Speech Understanding Using Confidence Measures and Dependency Relations

Kazutaka Shimada, Yoshitaka Uchida*, Shizue Sato, Sayaka Minewaki, Tsutomu Endo
Department of Artificial Intelligence
Kyushu Institute of Technology
680-4 Iizuka Fukuoka 820-8502 Japan

Abstract

In this paper, we propose a method for speech understanding using a corpus. First, the method extracts keywords from an N-best list of speech recognizer output. This process employs two measures: a confidence measure of speech recognizer output and an association probability between words. Next, the method uses dependencies between keywords in a corpus. Our method obtained high accuracy as compared with a method with only the confidence measure of the speech understanding module. The results show the effectiveness of the proposed method.

Keywords : Speech Understanding, Keyword Extraction, Stochastic Association, Association Probabilities, Corpus

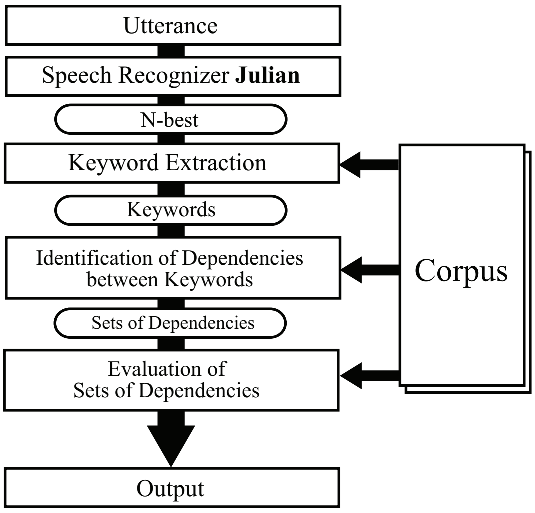

Utterance
Speech Recognizer **Julian**
N-best
Keyword Extraction
Keywords
Identification of Dependencies between Keywords
Sets of Dependencies
Evaluation of Sets of Dependencies
Output
Corpus

Figure 1: Outline of the proposed method.

1 Introduction

Speech understanding and dialogue systems have been developed to practical use recently. These systems often recognize user utterances incorrectly. It is important to deal with this problem for speech understanding systems. Funakoshi et al. [4] have reported a method to parse ill-formedness in Japanese speech. Although the accuracy for parsing the manual transcription was 90%, that for parsing the real speech recognizer output decreased to 60%. The results show the significance of handling speech recognition errors. However, it is essentially inevitable in handling the natural language by computers, even if vocabulary and grammar of the systems are tuned. The systems need to handle speech recognition errors appropriately.

As regards the problem, most of the researchers have studied a speech understanding method using keywords or key phrases [7][8][9]. Extracting keywords and understanding an utterance using them reduce speech recognition errors. Also the method allows spontaneous speech to a user. Bouwman et al. [1] and Komatani et al. [5] have reported a method for understanding an utterance using confidence measures calculated from an N-best list of a speech understanding module.

In this paper, we propose a method for speech understanding using confidence measures and dependencies. This proposed method differs from the traditional keyword based methods in that it uses dependencies between the extracted keywords for speech understanding. Also we employ a measure of semantic informativeness between words for the keyword extraction. This measure is easily calculated from a corpus.

Figure 1 shows the process flow of the proposed method. First, the method extracts keywords from an N-best list of speech recognizer output. In this process, we employ two measures for keyword extraction: confidence of speech recognizer output and association probabilities between words. Next, we use dependencies between keywords for speech understanding. We utilize a corpus for the association probabilities and dependencies.

In Sect. 2, we explain two measures for keyword extraction and in Sect. 3, we describe a method for identification of dependencies between the extracted keywords using a corpus. In Sect. 4, we evaluate the performance of our method, and con-

[*] He is now with Justsystem Corp.

clude the paper in Sect. 5.

2 Keyword extraction

Keyword extraction consists of two processes: (1) calculation of a confidence measure from speech recognizer output and (2) calculation of association probabilities between words using a corpus. The association probabilities are meanings of a word by stochastic association.

