COMBINING KEY-PHRASE DETECTION AND SUBWORD-BASED VERIFICATION FOR FLEXIBLE SPEECH UNDERSTANDING

Tatsuya Kawahara *

Department of Information Science
Kyoto University, Kyoto 606-01, Japan

Chin-Hui Lee *Biing-Hwang Juang*

Bell Laboratories, Lucent Technologies
Murray Hill, NJ 07974-0636, USA

ABSTRACT

A flexible speech understanding framework combining key-phrase detection and verification is presented. Detection of semantically-tagged key-phrases directly leads to robust understanding. In order to select reliable detection and eliminate false alarms, utterance verification technique is incorporated. A phrase verifier combines subword-based likelihood ratios of correct models and *anti-subword* alternate models. A confidence measure that focuses on mis-matched subwords is proposed and demonstrated as the most effective. The combined strategy drastically improves the semantic accuracy for out-of-grammar utterances, while maintaining the performance for in-grammar samples. We also found that utterance verification applied after grammar-based decoding is not so effective as the proposed detection and verification strategy.

1. INTRODUCTION

In the past years, several spoken dialogue systems have been evaluated in real-world applications. Mostly, these systems use finite state grammars to accept typical user utterances, because there are no data available to train statistical language models for specific tasks. However, it is observed that even after tuning the grammars by system developers, about 20~30% of user utterances are still out of the grammar specifications and they result in improper recognition. Most of these utterances contain meaningful key-phrases which could be detected and thus lead to understanding. Others are not relevant to the task and should be rejected.

When we review most of the spoken dialogue systems, their task specifications are highly well-defined, so that necessary information for the system is described with a definite set of task-related slots. Their examples include form filling or information retrieval by voice. Therefore, speech understanding problem can be formulated as extracting or detecting the task-related slots from unconstrained utterances. These slots are usually defined with keywords or key-phrases such as time and place. Thus, we have studied detection-based strategy[1] that focuses on and identifies the semantically significant portions and reject the *out-of-grammar* and *out-of-task* portions of the input utterance. Utterance verification technique enhances this property by giving confidence measures to recognition results. Combined with a flexible dialogue manager, the detection and verification framework will realize partial understanding and disambiguation of unclear portions through the subsequent dialogue session. In this paper, we mainly address how to combine verification process into detection-based recognition and evaluate our strategy by comparing with several conventional approaches.

*Work done while visiting Bell Laboratories during 1995-96

2. COMBINED DETECTION AND VERIFICATION STRATEGY

We adopt concept-based key-phrases as the detection unit[2]. They are defined so as to correspond to semantic slots such as time and place. Unlike bottom-up phrases defined by the n-gram scheme, our top-down phrases are directly mapped into semantic representations. Thus, detection of them directly leads to robust understanding. A key-phrase consists of one or a few keywords and functional words. For example, '*in the morning*' for a time period, and '*in downtown Chicago*' for a local area. In most situations, they are uttered without a break even in spontaneous speech. Thus, this longer speech unit realizes more stable matching than simple word spotting. A recurrent automata of phrase sub-grammars (*phrase network*) is used for detection.

The other main feature of the strategy is to incorporate utterance verification technique to realize ideal detection mechanism that does not match irrelevant portions of speech without using large-vocabulary non-keyword knowledge. The verification technique is used to select reliable detection and eliminate improper matching or false alarms. Based on the confidence measures, the system can reject portions that contain superfluous events such as out-of-vocabulary words and any form of disfluency.

The keyword or key-phrase verification is different from the conventional utterance verification, because it is not the final decision. False rejection of correct hypotheses is critical, while accepted false alarms can still be eliminated in the subsequent sentence parsing and verification process. Furthermore, since verification of phrases is done with partial input of fewer subword segments than the whole utterance verification, it demands more reliable confidence measures.

Finally in order to understand the whole utterance, we perform sentence-level processing that combines detected key-phrases and verifies the end result.

Thus, our overall strategy consists of the following steps, as depicted in Figure 1.

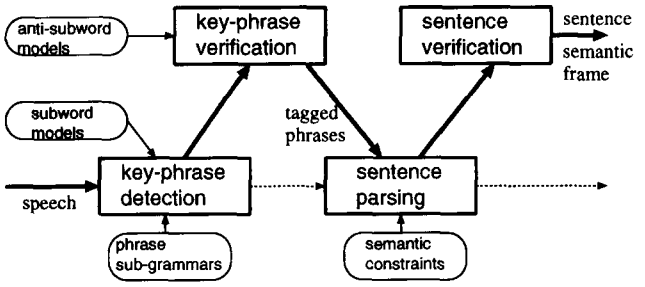

Figure 1. Outline of the strategy

1. key-phrase detection

   A set of key-phrases are detected using a set of phrase sub-grammars specific to the system prompt in the dialogue. The key-phrases are labeled with semantic tags, which are useful in sentence-level parsing. The detection algorithm is a modification of forward-backward search with hypothesis merging to generate a phrase lattice efficiently[1].

