Optimal Acoustic and Language Model Weights for Minimizing Word Verification Errors
Frank K. Soong  Wai-Kit Lo  Satoshi Nakamura
Spoken Language Translation Research Labs, ATR, Kyoto, Japan
{frank.soong, waikit.lo, satozi.nakamura}@atr.jp

Abstract
Generalized word posterior probability (GWPP), a confidence measure for verifying recognized words, needs to equalize and weight acoustic and language model likelihood contributions to minimize verification errors. In this study, we investigate the word verification error surface and use it to optimize these weights and the corresponding verification threshold in a development set. We test three different search algorithms for finding the optimal parameters, including: a full grid search, a gradient-based steepest descent search, and a downhill simplex search. The three search methods yield very similar solutions. Proper acoustic and language model weights, especially the ratio between them, changes with the relative importance (reliability) between the two knowledge sources. For a narrow beam width, the role of the acoustic model is less critical than language model in GWPP-based word verification, which is due to the noisy acoustic information maintained in a narrow beam. Using a large vocabulary continuous Japanese speech database (Basic Travel Expression Corpus), the largest relative improvement obtained is 33.2% for confidence error rate and 38.7% for a modified word accuracy.

1. Introduction
The current state-of-the-art speech recognition technology is still vulnerable to environmental changes such as noise and speaker variability. Selectively to accept/reject LVCSR word output can help improve the performance and usability of a spoken language system. By assessing confidence of speech recognition output, reliability information can be tagged and appropriate actions can be taken to improve the performance.

Confidence measure has found many applications in spoken language systems. For instance, a speech translation system can take advantage of some additional word reliability information to put more weight on reliable words. Similarly, such information can also be employed in a spoken dialogue system to decide whether to prompt the user to speak the whole utterance again, or to confirm the uncertain part only. This can improve the user interface by cutting down the number of unnecessary or superfluous dialogue turns.

There have been various approaches proposed for measuring confidence of speech recognition output. They can be roughly classified into three categories: i) feature-based; ii) explicit (extra) model based; and iii) posterior probability based. Feature based approaches [1] try to assess the confidence according to selected features (e.g., word duration, part-of-speech, acoustic and language model back-off, word graph density, etc.) using some trained classifiers. Explicit model based approaches employ a candidate model together with some competitor models [2,3] (e.g., anti-model or filler model, etc.) and a likelihood ratio test is usually applied. Finally, posterior probability based approach tries to estimate the posterior probabilities of a recognized entity (e.g., subword, word, or sentence) given all the acoustic observations [4,5].

In this study we generalize the concept of word posterior probability (WPP) to take into account the practical limitations in computing the WPP. Specifically, how to optimize the exponential weights of the acoustic and language models so as to minimize word verification errors is investigated in details. In a separated paper, also presented in this proceedings, we investigate the word verification in noisy environments using the generalized word posterior probability [6].

2. Generalized Word Posterior Probability
In maximum a posteriori (MAP) based speech recognition, the best recognized word string \( w^M \) is obtained by maximizing the corresponding string posterior probability (SPP) as

\[
\hat{w}^M = \arg\max_{w} p(w^M | x^T)
\]

\[
= \arg\max_{w} p\left(\frac{x^T | w^M}{p(x^T)}\right) p(w^M)
\]

\[
= \arg\max_{w} p\left(\frac{x^T | w^M}{p(w^M)}\right)
\]

(1)

While the string posterior probability is given as quotient, the denominator \( p(x^T) \) in (1) can be ignored in maximization since it is independent of the choice of the recognized word sequence. The word posterior probability (WPP) is similarly defined as the SPP. WPP is computed by summing posterior probabilities of all string hypotheses in the search space bearing the focused word, \( w \), at time \( s \) and ending at time \( t \). It is given by

\[
p([w;s,t] | x^T) = \sum_{M} \prod_{n=1}^{M} p\left(\frac{x^T_{w_n} | w^M}{p(x^T)}\right) p(w_n | w^M)
\]

(2)

where a word hypothesis is defined by the corresponding triple, \([w; s, t]\).