2.1 Confidence measure for content words

We use a grammar-based speech recognizer Julian. It outputs the N-best candidates and their scores. We set N=10. A score of each sentence output by the recognizer is a log-scaled likelihood.

The confidence measure from speech recognizer output in this paper is an expanded model of the method proposed in [5]. In [5], Komatani et al. calculated confidence measures of content words from the scores of speech recognizer output. The process is as follows:

1. Each i-th score ($1 \leq i \leq N$) is multiplied by a factor α ($\alpha < 1$). Then, they are transformed from log-scaled value to probability dimension by taking its exponential, and calculate a posteriori probability for each i-th candidate.

$$p_i = \frac{e^{\alpha \cdot score_i}}{\sum_{k=1}^{N} e^{\alpha \cdot score_k}} \qquad (1)$$

2. A posteriori probability for a word is calculated as follows:

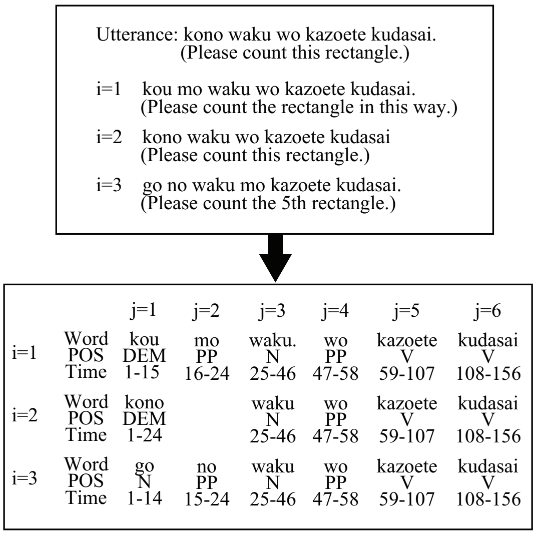
$$p_w = \sum_{i=1}^{N} p_i \cdot \delta_{w,i} \qquad (2)$$

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

They defined p_w of a content word as a confidence measure.

Although they calculate p_w for each i-th sentence only, we expand that into each morpheme. First, we align morphemes in an N-best list. We employ time and part of speech (POS) tags of morphemes for the alignment process. Figure 2 shows an example of the alignment. Next, we calculate confidence of a content word[1]:

$$P_c(w_j) = \sum_{i=1}^{N} p_i \cdot \delta_{w_j,i} \qquad (3)$$



Utterance: kono waku wo kazoete kudasai.
(Please count this rectangle.)

i=1 kou mo waku wo kazoete kudasai.
(Please count the rectangle in this way.)

i=2 kono waku wo kazoete kudasai
(Please count this rectangle.)

i=3 go no waku mo kazoete kudasai.
(Please count the 5th rectangle.)

		j=1	j=2	j=3	j=4	j=5	j=6
i=1	Word	kou	mo	waku.	wo	kazoete	kudasai
	POS	DEM	PP	N	PP	V	V
	Time	1-15	16-24	25-46	47-58	59-107	108-156
i=2	Word	kono		waku	wo	kazoete	kudasai
	POS	DEM		N	PP	V	V
	Time	1-24		25-46	47-58	59-107	108-156
i=3	Word	go	no	waku	wo	kazoete	kudasai
	POS	N	PP	N	PP	V	V
	Time	1-14	15-24	25-46	47-58	59-107	108-156

DEM: demonstrative, N: Noun, V: Verb,
PP: Postpositional Particle

Figure 2: An example of alignment.

where j denotes the position of a morpheme. If the j-th morpheme in the i-th sentence is a word w_j, let $\delta_{w_j,i} = 1$, and 0 otherwise. If the confidence is low, the word is rejected. We handle the words with high confidence as keywords.

2.2 Association probabilities between words

$P_c(w)$ in the previous section is a measure of confidence about speech recognizer outputs. Here we also calculate confidence of a word in a domain for keyword extraction. We employ association probabilities, which were proposed by Mochihashi et al. [6], between words for the confidence. The association probabilities are meanings of a word by stochastic association and are calculated from a corpus. First, they defined a measure of semantic informativeness of a word by its co-occurrence distribution. Mochihashi et al. called it association information. The association probabilities are calculated from a combination of the association information and co-occurrence probabilities.

