社団法人　情報処理学会　研究報告
IPSJ SIG Technical Report

2003－ＳＬＰ－49　(49)
2003／12／19

連続音声認識候補受理／リジェクションのためのワードスポッティング仮説検証手法

Frank K. SOONG　Wai-Kit LO　and　中村　哲

ATR 音声言語コミュニケーション研究所，〒619-0288「けいはんな学研都市」光台二丁目２番地２

E-mail:　{frank.soong, waikit.lo, satoshi.nakamura}@atr.co.jp

あらまし　音声認識での単語リジェクション問題は単語スポッティングの枠組みで定式化できる．スポッティングされた単語は２値判定，つまり，受理またはリジェクト判定される．信頼尺度として用いられる一般化された単語事後確率 (Generalized Word Posterior Probability, GWPP) は，forward-backward アルゴリズムによって単語グラフ内で計算されるか，または文ゆう度を用いてＮベストリストにおいて計算される．さらに，同じ単語 ID を持ち，時間的に重なるスポッティングされた単語すべてを組み込むことで GWPP を拡張する．日本語 BTEC 音声データベースでの評価により，信頼誤り率は２つの評価セットに対し，それぞれ 23.76%から 17.78%，20.18%から 15.57%へ著しく減少した．

キーワード　信頼尺度，単語事後確率，大語彙連続音声認識

A Word-spotting Hypothesis Testing for Accepting/Rejecting Continuous Speech Recognition Output

Frank K. SOONG　Wai-Kit LO　and　Satoshi NAKAMURA

Spoken Language Translation Research Laboratories, ATR　2-2-2 Keihanna Science City, Kyoto, 619-0288 Japan
E-mail:　{frank.soong, waikit.lo, satoshi.nakamura}@atr.co.jp

Abstract　The word rejection problem in speech recognition is formulated in a framework of word-spotting, where a spotted word is verified through a binary, acceptance/rejection decision. A generalized word posterior probability (GWPP), used as the sole confidence measure, is computed in a word graph, via the forward-backward algorithm or in an N-best list, using string likelihoods. The GWPP is further enhanced by incorporating all spotted words with the same word ID and overlapped time registrations. When tested on the Japanese BTEC speech database, the confidence error rate is significantly reduced, from 23.76% to 17.78% and 20.18% to 15.57% for the two test data sets, respectively.

Keyword　confidence measure, word posterior probability, large vocabulary continuous speech recognition

1. Introduction

The current state-of-the-art speech recognition has found a wide range of potential commercial applications (e.g., spoken dialog systems, speech translation systems) and for some of them encouraging successes have been obtained. However, the recognition technology is still not robust to changes such as environments, speakers and background noise conditions. The demand for a confidence measure of the recognition output to facilitate an acceptance/rejection mechanism always exists and increases further with more challenging applications.

A desirable confidence measure should be both computationally feasible and statistically meaningful. The word posterior probability has been advocated and tested in a word-graph or N-best list [1,2,3,4].

In this study we reformulate the problem of accepting or rejecting each word recognized by a continuous speech recognizer in a word-spotting framework. In addition to the focused, or the spotted word, all other word hypotheses in the word graph or N-best list are treated as fillers and the generalized word posterior probability of the spotted word is computed. By using this word/filler dichotomy, there is no need to construct a consensus network like the "sausage" [5], lattice chunking [6] or dynamic programming based string alignment [7]. Related issues are discussed and investigated, including:

(1) a reduced string hypothesis search space to test the word-spotting framework in a word graph or an N-best list;

(2) relaxation of the time constraints for finding a "consensus", by grouping the word posterior

probabilities of all reappearances of the hypothesized word;

(3) appropriate weighting of the acoustic and language model probabilities to alleviate the incompatibilities between these two models due to some convenient but not so cogent assumptions.