WPP is computed without using any additional model (e.g., anti-models), and hence no extra model training is needed. Also, WPP of each recognized word can be readily computed from the word graph or the N-best list, a by-product of LVCSR.
2.1. Three issues in computing WPP

In order to make the word posterior probability to be more effective for word verification applications, Eqn. 2 needs some modifications in practice [4,5]. Three issues are addressed here and the word posterior probability is generalized by considering: i) reduced search space; ii) relaxed time registration; and iii) optimal acoustic and language model weights for the best word verification performance.

2.1.1. Reduced search space

In an LVCSR, the search space in recognition has always to be pruned to make the search tractable. The reduced search space (e.g., word graph or N-best list) can also be conveniently used when computing the GWPP. The acoustic observation probability, \( p(x_{1:t}^i) \) (denominator in Eqn. 2), for normalization is computed using the same reduced rather then the original full search space [4,5].

2.1.2. Relaxed time registration

The definition of a word hypothesis in Eqn. 2 is a triple of word identity, and its given starting and ending time. However, the starting and ending time of a word in LVCSR output is only a by-product of the search and can be affected by various factors, e.g., pruning threshold, model resolutions, noise, etc. Experimentally it is found that the same word can have close but slightly different starting and ending time frames in different decoded string hypotheses. To decide whether a word reappears in a different string hypothesis, the time registration requirement needs to be relaxed. For example, words with the same identity and overlapping (but not identical) time registrations should also be considered as reappearances. This overlapping requirement may be adjusted, for example, by imposing a lower limit in the time overlap proportion.

2.1.3. Optimal acoustic and language model weights

In a conventional continuous ASR, some convenient but not quite accurate assumptions have been made to facilitate the modeling and decoding process. There are also some model incompatibilities among components in Eqn. 2. They include:

1. Difference in the dynamic range: The acoustic likelihoods computed by using a continuous Gaussian mixture HMM is based on the probability density functions which, in theory, have an unbounded dynamic range while the language model likelihoods based on the N-gram probabilities have values between 0 and 1.
2. Difference in the frequency of computation: Acoustic likelihoods are computed every frame and language model probabilities are computed only once per word.
3. Independence assumption: Neighbouring acoustic observations are assumed to be statistically independent in computing the acoustic likelihoods, a convenient but obviously wrong assumption.
4. Reduced search space: In practice, search space is reduced by pruning a word graph or an N-Best list of hypotheses is thus generated.

In order to compensate for all the above problems, we adjust the acoustic and language model weights jointly to optimize the word verification performance. The acoustic and language model weights are labeled as \( \alpha \) and \( \beta \), respectively and the final generalized word posterior probability (GWPP) is given as

\[
p(W_{x:t}; I \mid x_{1:t}^i) = \frac{\prod_{m=1}^{M} p^n(\phi_{w_m} \mid w_{x_{1:t}^i}) \cdot p^\beta(w_m \mid w_{x_{1:t}^i})}{p(x_{1:t}^i)} \tag{3}
\]

Note that in the generalized word posterior probability the reduced search space, either in a word graph or N-best list, is adopted in both the numerator and denominator computation in Eqn. 3. Also, time registrations are relaxed in finding the word reappearance in different string hypotheses; and \( \alpha \) and \( \beta \) are jointly optimized using a development set to minimize word verification errors.

3. Experimental Setups

3.1. Speech corpus

The corpus used in our experiments is a large vocabulary, continuous, read Japanese speech database called the Basic Travel Expression Corpus (BTExC) [7]. It was compiled and collected for a travel domain speech-to-speech translation project. In particular, two data sets are used as development and test sets in this study, consisting of 508 and 510 utterances, respectively. Each set has 10 speakers (gender balanced) reading different sentences in the travel domain.