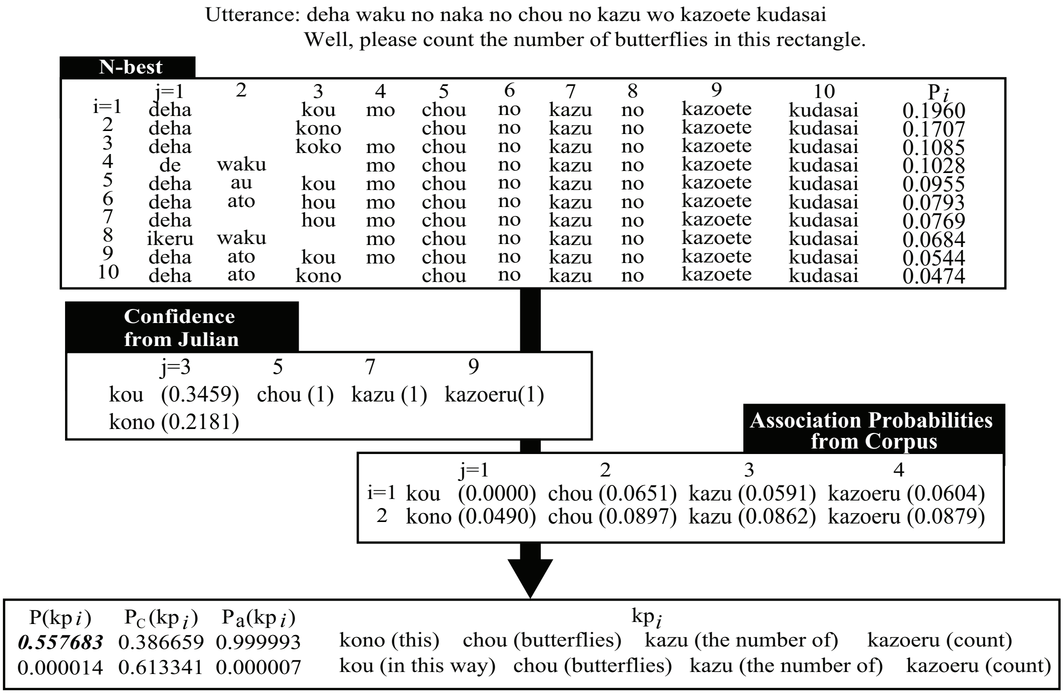
The association information of a word x is calculated as follows:

$$ap(x) = \exp(\sum_{w \in L} p(w|x) \log p(w|x)) \qquad (4)$$

where w and L are a word and words in a corpus respectively. $p(w|x)$ denotes a co-occurrence probability of w and x. $ap(x)$ is a mean co-occurrence probability of the word x. It is low if a word co-occurs with various words on the average[2].

[1] Content words in this paper are verbs, nouns, demonstratives, adjectives, attributes and adverbs.

[2] For example, postpositional particles.



Utterance: deha waku no naka no chou no kazu wo kazoete kudasai
Well, please count the number of butterflies in this rectangle.

N-best

	j=1	2	3	4	5	6	7	8	9	10	P$_i$
i=1	deha		kou	mo	chou	no	kazu	no	kazoete	kudasai	0.1960
2	deha		kono		chou	no	kazu	no	kazoete	kudasai	0.1707
3	deha		koko	mo	chou	no	kazu	no	kazoete	kudasai	0.1085
4	de	waku		mo	chou	no	kazu	no	kazoete	kudasai	0.1028
5	deha	au	kou	mo	chou	no	kazu	no	kazoete	kudasai	0.0955
6	deha	ato	hou	mo	chou	no	kazu	no	kazoete	kudasai	0.0793
7	deha		hou	mo	chou	no	kazu	no	kazoete	kudasai	0.0769
8	ikeru	waku		mo	chou	no	kazu	no	kazoete	kudasai	0.0684
9	deha	ato	kou	mo	chou	no	kazu	no	kazoete	kudasai	0.0544
10	deha	ato	kono		chou	no	kazu	no	kazoete	kudasai	0.0474