2. key-phrase verification

   The detected key-phrases are verified and assigned confidence measures. The process attempts to eliminate false alarms. It is a combination of subword-level verifications that use *anti-subword models* to test the individual subwords of the recognized results.

3. sentence parsing

   The verified key-phrase candidates are connected into sentence hypotheses using task-specific semantic knowledge. A stack decoder is used to search for the optimal hypotheses that satisfy the semantic constraints[2].

4. sentence verification

   The best sentence hypotheses are verified both acoustically and semantically for the final output.

The framework will realize not only flexible understanding but also portable and general one. For vocabulary independent recognition, universal context dependent subword units are selected and trained without influence of a specific vocabulary set[3]. The verification is also formulated in a subword-based manner. Both phrase verification and sentence verification are carried out by combining likelihood ratio scores of constituting subwords[4]. Moreover, specifying phrase sub-grammars from semantic slots is much easier than writing a whole sentence grammar.

3. SUBWORD-LEVEL VERIFICATION

For every subword n in a phrase sequence, a verification score is computed based on its corresponding ***likelihood ratio*** (LR) statistic, defined as,

$$LR_n = \frac{P(O|H_0)}{P(O|H_1)} = \frac{P(O|\lambda_n^c)}{P(O|\lambda_n^a)} \qquad (1)$$

where O is the observed speech segment, H_0 is the *null hypothesis* that subword unit n is present in the segment O, H_1 is the *alternative hypothesis* that subword n is not in the segment O, and λ_n^c and λ_n^a are the corresponding subword and *anti-subword* models for subword n, respectively[5]. The observation sequence O is aligned for subword n with the Viterbi algorithm as the result of recognition.

The anti-subword model characterizes the alternative hypothesis H_1. For every subword model, a corresponding anti-subword model is trained specifically for the verification task by clustering the highly confusing subword classes [4]. The use of an anti-subword model as a reference is more discriminative than unconstrained decoding of subword models[4], because the anti-subword model is more sensitive to the similarity of subwords and free from the performance of subword-level recognition. The anti-subword model has the same structure, i.e. number of states and mixtures, as the correct subword HMM, except that we use a context-independent model for verification.

By taking the logarithm of Eq. 1 and normalizing it by the duration l_n of the speech segment O, we define LLR_n.

$$LLR_n = \{\log P(O|\lambda_n^c) - \log P(O|\lambda_n^a)\}/l_n \qquad (2)$$

Since the first term of the equation is exactly the recognition score, we just offset the score by that computed with the anti-subword model.

4. CONFIDENCE MEASURES OF KEY-PHRASE

A confidence measure for phrase verification combines the subword-level verification scores. It is a *joint statistic* and a function of likelihood ratios of all constituting subwords.

We have investigated several functional forms of the confidence measure. The first confidence measure CM_1 is based on frame duration normalization. It is exactly the difference of the two Viterbi scores of the subword models and the corresponding anti-subword models defined as,

$$CM_1 = \frac{1}{L}\sum_n (l_n * LLR_n) \qquad (3)$$

where l_n is duration of subword n and L is total duration of the phrase, i.e. $L = \sum l_n$.

The second one CM_2 is based on subword segment-based normalization. It is a simple average of log likelihood ratios of all the subwords defined as,

$$CM_2 = \frac{1}{N}\sum_n LLR_n \qquad (4)$$

where N is the total number of subwords in the phrase.

The third one CM_3 focuses on less confident subwords rather than averaging all the subwords. This is because some subwords of an incorrect phrase may exactly match the input. In order to find less confident subwords, we normalize the log likelihood ratio assuming a Gaussian distribution for every subword. The means and variances are estimated with the samples used for training subword and anti-subword models. We denote this normalized log likelihood as LLR_n^*. Then, we pick up those subwords whose likelihood ratios are less than their means. Thus, CM_3 is defined as,

$$CM_3 = \frac{1}{N}\sum_n \begin{cases} LLR_n^* & \text{if } LLR_n^* < 0 \\ 0 & \text{otherwise} \end{cases} \qquad (5)$$

For every confidence measure, a specific threshold is set up. If its value is below the threshold, the candidate is discarded from the phrase lattice.

5. SENTENCE VERIFICATION

The sentence verification module makes the final decision on the recognition output. It uses the global acoustic and semantic information on the entire input utterance. While the key-phrase verification makes only local decisions, the sentence verification process combines its results and realizes similar effect as the conventional utterance verification algorithm, although it attempts to accept the input even if it contains unexpected extraneous words.