The rest of the paper is organized as follows. In Section 2 the problem of accepting or rejecting a word in a recognized string is reformulated in a word-spotting framework. In Section 3 and 4, the string and word posterior probabilities are reviewed for their relevance in the HMM-based continuous speech recognition and their appropriateness for measuring confidence are discussed. Modifications needed to make the word posterior probability more efficient to compute and more effective as a good confidence measure are also proposed. In Section 5 and 6, experimental setups and their results are given. Discussions on the experimental results are given in Section 7. In Section 8, a conclusion of this paper is given.

2. Word-spotting based hypothesis testing approach

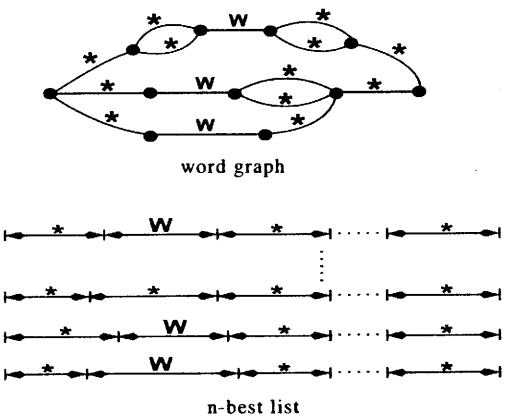

Figure 1. Reduced search space in the form of a word graph and an N-best list with spotted word "w" and fillers "*".

Conceptually it is insightful to view the acceptance/rejection of a recognized word in the framework of word-spotting. Two diagrams, one in a word graph and the other in an N-best list, are shown in Figure 1. Different from the conventional word graph or N-best list where explicit word arc labels are marked, the new diagrams only show the spotted word, w, and all the other words are labeled as fillers by the symbol "*".In the next section, we will review the HMM-based continuous speech recognition by using the posterior string probability first and the posterior string probability will then be used in computing the word posterior probability of each spotted-word individually.

3. String posterior probability

In an HMM-based speech recognizer, the optimal word sequence, $w_1^{*M} = w_1^*, w_2^*, \cdots w_M^*$, for a given acoustic observation sequence, $x_1^T = x_1, x_2, \cdots x_T$, is found by searching over all possible word sequences in the maximum posterior probability (MAP) sense as

$$\begin{aligned} w_1^{*M} &= \arg\max_{M, w_1^M} p(w_1^M \mid x_1^T) \\ &= \arg\max_{M, w_1^M} \frac{p(x_1^T \mid w_1^M)\, p(w_1^M)}{p(x_1^T)} \\ &= \arg\max_{M, w_1^M} p(x_1^T \mid w_1^M)\, p(w_1^M) \end{aligned} \quad (1)$$

where $p(x_1^T \mid w_1^M)$, the acoustic model probability; $p(w_1^M)$, the language model probability; and $p(x_1^T)$, the acoustic observation probability.

It should be noted that string length, M, is also variable in the search. In the last equation for finding the optimal string the term $p(x_1^T)$ can be dropped because it is a constant bias term independent of the choice of the word sequences, w_1^M. Without using $p(x_1^T)$ the optimal word sequence, w_1^M, can still be found by comparing all possible word sequences in terms of their string likelihood calculated by the acoustic and language components, $p(x_1^T \mid w_1^M)$and $p(w_1^M)$, respectively. But for many practical ASR applications, finding the optimal sequence of words is just the first step to make the recognizer useful and the normalization factor $p(x_1^T)$ is still crucial for assessing the reliability of each recognizer word via the posterior probability.