Confidence from Julian

j=3	5	7	9
kou (0.3459)	chou (1)	kazu (1)	kazoeru(1)
kono (0.2181)			

Association Probabilities from Corpus

	j=1	2	3	4
i=1	kou (0.0000)	chou (0.0651)	kazu (0.0591)	kazoeru (0.0604)
2	kono (0.0490)	chou (0.0897)	kazu (0.0862)	kazoeru (0.0879)

P(kp$i$)	P$_C$(kp$_i$)	P$_a$(kp$_i$)	kp$_i$			
0.557683	0.386659	0.999993	kono (this)	chou (butterflies)	kazu (the number of)	kazoeru (count)
0.000014	0.613341	0.000007	kou (in this way)	chou (butterflies)	kazu (the number of)	kazoeru (count)

Figure 3: An example of keyword extraction.

The association probability $a(w_j|w_i)$ is a co-occurrence probability weighted with the association information of w_j.

$$a(w_j|w_i) = \frac{p(w_j|w_i)ap(w_j)}{\sum_{w_j} p(w_j|w_i)ap(w_j)} \tag{5}$$

where $a(w_j|w_i)$ denotes the association of w_i to w_j.

2.3 Keyword extraction using two measures

We calculate a score using the confidence measure (CM_{CW}) in Sect. 2. 1 and the association probabilities in Sect. 2. 2. Figure 3 shows an example of the keyword extraction process.

First, keywords are extracted by using CM_{CW}. We combine the extracted keywords. We call the combined keywords a keyword list kp.

$$KP = \{kp_1, kp_2, kp_3, \cdots, kp_n\}$$

where $kp_i = \{k_1, k_2, \cdots, k_j, \cdots, k_m\}$ and k_j is a keyword. m is the number of keywords in a keyword list. n is the number of keyword lists. In Figure 3, $m = 4$ and $n = 2$.

Second, the association probability $P_a(k_j)$ for k_j in kp_i is calculated as follows:

$$P_a(k_j) = \sum_{\substack{l=1 \\ l \neq j}}^{m} a(k_l|k_j) \tag{6}$$

Then, the confidence measure $P_c(kp_i)$ and the association probability $P_a(kp_i)$ for a keyword list kp_i are calculated as follows:

$$P_c(kp_i) = \prod_{j=1}^{m} P_c(k_j), \tag{7}$$

$$P_a(kp_i) = \prod_{j=1}^{m} P_a(k_j) \tag{8}$$

Next, we normalize $P_c(kp_i)$ and $P_a(kp_i)$.

$$NP_c(kp_i) = \frac{P_c(kp_i)}{\sum_{k=1}^{n} P_c(kp_k)} \tag{9}$$

$$NP_a(kp_i) = \frac{P_a(kp_i)}{\sum_{k=1}^{n} P_a(kp_k)} \tag{10}$$

Finally, we calculate the harmonic mean of them.

$$P(kp_i) = \frac{2}{\frac{1}{NP_c(kp_i)} + \frac{1}{NP_a(kp_i)}} \tag{11}$$

where $P(kp_i)$ denotes the score of a keyword list kp_i. We calculate this score for KP and return them in descending order of the scores.

3 Dependencies between keywords

We proposed a method for keyword extraction in the previous section. A speech understanding method with keyword extraction, however, is insufficient because a keyword possesses relations to other ones. In this section, we describe a method