The semantic verification process judges if the semantic representation in the output is completed. In dialogue applications, we often observe incomplete utterances; for example, saying a month "August" without specifying any days of the month. Ideally, they should be accepted with the assumption that remaining semantic slots will be completed in the subsequent dialogue exchanges. However, unconditional approval of partial sentences invalidates the effect of utterance verification and accepts false alarms as well. Therefore, we reject a sentence hypothesis only if its semantic representation is not completed *and* most of the input segments are rejected by the likelihood ratio tests.

6. EXPERIMENTAL EVALUATION

We have evaluated our strategy in two spoken dialogue applications; Car Reservation task and Movie Locator task. The first one involves several interactions of simple utterances, while the latter task is generally completed with a single query of a rather complex sentence. Trials were performed on dialogue systems of the tasks using a speech recognizer. All the data were collected via telephone lines and uttered by general public users.

For evaluation, we define the semantic accuracy in much the same way as the word accuracy. In particular, the semantic error is defined based on the sum of substitution, insertion and deletion errors by matching the content of the semantic slots instead of the recognized words.

The sample utterances are classified into three categories. *In-grammar* sentences consist of valid phrases and are covered by the conventional sentence grammars. *Out-of-grammar* sentences have out-of-vocabulary or fragmental words, or segments with more than one assignment to a semantic slot. *Out-of-task* sentences contain no key-phrases and should be rejected. For a unified definition of the semantic accuracy, we prepare a null slot as an answer for them. Thus, the semantic accuracy for out-of-task samples means the correct rejection rate.

6.1. Car Reservation Task

In the Car Reservation task, a user is prompted to provide specific information to fill the reservation form such as date and location[6]. We refer to each pair of the prompt and the answer specifying such information as a sub-task. Here, we choose the DATE sub-task for the primary evaluation, because it contains the largest number of samples and typical dialogue phenomena. The phrase sub-grammar allows iterations of days of the week, months, days of the month and years with some constraints. The vocabulary size is 99.

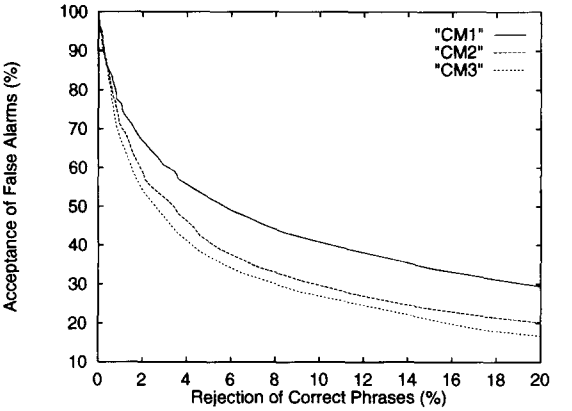

Figure 2. Effect of phrase verification (DATE sub-task)

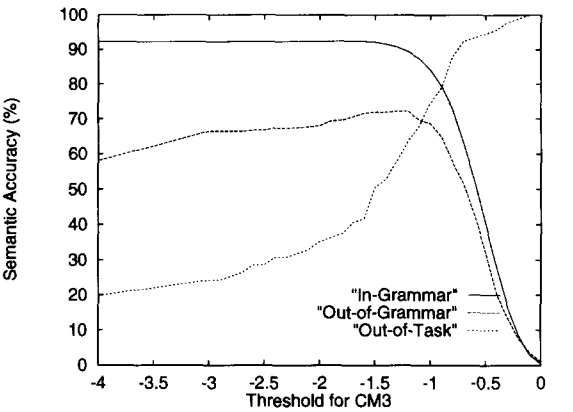

Figure 3. Accuracy vs. threshold (DATE sub-task)

Phrase verification was performed with several confidence measures defined in the previous section. Figure 2 shows comparison of the confidence measures with the acceptance rate of incorrect phrases (false alarms) versus the false rejection rate of correct phrases. The frame duration-based confidence measure (CM_1) is inferior to the subword segment-based ones. The confidence measure CM_3 proposed in this work achieves the best performance. This measure reduces the false alarms to a half with 2.5% rejection of correct hypotheses.

This reduction improves the semantic accuracy of out-of-grammar and out-of-task samples as a result of the sentence parsing. Figure 3 shows the semantic accuracy for each category of samples depending on the threshold values for CM_3. The left-most of the graph corresponds to the baseline detection method without any verification. While the curves for in-grammar and out-of-task utterances are monotonous, there is a performance peak on the out-of-grammar samples that affects choice of the threshold value.

Then, several approaches for speech understanding were investigated. Here, sentence verification was incorporated. For comparison, a rigid grammar was also applied. It is fundmentally the same as the one used for the field trial, and uses the constraint of typical sequences of phrases, which detection does not assume. We also compared with the decoding followed by the verification procedure as in [7].