The string posterior probability, $p(w_1^M \mid x_1^T)$, which measures the likelihood of a recognized string, w_1^M, given the acoustic observations x_1^T, is hypothesized with its corresponding time segmentations by the Viterbi search, i.e.,

$$[w;s,t]_1^M = [w_1;s_1,t_1]\cdots[w_M;s_M,t_M]$$

where s and t are the starting and ending time frames of the word w, where $s_1 = 1$, $t_M = T$ and $t_m + 1 = s_{m+1}$ for $1 \le m \le M-1$. By assuming that the acoustic observations, $x_{s_m}^{t_m}$, are dependent solely on the corresponding word, w_m, we can rewrite Eqn.(1) as

$$\begin{aligned} p(w_1^M \mid x_1^T) &= \frac{p(x_1^T \mid [w;s,t]_1^M)\, p([w;s,t]_1^M)}{p(x_1^T)} \\ &= \frac{\prod_{m=1}^{M} p(x_{s_m}^{t_m} \mid w_m)\, p(w_m \mid w_1^{m-1})}{p(x_1^T)} \end{aligned} \quad (2)$$

where the string posterior probability is decomposed into a product of all the acoustic and language model

probabilities of the corresponding word components at the corresponding segmentation points s_m and t_m. The recursive dependency between the current word and preceding words is addressed in the language model (N-gram).

Due to mismatches between training and testing environments, speakers, noises, etc., the "optimal" word sequence may have many potential word errors. A confidence measure, mathematically tractable and statistically appealing, should be adopted to check the reliability or how trustful the whole recognized string or the individual word content of the string is. We shall advocate that the posterior probability, both at the string and word levels, is an ideal choice for our purposes.

4. Word posterior probability

4.1. Basic formulation

The string posterior probability, $p(w_1^M \mid x_1^T)$, is a natural choice for assessing the reliability of the whole recognized string. For an ideal, high performance ASR, accepting or rejecting a string is probably adequate and no checking of the reliability of each word is needed. However, for many practical applications using the current ASR technology, depending upon the vocabulary sizes, task perplexities, operating environments, more than often a string is recognized with some defect, or misrecognized word content. The string posterior probability will reject the whole recognized string frequently, due to the low operating string recognition rate. These defective strings, if not of too poor a quality, should be checked not in terms of their overall string but the individual word reliability.

A confidence measure, appropriate for measuring the word reliability is then the word posterior probability, $p([w;s,t] \mid x_1^T)$, which is defined by summing all the posterior probabilities of strings consisting of the specific word, $[w;s,t]$ at given starting and ending time frames s and t:

$$p([w;s,t] \mid x_1^T) = \sum_{\substack{M,[w;s,t]_k^M \\ \exists n, 1 \le n \le M \\ [w_n;s_n,t_n]=[w;s,t]}} \frac{\prod_{m=1}^{M} p(x_{s_m}^{t_m} \mid w_m)\, p(w_m \mid w_1^{m-1})}{p(x_1^T)} \qquad (3)$$

4.2. Three practical issues

Several practical issues still need to be investigated before the word posterior probability can be used as a practical and functional confidence measure. The issues are presented in the following subsections and the resultant word posterior probability (WPP) we shall refer to as a generalized WPP (GWPP).

4.2.1. Number of hypotheses to be considered in computing the word posterior probability

The search space of all possible word strings in a large vocabulary continuous speech recognizer (LVCSR) is in general very large while each string's posterior probability is quite different, assessed by the corresponding acoustic and language model evidence. As a result, comparing with the top strings, most of the strings have relatively low likelihoods and a much more reduced search space is both desirable and reasonable for computing the word posterior probabilities. In the Viterbi search, a beam is usually imposed to prune out unlikely partial hypotheses and a word graph [1] or an N-best string list [2,3] can be generated by keeping a subset of string hypotheses which are much more likely than other strings in the unpruned, original search space. We will be using such a subset, in computing the word posterior probability in later experiments. Obviously it is an approximation of the true word posterior probability by considering only a much smaller reduced set of strings, but a practical and reasonable one.

When only a subset of strings is used, the acoustic probability, $p(x_1^T)$, in the denominator of Eqn. (3) is then computed by summing up all the string posterior probabilities in the chosen word graph or the N-best list.