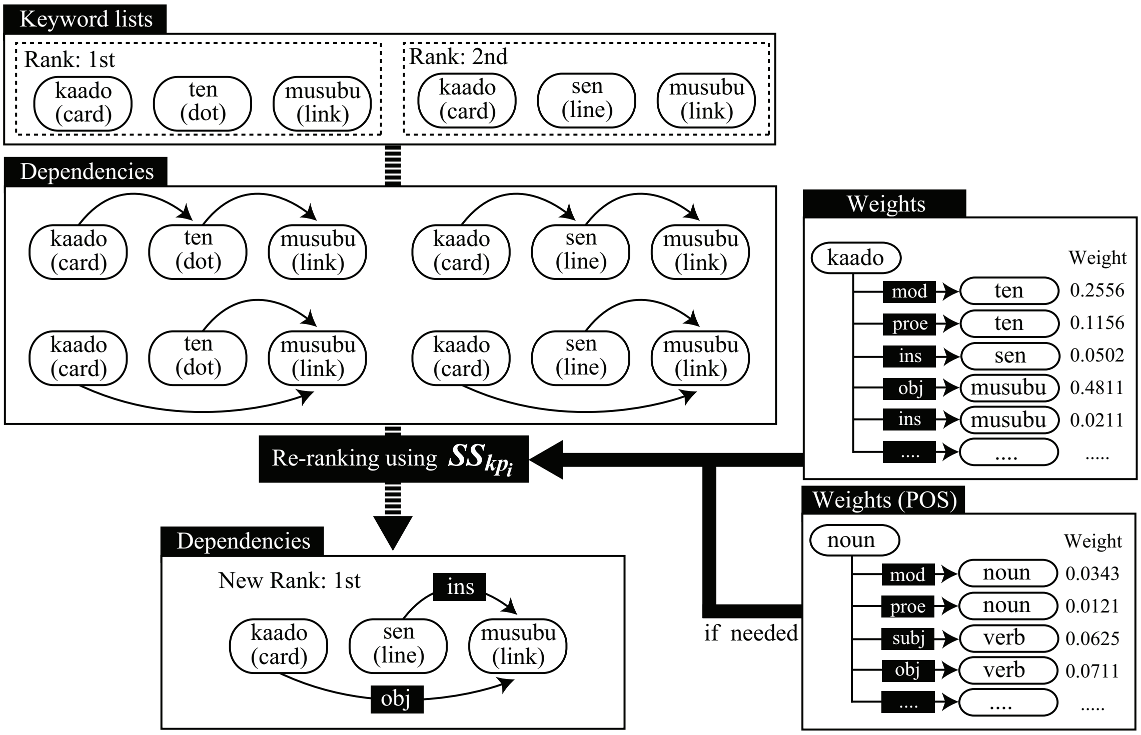


Figure 4: An example of the scoring process

for extracting a correct keyword list using dependencies between keywords.

Dependencies which are used in the process are based on the following characteristics of Japanese dependencies:

1. dependencies are directed from left to right.
2. dependencies do not cross.
3. all words except the rightmost one depend on only one other word.

We generate combined keyword lists from the extracted keywords, on the basis of these 3 characteristics.

Next, we calculate a score for the keyword lists. Weights of dependencies are computed from a corpus.

$$w_{x,y_i} = \frac{freq(x, y_i, label)}{\sum_{j=1}^{m} freq(x, y_j, label)} \tag{12}$$

The score $w_{x,y}$ is a probability that a word x depends on a word y with a relation $label$. m is the number of words which a word x depends on with a relation $label$. The $freq$ in Eq. (12) denotes the frequency of x with y and $label$. Table 1 shows examples of dependency relations.

Table 1: Case labels

Label	Role
sbj	subject
obj	object
loc	location
locf	location-from
ins	instrument

The score for a keyword list is computed as follows:

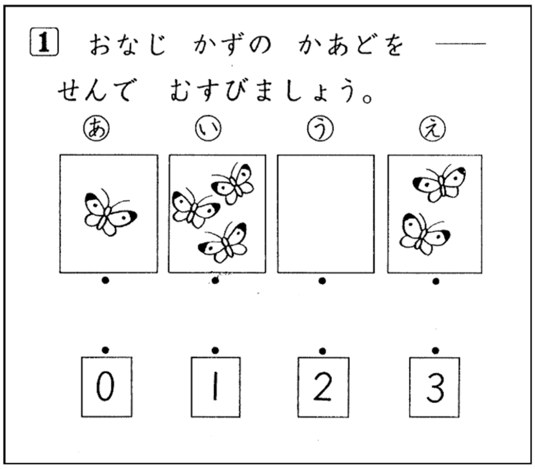
$$SS_{kp_i} = P(kp_i) + \beta \cdot \prod_{j=1}^{m-1} w_{k_j,k_l} \cdot dist_j \tag{13}$$

where $P(kp_i)$ is computed with Eq. (11) in Sect. 2.3. β is used for normalization of the number of keywords in keyword lists. $dist_j$ is the weight for the distance between k_j and k_l.

