for every utterance, the dialogue will become too redundant and consequently troublesome for users. Previous works have shown that confirmation strategy should be decided according to the frequency of speech recognition errors, using mathematical formula (Niimi and Kobayashi, 1996) and using computer-to-computer simulation (Watanabe et al., 1998). These works assume fixed performance (averaged speech recognition accuracy) in whole dialogue with any speakers. For flexible dialogue management, however the confirmation strategy must be dynamically changed based on the individual utterances. For instance, we human make confirmation only when we are not confident. Similarly, confidence measures (CMs) of every speech recognition output should be modeled as a criterion to control dialogue management.

CMs have been calculated in previous works using transcripts and various knowledge sources (Litman et al., 1999) (Pao et al., 1998). For more flexible interaction, it is desirable that CMs are defined on each word rather than whole sentence, because the system can handle only unreliable portions of an utterance instead of accepting/rejecting whole sentence.

In this paper, we propose two concept-level CMs that are on content-word level and on semantic-attribute level for every content word. Because the CMs are defined using only speech recognizer output, they can be computed in real time. The system can make efficient confirmation and effective guidance according to the CMs. Even when successful interpretation is not obtained on content-word level, the system generates system-initiative guidances based on the semantic-attribute level, which lead the next user's utterance to successful interpretation.

2 Definition of Confidence Measures (CMs)

Confidence Measures (CMs) have been studied for utterance verification that verifies speech recognition result as a post-processing (Kawahara et al., 1998). Since an automatic speech recognition is a process finding a sentence hypothesis with the maximum likelihood for an input speech, some measures are needed in order to distinguish a correct recognition result from incorrect one. In this section, we describe definition of two level CMs which are on content-words and on semantic-attributes, using 10-best output of the speech recognizer and parsing with phrase-level grammars.

2.1 Definition of CM for Content Word

In the speech recognition process, both acoustic probability and linguistic probability of words are multiplied (summed up in log-scale) over a sentence, and the sequence having maximum likelihood is obtained by a search algorithm. A score of sentence derived from the speech recognizer is log-scaled likelihood of a hypothesis sequence. We use a grammar-based speech recognizer Julian (Lee et al., 1999), which was developed in our laboratory. It correctly obtains the N-best candidates and their scores by using A* search algorithm.

Using the scores of these N-best candidates, we calculate content-word CMs as below. The content words are extracted by parsing with phrase-level grammars that are used in speech recognition process. In this paper, we set $N = 10$ after we examined various values of N as the number of computed candidates [1].

[1]Even if we set N larger than 10, the scores of i-th hypotheses ($i > 10$) are too small to affect resulting CMs.

First, each i-th score is multiplied by a factor $\alpha(\alpha < 1)$. This factor smoothes the difference of N-best scores to get adequately distributed CMs. Because the distribution of the absolute values is different among kinds of statistical acoustic model (monophone, triphone, and so on), different values must be used. The value of α is examined in the preliminary experiment. In this paper, we set $\alpha = 0.05$ when using triphone model as acoustic model. Next, they are transformed from log-scaled value ($\alpha \cdot scaled_i$) to probability dimension by taking its exponential, and calculate a posteriori probability for each i-th candidate (Bouwman et al., 1999).

$$p_i = \frac{e^{\alpha \cdot scaled_i}}{\sum_{j=1}^{n} e^{\alpha \cdot scaled_j}}$$

This p_i represents a posteriori probability of the i-th sentence hypothesis.

Then, we compute a posteriori probability for a word. If the i-th sentence contains a word w, let $\delta_{w,i} = 1$, and 0 otherwise. A posteriori probability that a word w is contained (p_w) is derived as summation of a posteriori probabilities of sentences that contain the word.

$$p_w = \sum_{i=1}^{n} p_i \cdot \delta_{w,i}$$

We define this p_w as the content-word CM (CM_w). This CM_w is calculated for every content word. Intuitively, words that appear many times in N-best hypotheses get high CMs, and frequently substituted ones in N-best hypotheses are judged as unreliable.

In Figure 1, we show an example in CM_w calculation with recognizer outputs (i-th recognized candidates and their a posteriori probabilities) for an utterance "*Futaishisetsu ni resutoran no aru yado* (Tell me hotels with restaurant facility.)". It can be observed that a correct content word 'restaurant as facility' gets a high CM value ($CM_w = 1$). The others, which are incorrectly recognized, get low CMs, and shall be rejected.