4.2.2. The time registration of the word hypothesized

The time registration of the word, denoted by $[w;s,t]$, is hypothesized from the recognized string in the Viterbi algorithm based recognition process. However, it may not be the ground truth of the true beginning and ending time frames of the word, only as a byproduct of the optimal (in the Viterbi sense) word sequence search. Actually, if the forward algorithm, which is in more agreement with the Baum-Welch or the forward-backward HMM model training, is used for continuous ASR and the starting and ending frames of a word hypothesis is by definition "soft", i.e., not fixed. The time registrations of other strings in the reduced search space like a word graph or N-best list may deviate from the given ones. Since the goal of speech recognition is to recognize the word content of an utterance, rather than the exact timing information of each individual word, we shall relax naturally the exact timing constraints imposed by s and t to a softer time interval intersecting constraint. That is, as long as the same word appears in a string with a time interval intersecting (overlapping) with the time interval $[s,t]$, we will include this string in computing the word posterior probability. However, if a word with the same identity but does not intersect with the spotted word, that string where the word resides will not be included in computing the GWPP. An example is given in Figure 2.

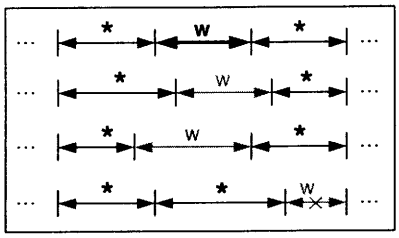

Figure 2. Illustration of the time registration relaxation. The word "w" in the top hypothesis is being spotted. Other strings with "w" appear with intersecting time interval (the second and third string) will be included. String with word "w" but does not intersect (the last string) is excluded.

Eqn.(3) is now

$$p([w;s,t]\mid x_1^T)=\sum_{\substack{M,[w;s,t]_n^M \\ \exists n,1\le n\le M \\ w=w_n \\ [s,t]\cap[s_n,t_n]\neq\phi}}\frac{\prod_{m=1}^{M}p(x_{s_m}^{t_m}\mid w_m)\,p(w_m\mid w_1^{m-1})}{p(x_1^T)} \qquad (4)$$

4.2.3. The weighting of acoustic and language model likelihood

It is well known that in the popular HMM based continuous ASR, some incompatibilities exist between the acoustic model and the language model and some convenient but not quite accurate assumption has been made. They are:

(1) the statistical independence assumption of the neighboring acoustic observations in computing acoustic likelihoods;

(2) the dynamic range difference between the probability density function (pdf) used in the continuous Gaussian mixture based acoustic model and the probabilities (i.e., between 0 and 1) used in the N-gram language model;

(3) the language model likelihood is computed once at every hypothesized word boundary while acoustic model likelihood is computed at every frame interval;

(4) in our computation of WPP, a reduced search space i.e., word graph or N-best list, is used.

To accommodate these modeling discrepancies in practical implementations and to prevent the word posterior probability from being dominated by just a few top strings with high likelihoods, we modified Eqn.(4) to a generalized word posterior probability (GWPP) as

$$p([w;s,t]\mid x_1^T)=\sum_{\substack{M,[w;s,t]_n^M \\ \exists n,1\le n\le M \\ w=w_n \\ [s,t]\cap[s_n,t_n]\neq\phi}}\frac{\prod_{m=1}^{M}p^{\alpha}(x_{s_m}^{t_m}\mid w_m)\,p^{\beta}(w_m\mid w_1^{m-1})}{p(x_1^T)} \qquad (5)$$

Obviously, the denominator term, $p(x_1^T)$, when summed over all string hypotheses in the reduced search space of a word graph or an N-best list, need to be scaled by α and β accordingly to make sure the normalization is appropriate.

5. Experimental setup

In this study, the word-spotting framework for word acceptance/rejection has been tested using the Japanese Basic Travel Expression Corpus (BTEC) [8]. Two testing sets were used, namely set01 and set02 with 510 and 508 utterances, respectively. Each data set contains different utterances recorded from 10 different speakers who are different from one set to the other. The recognition systems used in our experiments is the ATRIUMS Version 2.2 from ATR [9]. Specifically, for our investigation, the LVCSR is configured to generate 100-best hypotheses recognition output together with the word graph for every utterance and the search is constrained with a narrow beam width to investigate its potential implications in running real-time demonstrations.