w_{x,y_i} is the weight between words that appear in a corpus. However, all dependencies do not exist in the corpus. The score of dependencies is 0 if a dependency between words in a keyword list does not exist in a corpus. We employ dependencies between parts of speech (POS) in such a case. The weight of POS is computed as follows.

$$pw_{pos_i,pos_j} = \frac{freq(pos_i, pos_j, label)}{\sum_{k=1}^{n} freq(pos_i, pos_k, label)} \times \gamma \tag{14}$$

where pos_i, pos_j and pos_k are POS tags. n is the number of POS tags in a corpus. γ is a parameter for decreasing the value of POS weights. We define γ as the average of minimum and maximum values of the weights between words. We use the weight of



```
S1: nanto onazi kazudesuka.
    (What is the number equal to?)
T1: kono waku no naka no chou wo kazoetekudasai.
    (Count the number of butterflies in this frame)
S2: kore wo kazoerunodesune.
    (May I count them?)
T2: soudesu.
    (yes)
S3: dokokara kaado wo erabunodesuka.
    (Where may I select from?)
T3: kokokara desu.
    (From here.)
S4: kono chou to kono kaado wo musubunodesune.
    (May I link card(1) and butterfly(1)?)
    .....
    ..... to be continued
```

Figure 5: An example of a drill text and a dialogue

POS as a substitute for the weight between words in this scoring process only if the weight between words does not exist in a corpus.

Figure 4 shows an example of the scoring process with dependencies.

4 Experiment

In this section, we evaluate the proposed method.

4.1 Task and corpus

We are developing a problem solving and knowledge acquisition system based on co-reference between drill texts and dialogue with a teacher, focusing on first-grade mathematics [2]. In this paper, we handled this task as a domain.

We collected a corpus consisting of simulated dialogues for this domain. The number of test subjects is 24. We obtained 2211 utterances (72 dialogues). The 72 dialogues consist of 27 dialogues with 9 test subjects for person-to-person and 45 dialogues with 15 test subjects for person-to-machine. The simulated dialogues for person-to-machine were collected by Wizard of OZ method [3]. Figure 5 shows an example of a drill text and a dialogue.

We transcribed the dialogues manually, and then tagged fillers and ill-formed expressions, such as self-correction and hesitation, in them. Words in the corpus have dependencies.

4.2 Experimental result

We used Julian, which is a grammar-based speech recognizer. It contained 107 rules as the grammars, and the number of vocabularies was 511 words. Here we set $N = 10$ and $\alpha = 0.05$. For constructing the model of association probabilities, we employ 72 dialogues consisting of 2211 utterances in our corpus and 51 dialogues consisting of 1084 utterances in the PASD corpus[3]. We evaluated 300 utterances (6 test subjects × 50 utterances). The average of keywords in an utterance is 4.42 words (1326 words / 300 utterances). We evaluated this process with the following measures: accuracy of the 1st keyword list (Acc 1st), recall and precision rates for the 1st keyword list. The recall and precision rates are computed as follows:

$$Recall = \frac{\#\ of\ keywords\ extracted\ correctly}{\#\ of\ keywords} \tag{15}$$

$$Precision = \frac{\#\ of\ keywords\ extracted\ correctly}{\#\ of\ extracted\ keywords} \tag{16}$$

Table 2 shows the results for keyword extraction. We compared our methods with Julian and only the confidence measure (CM_{CW}) described in Sect. 2.1.

The proposed methods outperformed output of Julian and CM_{CW}, with respect to all measures for the evaluation. For the 1st keyword set, the accuracy increased by 16% as compared with the CM_{CW} . The results show the effectiveness of our methods that used two measures and dependencies. The reason why the accuracy was low is that the accuracy of the speech recognizer was low. The accuracy of the 1st sentence by Julian was 22.33%. In short, the speech recognizer output did not often contain keywords that we want. In future work we can supplement keywords not containing in keyword lists by using the association probabilities because they denote the likelihood of co-occurrence between words.