2.2 CM for Semantic Attribute

A concept category is semantic attribute assigned to content words, and it is identified by parsing with phrase-level grammars that are used in speech recognition process and represented with Finite State Automata (FSA). Since

i	Recognition candidates	p_i
1	*aa shisetsu ni resutoran no kayacho* with restaurant facility / Kayacho(location)	.24
2	*aa shisetsu ni resutoran no katsura no* with restaurant facility / Katsura(location)	.24
3	*aa shisetsu ni resutoran no kamigamo* with restaurant facility / Kamigamo(location)	.20
4	<g> *shisetsu ni resutoran no kayacho* with restaurant facility / Kayacho(location)	.08
5	<g> *shisetsu ni resutoran no katsura* with restaurant facility / Katsura(location)	.08
6	<g> *shisetsu ni resutoran no kamigamo* with restaurant facility / Kamigamo(location)	.06
7	*aa shisetsu ni resutoran no kafe* with restaurant facility / cafe(facility)	.05
8	<g> *shisetsu ni resutoran no kafe* with restaurant facility / cafe(facility)	.02
9	<g> *setsubi wo resutoran no kayacho* with restaurant facility / Kayacho(location)	.01
10	<g> *setsubi wo resutoran no katsura no* with restaurant facility / Katsura(location)	.01

<g>: filler model

CM_w	(content word)	@	(semantic attribute)
1	restaurant	@	facility
0.33	Kayacho	@	location
0.33	Katsura	@	location
0.25	Kamigamo	@	location
0.07	cafe	@	facility

Figure 1: Example of content-word CM (CM_w)

these FSAs are classified into concept categories beforehand, we can automatically derive the concept categories of words by parsing with these grammars. In our hotel query task, there are seven concept categories such as 'location', 'facility' and so on.

For this concept category, we also define semantic-attribute CMs (CM_c) as follows. First, we calculate a posteriori probabilities of N-best sentences in the same way of computing content-word CM. If a concept category c is contained in the i-th sentence, let $\delta_{c,i} = 1$, and 0 otherwise. The probability that a concept category c is correct (p_c) is derived as below.

$$p_c = \sum_{i=1}^{n} p_i \cdot \delta_{c,i}$$

We define this p_c as semantic-attribute CM (CM_c). This CM_c estimates which category the user refers to and is used to generate effective guidances.

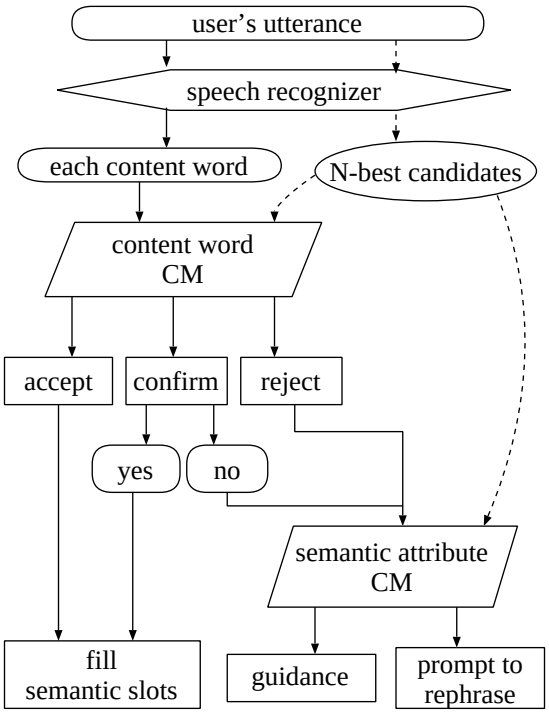


Figure 2: Overview of our strategy

3 Mixed-initiative Dialogue Strategy using CMs

There are a lot of systems that have adopted a mixed-initiative strategy (Sturm et al., 1999)(Goddeau et al., 1996)(Bennacef et al., 1996). It has several advantages. As the systems do not impose rigid system-initiated templates, the user can input values he has in mind directly, thus the dialogue becomes more natural. In conventional systems, the system-initiated utterances are considered only when semantic ambiguity occurs. But in order to realize robust interaction, the system should make confirmations to remove recognition errors and generate guidances to lead next user's utterance to successful interpretation. In this section, we describe how to generate the system-initiated utterances to deal with recognition errors. An overview of our strategy is shown in Figure 2.

3.1 Making Effective Confirmations

Confidence Measure (CM) is useful in selecting reliable candidates and controlling confirmation strategy. By setting two thresholds $\theta_1, \theta_2 (\theta_1 > \theta_2)$ on content-word CM (CM_w), we provide the confirmation strategy as follows.