5.1. Performance measures

The purpose of a word acceptance/rejection decision process is to identify potentially erroneous words in recognition output. These potentially erroneous words are verified and a rejection or an acceptance decision is then made on each individual word. In general, there are two kinds of decision errors, false rejection (FR) when a correctly recognized word is rejected and false acceptance (FA) when a misrecognized word is accepted.

Confidence error rate (CER) [7] can be used as a performance measure for word acceptance/rejection decision. CER is defined as the ratio of all errors (FA + FR) to the total number of recognized words.

$$\text{CER}=\frac{\#\text{ of false acceptances}+\#\text{ of false rejections}}{\text{total }\#\text{ of word in the recognition output}} \qquad (6)$$

It should be noted that deleted words cannot not be recovered via a word acceptance/rejection decision and therefore they are not included in the evaluation measure. In the acceptance/rejection experiments, there are also two relevant performance measures, error recall and rejection precision. Among them, error recall measures the performance of correctly identifying errors by the ratio between the correctly identified errors and the total number of errors given as

$$\text{CER}=\frac{\#\text{ of false acceptances}+\#\text{ of false rejections}}{\text{total }\#\text{ of word in the recognition output}} \qquad (7)$$

and rejection precision measures the accuracy of correct rejections by the ratio between the correctly identified errors and the total number of rejections given as

$$\text{Rejection precision} = \frac{\#\text{ of correct rejections}}{\text{total }\#\text{ of rejected words}} \quad (8)$$

6. Results and Analyses

If we accept all recognition output from the LVCSR without any rejection, all decision errors are false acceptance of incorrect words, including insertions or substitutions. This is used as the baseline CER performance in our experiments.

A simple and intuitively appealing approach to compute the generalized word posterior probability is by counting the number of hypotheses where the specific word appears, denoted as the reappearance of a word. The ratio between this count and the total number of hypotheses returned by the LVCSR can give a quantized (either 0 or 1 for every string), rough estimate of the generalized word posterior probability. This word reappearance rate approach is equivalent to setting both α and β in Eqn.(5) to zero.

6.1. N-best list

Figure 3 shows the total decision errors at different combinations of α and β in a contour plot. The total errors (false acceptance + false rejection) are shown with a gray level of intensity, i.e., the larger the error counts, the lighter the intensity. We first made a wider range, coarse scale search to get a global view of the general behavior of decision errors with respect to these weights. A finer grid search around the "near optimal" region located is then performed. The final operating points are determined in the finer grid search and the optimal pair of α and β yielding the lowest total errors are obtained.

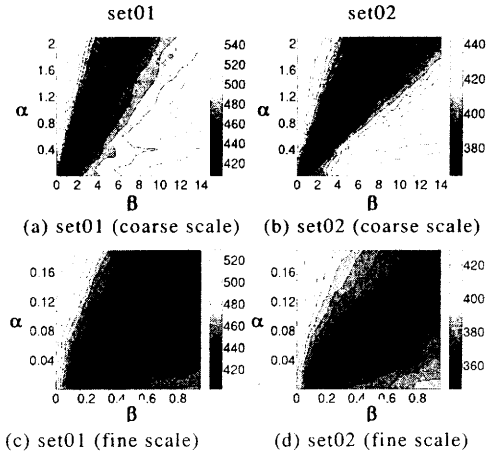

Figure 3. Contour plots of total errors at different acoustic (α) and language model (β) weights when using GWPP derived from N-best list (N<=100). Along the vertical axis of α and the horizontal axis of β, the total errors are intensity-coded, i.e., the darker the gray level, the lower the total errors. The corresponding scale for each of the contour plots is given as a vertical bar to the right. (a) and (b) are wider range contour plots on a coarser scale. (c) and (d) are obtained at a finer grid around the optimal regions.