Table 1. Semantic accuracy with several approaches (DATE sub-task)

	in-grammar samples	out-of-grammar samples	out-of-task samples	total
number of samples	1123	154	91	1368
decoding (with rigid grammar)	92.7%	29.4%	18.7%	83.4%
+ phrase verification ($CM3$)	92.8%	42.3%	39.6%	85.6%
+ sentence verification	92.8%	41.3%	48.4%	85.7%
detection (with phrase network)	92.2%	58.1%	19.8%	86.2%
+ phrase verification ($CM3$)	92.3%	71.6%	41.8%	88.5%
+ sentence verification	92.2%	71.6%	51.6%	88.7%

For phrase verification, CM_3 was adopted. The same beam width was used for all methods.

The results are listed in Table 1. It is clear that our detection strategy outperforms the conventional decoding scheme. It achieves much higher accuracy for out-of-grammar samples while keeping comparable performance for in-grammar ones. Detection with the phrase network almost doubles the accuracy for out-of-grammar samples, and the use of phrase verification improves it further. The verification applied after decoding improves the rejection performance for out-of-task utterances, but it is not so effective in recognizing out-of-grammar samples. This is because key-phrases cannot be recovered from the result of the initial decoding processing with the rigid grammar. The sentence-level verification has little effect, but it improves rejection of out-of-task utterances.

We have done experiments on other sub-tasks in the Car Reservation task and confirmed much the same tendency.

6.2. Movie Locator Task

The Movie Locator task allows a user to make an inquiry on movies being played at theaters. Concretely, a user can ask about movie titles, theaters or the time, by specifying a movie title, a category, a theater or a location area. The utterances are complex and involve multiple phrases as well as extraneous words. We observed a variety of out-of-grammar samples, which constitute more than 25% of the collected samples. The number of samples used for evaluation is 2303, and the vocabulary size is 474.

The phrase network was derived by connecting parallel phrase sub-grammars, while a rigid grammar to cover whole sentences was also used for comparison.

The sentence understanding results are listed in Table 2. Because there were only a few (40) out-of-task utterances in the test database, sentence verification was not tested and results for out-of-task samples are not listed, although the total accuracy counts such samples. Much the same tendency as in the Car Reservation task is confirmed. The detection strategy achieves higher accuracy than the decoding one, and the verification process improves it further. Among the confidence measures, CM_3 is the best.

7. CONCLUSION

We have proposed a key-phrase detection and verification approach oriented for flexible spoken language systems. The experimental results on several tasks demonstrate that the proposed approach is more effective than the conventional decoding with rigid grammars. It drastically improves the accuracy for out-of-grammar utterances while keeping comparable performance for in-grammar ones. The verification applied after decoding is effective only for rejecting out-of-task utterances but does not realize flexible understanding of out-of-grammar ones.

Table 2. Semantic accuracy (MOVIE-2 task)

	in-grammar samples	out-of-grammar samples	total
number of samples	1662	601	2303
decoding	78.1%	33.5%	65.6%
+ verification ($CM3$)	76.8%	42.4%	67.3%
detection	79.2%	44.8%	69.5%
+ verification ($CM1$)	78.9%	45.4%	69.4%
+ verification ($CM2$)	79.5%	47.4%	70.4%
+ verification ($CM3$)	78.0%	51.3%	70.5%

The key property of our framework is portability and generality. Both the detection and verification are vocabulary independent subword-based, thus applicable to a variety of new tasks. Moreover, the language model of the key-phrase network is easily derived from task specifications.

REFERENCES

[1] T.Kawahara, C.H.Lee, and B.H.Juang. Key-phrase detection and verification for flexible speech understanding. In *Proc. ICSLP*, pages 681–684, 1996.

[2] T.Kawahara, N.Kitaoka, and S.Doshita. Concept-based phrase spotting approach for spontaneous speech understanding. In *Proc. ICASSP*, pages 291–294, 1996.

[3] C.-H.Lee, B.-H.Juang, W.Chou, and J.J.Molina-Perez. A study on task-independent subword selection and modeling for speech recognition. In *Proc. ICSLP*, pages 1816–1819, 1996.

[4] R.A.Sukkar and C.-H.Lee. Vocabulary independent discriminative utterance verification for non-keyword rejection in subword based speech recognition. *IEEE Trans. Speech & Audio Process.*, (to appear), 1997.

[5] R.C.Rose, B.-H.Juang, and C.-H.Lee. A training procedure for verifying string hypotheses in continuous speech recognition. In *Proc. ICASSP*, pages 281–284, 1995.

[6] S.M.Marcus, D.W.Brown, and R.G.Goldberg. Prompt constrained natural language – evolving the next generation of telephony services. In *Proc. ICSLP*, pages 857–860, 1996.

[7] E.Lleida and R.C.Rose. Efficient decoding and training procedures for utterance verification in continuous speech recognition. In *Proc. ICASSP*, pages 507–510, 1996.