Next we discussed the accuracy of identification of dependencies. For the keyword list extraction the results of the method with POS weights and without POS weights were not significantly different (see Table 2). We compared the accuracy of the methods. We evaluated them with the following points.

Eval (1)
Accuracy of identification of a dependency between two words.

Eval (2)
Accuracy of identification of dependencies of

[3] http://winnie.kuis.kyoto-u.ac.jp/taiwa-corpus/doc/

Table 2: The results of keyword extraction

	Julian	CM_{CW}	Two measures	with Dependency without POS	with Dependency with POS
Acc 1st	43.67	34.33	46.00	49.48	50.87
Precision	80.64	78.21	81.86	84.34	84.45
Recall	85.14	82.58	86.43	88.39	88.24

Table 3: The results of identification of dependencies

	without POS	with POS
Eval(1)	83.97	90.37
Eval(2)	36.21	47.64
Eval(3)	64.25	79.13

all words in the extracted keyword lists.

Eval (3)
Accuracy of identification of dependencies of all words in the keyword lists which were extracted correctly.

Table 3 shows the results of the identification process. The accuracy of the method with POS weights was higher than that without POS weights. These results show the significance of POS weights for dependency identification. To accomplish higher accuracy in the keyword extraction process, we need to examine some parameters, such as $dist_j$ in Eq. (13) and γ in Eq. (14).

5 Conclusions

In this paper, we reported a method for speech understanding. The method was based on the following processes: (1) keyword extraction using a corpus and (2) identification of dependencies between keywords for speech understanding. This method dealt with speech recognition errors appropriately. The keyword extraction process employed two measures: confidence of results of the speech understanding module, and stochastic associations between words. The accuracy of the 1st keywords list that was produced by the method consisting of two measures and dependencies increased by 16%, as compared with the method with one confidence measure. The experimental results showed the effectiveness of the proposed method.

Our future work includes (1) a large-scale experiment for this task and evaluation for other domains with the proposed method, and (2) applying the context and utterance intentions in dialogue to the method.

References

[1] C. Bouwman, J. Sturm and L. Boves, “Incorporating Confidence Measures in the Dutch Train Timetable Information System Developed in the ARICE project,” Proc. ICASSP, 1999.

[2] T. Endo and T. Kagawa, “Cooperative Understanding of Utterances and Gestures in a Dialogue-based Problem Solving System,” Computational Intelligence, Vol.5, No.2, pp.152–169, 1999.

[3] N. Fraser and G. Gilbert, “Simulating Speech Systems,” Computer Speech and Language, Vol.5, No.1, pp.81-99, 1991.

[4] K. Funakoshi, T. Tokunaga and H. Tanaka, “Evaluation of a robust parser for spoken Japanese,” Proceedings of Disfluency in Spontaneous Speech Workshop, pp. 53–56, 2003.

[5] K. Komatani and T. Kawahara, “Flexible mixed-initiative dialogue management using concept-level confidence measures of speech recognizer output,” Proc. COLING 2000, Vol. 1, pp. 467-473, 2000.

[6] D. Mochihashi and Y. Matsumoto, “Meanings as association,” IPSJ SIGNL-134 pp. 155–162, 1999.

[7] M. Nakano, N. Miyazaki and J. Hirasawa, “Understanding Unsegmented User Utterances in Real-Time Spoken Dialogue Systems,” Proc. 37th Annual Meeting of the Association for Computational Linguistics (ACL), pp.200–207, 1999.

[8] Y. Takebayashi, “Spontaneous speech dialogue system TOSBURG II - towards the user-centered multimodal interface -,” IEICE Transaction on Information and Systems, Vol.J77-D-II No.8 pp.1417-1428, 1994.

[9] T. Yano, M. Sasajima and Y. Kono, “BTH:: an efficient parsing algorithm for keyword spotting,” Journals of JSAI, Vol.17, No.6, pp.658–666, 2002.