- $CM_w > \theta_1$
  → accept the hypothesis
- $\theta_1 \geq CM_w > \theta_2$
  → make confirmation to the user
  "Did you say ...?"
- $\theta_2 \geq CM_w$
  → reject the hypothesis

The threshold θ_1 is used to judge whether the hypothesis is accepted or should be confirmed, and the threshold θ_2 is used to judge whether it is rejected.

Because CM_w is defined for every content word, judgment among acceptance, confirmation, or rejection is made for every content word when one utterance contains several content words. Suppose in a single utterance, one word has CM_w between θ_1 and θ_2 and the other has below θ_2, the former is given to confirmation process, and the latter is rejected. Only if all content words are rejected, the system will prompt the user to utter again. By accepting confident words and rejecting unreliable candidates, this strategy avoids redundant confirmations and focuses on necessary confirmation.

We optimize these thresholds θ_1, θ_2 considering the false acceptance (FA) and the false rejection (FR) using real data.

Moreover, the system should confirm using task-level knowledge. It is not usual that users change the already specified slot values. Thus, recognition results that overwrite filled slots are likely to be errors, even though its CM_w is high. By making confirmations in such a situation, it is expected that false acceptance (FA) is suppressed.

3.2 Generating System-Initiated Guidances

It is necessary to guide the users to recover from recognition errors. Especially for novice users, it is often effective to instruct acceptable slots of the system. It will be helpful that the system generates a guidance about the acceptable slots when the user is silent without carrying out the dialogue.

The system-initiated guidances are also effective when recognition does not go well. Even when any successful output of content words is not obtained, the system can generate effective guidances based on the semantic attribute with high confidence. An example is shown in Figure 3. In this example, all the 10-best candidates are concerning a name of place but their CM_w values are lower than the threshold (θ_2). As a result, any word will be neither accepted nor confirmed. In this case, rather than rejecting the whole sentence and telling the user "Please say again", it is better to guide the user based on the attribute having high CM_c, such as "Which city is your destination?". This guidance enables the system to narrow down the vocabulary of the next user's utterance and to reduce the recognition difficulty. It will consequently lead next user's utterance to successful interpretation.

When recognition on a content word does not

utterance: "*shozai ga oosakafu no yado*" (hotels located in Osaka pref.)
correct: Osaka-pref.@location

i	recognition candidates (<g>: filler model)
1	*shozai ga potoairando no* <g> located in Port-island
2	*shozai ga potoairando no* <g> located in Port-island
3	*shozai ga oosakafu no* <g> located in Osaka-pref.
4	*shozai ga oosakafu no* <g> located in Osaka-pref.
5	*shozai ga oosakashi no* <g> located in Osaka-city
6	*shozai ga oosakashi no* <g> located in Osaka-city
7	*shozai ga okazaki no* <g> located in Okazaki
8	*shozai ga okazaki no* <g> located in Okazaki
9	*shozai ga oohara no* <g> located in Ohara
10	*shozai ga oohara no* <g> located in Ohara

CM_c	semantic attributes
1	location

CM_w	content words
0.38	Port-island@location
0.30	Osaka-pref.@location
0.13	Osaka-city@location
0.11	Okazaki@location
0.08	Ohara@location

Figure 3: Example of high semantic attribute confidence in spite of low word confidence

go well repeatedly in spite of high semantic-attribute CM, it is reasoned that the content word may be out-of-vocabulary. In such a case, the system should change the question. For example, if an utterance contains an out-of-vocabulary word and its semantic-attribute is inferred as "location", the system can make guidance, "Please specify with the name of prefecture", which will lead the next user's utterance into the system's vocabulary.

4 Experimental Evaluation

4.1 Task and Data

We evaluate our method on the hotel query task. We collected 120 minutes speech data by 24 novice users by using the prototype system with GUI (Figure 4) (Kawahara et al., 1999). The users were given simple instruction beforehand on the system's task, retrievable items, how to cancel input values, and so on. The data is segmented into 705 utterances, with a pause of 1.25 seconds. The vocabulary of the system contains 982 words, and the number of database records is 2040.

Out of 705 utterances, 124 utterances (17.6%) are beyond the system's capability, namely they are out-of-vocabulary, out-of-grammar, out-of-task, or fragment of utterance. In following experiments, we evaluate the system performance using all data including these unacceptable utterances in order to evaluate how the system can reject unexpected utterances appropriately as well as recognize normal utterances correctly.

4.2 Thresholds to Make Confirmations

In section 3.1, we presented confirmation strategy by setting two thresholds $\theta_1, \theta_2 (\theta_1 > \theta_2)$ for content-word CM (CM_w). We optimize these threshold values using the collected data. We count errors not by the utterance but by the content-word (slot). The number of slots is 804.