In Table 1, the optimal performance located from the contour plots in Figure 3 is shown together with the baseline. As mentioned before, when α and β are both set to zero, it is actually counting the number of hypotheses with the reappearance of the spotted word in the word graph or the N-best list. We also made use of set01 as the development set to obtain the optimal values of α, β and the decision threshold, and then applied to set02 for testing. The same process was also applied to set02 and tested on set01. We obtained CER at 17.90% and 16.15% for set01 and set02, respectively. These correspond to relative improvements of 24.6% and 19.9% in CER, respectively.

N-best list	set01	set02
baseline	23.76	20.18
α=0, β=0	21.13 (22.68, 66.15)	17.67 (29.93, 63.08)
α=0.09, β=0.4 (set01 optimal)	17.06* (49.29, 70.05)	**16.15** **(39.47, 66.92)**
α=0.01, β=0.2 (set02 optimal)	**17.90** **(43.75, 69.60)**	15.48* (39.47, 70.92)

Table 1. Performance of word acceptance / rejection decision based on GWPP derived from N-best list. The performance is measured in CER (%). The error recall and rejection precision rates (%) are included in the parentheses, respectively. Values with "*" are the optimal performance in a closed test.

6.2. Word graph

Figure 4 shows the contour plots of total errors at different combinations of α and β when the GWPPs are derived from word graphs.

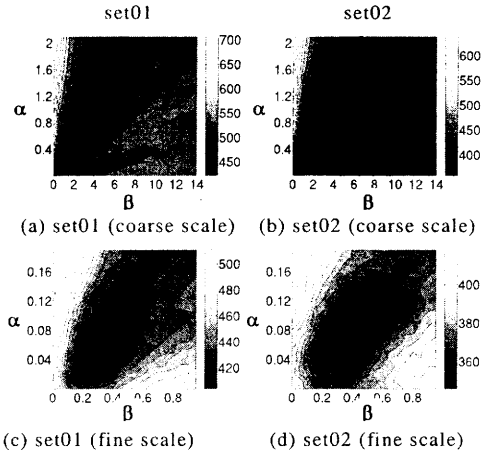

Figure 4. Contour plots of total errors at different acoustic (α) and language model (β) weights when using GWPP derived from word graphs. Along the vertical axis of α and the horizontal axis of β, the total errors are intensity-coded, i.e., the darker the color, the lower the total errors. The corresponding scale for each of the contour plots is given as a vertical bar to the right. (a) and (b) are wider range contour plots on a coarser scale. (c) and (d) are obtained at a finer grid around the optimal regions.

Table 2 shows the optimal performance located from the contour plots from Figure 4 where word graphs are used. These results show that a CER of 17.78% and 15.57% can be obtained from set01 and set02, respectively. These correspond to relative improvements of 25.1% and 22.8% in CER, respectively.

Word graph	Set01	set02
Baseline	23.76	20.18
α=0, β=0	21.34 (19.64, 67.48)	18.12 (21.95, 65.13)
α=0.04, β=0.3 (set01 optimal)	17.14* (45.00, 72.41)	**15.57** **(38.80, 70.85)**
α=0.05, β=0.2 (set02 optimal)	**17.78** **(46.96, 68.31)**	15.48* (40.80, 69.96)

Table 2. Performance of word acceptance / rejection decision based on GWPP derived from N-best list. The performance is measured in CER (%). The error recall and rejection precision rates (%) are included in the parentheses, respectively. Values with "*"are the optimal performance in a closed test.

7. Discussions

In these experiments, the generalized word posterior probability (GWPP) is used as the confidence measure while its acoustic model weight and language model weight are optimized by minimizing the confidence error rate (CER), or the total decision errors (sum of false acceptances and false rejections). Also, on average an error recall of 42% and rejection precision of 69% are achieved.