The threshold θ_1 decides between acceptance and confirmation. The value of θ_1 should be determined considering both the ratio of incorrectly accepting recognition errors (False Acceptance; FA) and the ratio of slots that are not filled with correct values (Slot Error; SErr). Namely, FA and SErr are defined as the complements of precision and recall rate of the output, respectively.

$$FA = \frac{\text{\# of incorrectly accepted words}}{\text{\# of accepted words}}$$

$$SErr = 1 - \frac{\text{\# of correctly accepted words}}{\text{\# of all correct words}}$$

After experimental optimization to minimize FA+SErr, we derive a value of θ_1 as 0.9.

Similarly, the threshold θ_2 decides confirmation and rejection. The value of θ_2 should be decided considering both the ratio of incorrectly rejecting content words (False Rejection; FR) and the ratio of accepting recognition errors into the confirmation process (conditional False Acceptance; cFA).

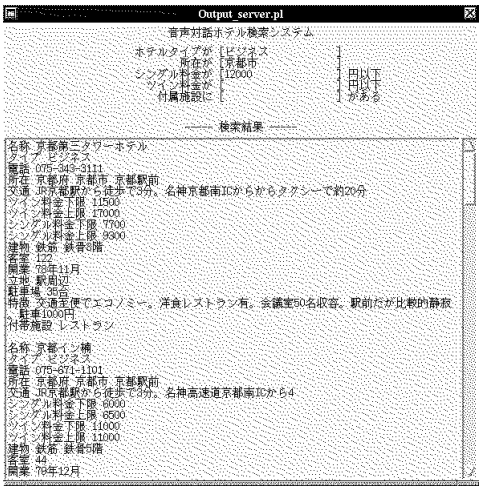
$$FR = \frac{\text{\# of incorrectly rejected words}}{\text{\# of all rejected words}}$$

If we set the threshold θ_2 lower, FR decreases and correspondingly cFA increases, which means that more candidates are obtained but more confirmations are needed. By minimizing FR+cFA, we derive a value of θ_2 as 0.6.

4.3 Comparison with Conventional Methods

In many conventional spoken dialogue systems, only 1-best candidate of a speech recognizer output is used in the subsequent processing. We compare our method with a conventional method that uses only 1-best candidate in interpretation accuracy. The result is shown in Table 1.

In the 'no confirmation' strategy, the hypotheses are classified by a single threshold (θ) into either the accepted or the rejected. Namely, content words having CM_w over threshold θ are accepted, and otherwise simply rejected. In this case, a threshold value of θ is set to 0.9 that gives minimum FA+SErr. In the 'with confirmation' strategy, the proposed confirmation strategy is adopted using θ_1 and θ_2. We set $\theta_1 = 0.9$ and $\theta_2 = 0.6$. The 'FA+SErr' in Table 1 means FA(θ_1)+SErr(θ_2), on the assumption that the confirmed phrases are correctly either accepted or rejected. We regard this assumption as appropriate, because users tend to answer 'yes' simply to express their affirmation (Hockey et al., 1997), so the system can distinguish affirmative answer and negative one by grasping simple 'yes' utterances correctly.



(a) A real system in Japanese

Hotel Accommodation Search

hotel type is [Japanese-style]

location is [downtown Kyoto]

room rate is less than [10,000] yen

. . . . . .

These are query results :

(b) Upper portion translated in English

Figure 4: An outlook of GUI (Graphical User Interface)

Table 1: Comparison of methods

	FA+SErr	FA	SErr
only 1st candidate	51.5	27.6	23.9
no confirmation	46.1	14.8	31.3
with confirmation	40.0	14.8	25.2

FA: ratio of incorrectly accepting recognition errors
SErr: ratio of slots that are not filled with correct values

As shown in Table 1, interpretation accuracy is improved by 5.4% in the 'no confirmation' strategy compared with the conventional method. And 'with confirmation' strategy, we achieve 11.5% improvement in total. This result proves that our method successfully eliminates recognition errors.

By making confirmation, the interaction becomes robust, but accordingly the number of whole utterances increases. If all candidates having CM_w under θ_1 are given to confirmation process without setting θ_2, 332 vain confirmation for incorrect contents are generated out of 400 candidates. By setting θ_2, 102 candidates having CM_w between θ_1 and θ_2 are confirmed, and the number of incorrect confirmations is suppressed to 53. Namely, the ratio of correct hypotheses and incorrect ones being confirmed are almost equal. This result shows indistinct candidates are given to confirmation process whereas scarcely confident candidates are rejected.

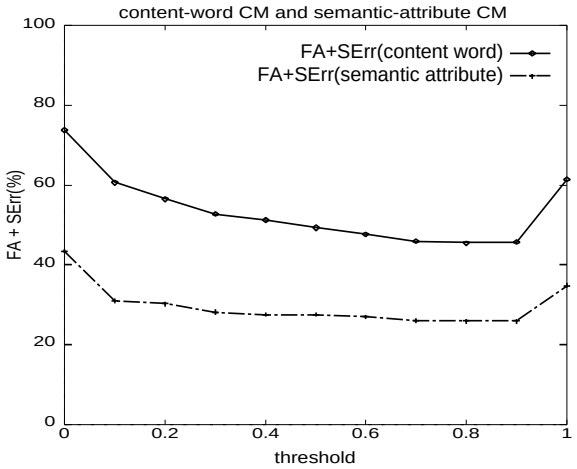


Figure 5: Performance of the two CMs

4.4 Effectiveness of Semantic-Attribute CM

In Figure 5, the relationship between content-word CM and semantic-attribute CM is shown. It is observed that semantic-attribute CMs are estimated more correctly than content-word CMs. Therefore, even when successful interpretation is not obtained from content-word CMs, semantic-attribute can be estimated correctly.

In experimental data, there are 148 slots[2] that are rejected by content-word CMs. It is also observed that 52% of semantic-attributes

[2]Out-of-vocabulary and out-of-grammar utterances are included in their phrases.

with CM_c over 0.9 is correct. Such slots amount to 34. Namely, our system can generate effective guidances against 23% (34/148) of utterances that had been only rejected in conventional methods.

5 Conclusion

We present dialogue management using two concept-level CMs in order to realize robust interaction. The content-word CM provides a criterion to decide whether an interpretation should be accepted, confirmed, or rejected. This strategy is realized by setting two thresholds that are optimized balancing false acceptance and false rejection. The interpretation error (FA+SErr) is reduced by 5.4% with no confirmation and by 11.5% with confirmations. Moreover, we define CM on semantic attributes, and propose a new method to generate effective guidances. The concept-based confidence measure realizes flexible mixed-initiative dialogue in which the system can make effective confirmation and guidance by estimating user's intention.

References

S. Bennacef, L. Devillers, S. Rosset, and L. Lamel. 1996. Dialog in the RAILTEL telephone-based system. In *Proc. Int'l Conf. on Spoken Language Processing.*

G. Bouwman, J. Sturm, and L. Boves. 1999. Incorporating confidence measures in the Dutch train timetable information system developed in the ARISE project. In *Proc. ICASSP.*

D. Goddeau, H. Meng, J. Polifroni, S. Seneff, and S. Busayapongchai. 1996. A form-based dialogue manager for spoken language applications. In *Proc. Int'l Conf. on Spoken Language Processing.*

B. A. Hockey, D. Rossen-Knill, B. Spejewski, M. Stone, and S. Isard. 1997. Can you predict responses to yes/no questions? yes,no,and stuff. In *Proc. EUROSPEECH'97.*

T. Kawahara, C.-H. Lee, and B.-H. Juang. 1998. Flexible speech understanding based on combined key-phrase detection and verification. *IEEE Trans. on Speech and Audio Processing*, 6(6):558–568.

T. Kawahara, K. Tanaka, and S. Doshita. 1999. Domain-independent platform of spoken dialogue interfaces for information query. In *Proc. ESCA workshop on Interactive Dialogue in Multi-Modal Systems*, pages 69–72.

A. Lee, T. Kawahara, and S. Doshita. 1999. Large vocabulary continuous speech recognition parser based on A* search using grammar category category-pair constraint (*in Japanese*). *Trans. Information Processing Society of Japan*, 40(4):1374–1382.

D. J. Litman, M. A. Walker, and M. S. Kearns. 1999. Automatic detection of poor speech recognition at the dialogue level. In *Proc. of 37th Annual Meeting of the ACL.*

Y. Niimi and Y. Kobayashi. 1996. A dialog control strategy based on the reliability of speech recognition. In *Proc. Int'l Conf. on Spoken Language Processing.*

C. Pao, P. Schmid, and J. Glass. 1998. Confidence scoring for speech understanding systems. In *Proc. Int'l Conf. on Spoken Language Processing.*

J. Sturm, E. Os, and L. Boves. 1999. Issues in spoken dialogue systems: Experiences with the Dutch ARISE system. In *Proc. of ESCA IDS'99 Workshop.*

T. Watanabe, M. Araki, and S. Doshita. 1998. Evaluating dialogue strategies under communication errors using computer-to-computer simulation. *Trans. of IEICE, Info & Syst.*, E81-D(9):1025–1033.