Our experimental results also show that the optimal operating region is closer to the origin than the original weights used in the decoding (recognition) process. Assigning zero to both α and β is equivalent to counting, while larger values emphasizing top-ranking hypotheses in computing the GWPP. In the extreme case, when α and β are set to ∞, only the best hypothesis is considered. By optimizing the values of α and β, an optimal weighted combination of hypotheses is used in computing GWPP.

It is also observed that there is a "preferred ratio" between α and β, which can be observed from Figure 3 and Figure 4. Along the slope, a shaded area of "preferred ratio", we can locate the sub-optimal combinations of α and β for any fixed value of α (or β). One may search for the truly optimal operating point by identifying the "preferred ratio" line and then search for the globally optimal point along this identified line. This is more efficient than a full-grid search among different combinations of α and β.

When the weights (α and β) are closed to zero, there is larger variations of performance. For GWPP derived from N-best list, the total errors tend to settle at a steady value. For GWPP in a word graph, the total errors increase significantly towards β=0. This is because the maximum number of hypotheses in an N-best list is more restricted, i.e., clamped to N, than that of a word graph. As mentioned above, use of small weights will "wash out" the dominance of top-ranking hypotheses. In an N-best list, at most N hypotheses will have evenly distributed contributions to GWPP computation. However, in a word graph, more lower-ranking hypotheses with bad acoustic scores will contribute to the GWPP computation. As a result, the total errors continue to increase.

8. Acknowledgement

The research reported here was supported in part by a contract with the Telecommunications Advancement Organization of Japan entitled, "A study of speech dialogue translation technology based on a large corpus".

References

[1] S. Ortmanns, H. Ney, and X. Aubert, "A Word Graph Algorithm for Large Vocabulary Continuous Speech Recognition," Computer Speech and Language, vol. 11, no. 1, pp. 43–72, Jan.1997.

[2] R. Schwartz and Y.-L. Chow, "The N-best Algorithm: An Efficient and Exact Procedure for Finding the N Most Likely Sentence Hypotheses," Proc. of ICASSP1990, vol. 1, pp. 81–94, Albuquerque, New Mexico, USA, Apr.1990.

[3] F. Soong and E.-F. Huang, "A Tree-Trellis Based Fast Search for Finding the N Best Sentence Hypotheses in Continuous Speech Recognition," Proc. of ICASSP1991, vol. 1, pp. 705–708, Toronto, Canada, May.1991.

[4] A. Lee, K. Shikano, and T. Kawahara, "Confidence Scoring Based on Recognition Engine Julius," The 2003 Autumn Meeting of the Acoustical Society of Japan, pp.117-118, Nagoya, Japan, Sep.2003.

[5] L. Mangu, E. Brill, and A. Stolckes, "Finding Consensus in Speech Recognition: Word Error Minimization and Other Applications of Confusion Networks," Computer Speech and Language, vol. 14, no. 4, pp. 373–400, Oct.2000.

[6] D. Hakkani-Tur and G. Riccardi, "A General Algorithm for Word Graph Matrix Decomposition," Proc. of ICASSP2003, vol. 1, pp. 596–599, Hong Kong, China, Apr.2003.

[7] F. Wessel, R. Schluter, K. Macherey, and H. Ney, "Confidence Measures for Large Vocabulary Continuous Speech Recognition," IEEE Transactions on Speech and Audio Processing, vol. 9, no. 3, pp. 288–298, Mar.2001.

[8] T. Takezawa, E. Sumita, F. Sugaya, H. Yamamoto, and S. Yamamoto, "Toward a broad-coverage bilingual corpus for speech translation of travel conversations in the real world," Proc. of LREC2002, pp. 147-152, Canary Islands, Spain, May.2002.

[9] T. Shimizu, H. Yamamoto, H. Masataki, S. Matsunaga, and Y. Sagisaka, "Spontaneous Dialogue Speech Recognition Using Cross-Word Context Constrained Word Graph," Proc. of ICASSP1996, vol. 1, pp. 145-148, Atlanta, Georgia, U.S.A., May.1996.