3.2. LVCSR

The LVCSR used is the ATRASR [8], running in multi-pass with a word bigram language model and a 47k word lexicon. Generated word graphs in the recognition process are then rescored using a word trigram language model to obtain the final recognition output. In this paper, experiments have been carried out with different beam widths that in turn produce word graphs of different densities. The beam widths labeled as “narrowest”, “narrow”, “wide”, and “widest” have word graph densities of 6.5, 10.2, 16.7, and 80.2 respectively.

3.3. GWPP computation and word verification

In order to compute the GWPP efficiently for each of the recognized words in the LVCSR output, the forward-backward algorithm is applied to calculate both the numerator and denominator in Eqn. 3. Optimal parameters (\( \alpha \), \( \beta \) and verification threshold) are determined from the development set and then applied to the test set. Words with GWPP below the threshold are rejected and the remaining words are accepted.

3.4. Performance evaluation metrics

Two metrics are used for evaluating our word verification performance: confidence error rate (CER) [4] and modified word accuracy (mAcc). CER is a measure of the total number of verification errors, normalized by the total number of recognized words in the LVCSR output. It has a range from 0 to 1 (the smaller the better) and is defined as

\[
CER = \frac{\#\text{false acceptance} + \#\text{false rejection}}{\#\text{recognized words}} \times 100\% \tag{4}
\]
A modified word accuracy (mAcc), a measure proposed here to evaluate word accuracy performance after word verification, is defined as

\[
\text{mAcc} = \frac{\#\text{recog} - \#\text{delete} - \#\text{subs} - \#\text{insert} - \#\text{FR} + \#\text{CR}}{\#\text{ref}} \times 100\%
\]  

where \#recog: the no. of recognized words; \#delete, \#subs, \#insert: deletion, substitution and insertion errors; \#ref: words in reference transcriptions; \#FR and \#CR: false and correct rejections; \#CRsub: correct rejections of substituted words.

\[\text{(5)}\]

than that of baseline, we chose the origin as the starting point.

In order to study the characteristics of verification errors, a full grid search is applied to the data. Example contour plots of word verification errors are shown in Fig 1. In general, a dark region (least errors) is observed at the lower left corner where both weights are small. A finer grid search around this region is then applied to obtain the optimal \(\alpha\) and \(\beta\).

Based on the observed error distributions, more efficient search methods are devised, including: a gradient-based, grid-constrained steepest descent method and the Downhill Simplex Method [9].

The steepest descent method searches for the minimum by starting from an initial point. In each step, differences of errors between its surrounding grid points are calculated. This initial point is then replaced by the point that yields the greatest error reduction (steepest descent). This process repeats itself until no further improvement is obtained. Since total errors at the origin (\(\alpha, \beta\)=0, i.e., rprate) are always fewer than that of baseline, we chose the origin as the starting point.

Downhill simplex method is a simple and effective minimization method that needs no derivative computation [9]. Given an initial simplex (a triangle in a 2-D search), it iteratively updates the vertices of the simplex in the direction towards the minimum error. This algorithm terminates when the difference between the maximum and the minimum among all vertices of the simplex falls below a threshold.

Fig 2 shows the word verification errors obtained by the three search methods at different beam widths. The results are essentially the same, however, with different computational complexities.

\[\text{Fig. 2: Total errors obtained in full grid, steepest descent and downhill simplex searches at different beam widths.}\]

Fig 3 shows the error surfaces for four different beam widths. The optimal parameters (\(\alpha, \beta\) threshold) for the “narrowest”, “narrow”, “wide”, and “widest” beam widths are (0.06, 0.6, 0.97), (0.02, 0.3, 0.68), (0.03, 0.3, 0.058) and (0.06, 0.7, 0.57), respectively.

Full grid searches for word graphs generated using four different beam widths were carried out to investigate the sensitivity of GWPP to changes in recognizer settings. Error surfaces obtained are shown in Fig 3 where the total verification error contours are plotted in gray levels with respect to the corresponding acoustic weight (\(\alpha\)) and language model weight (\(\beta\)). As it is shown in the figure, the darker the gray level is, the fewer the verification errors are. Also, a “preferred ratio” between the acoustic and language model weights can be observed as a low-error valley (darker region) on the error surface. With a reduced beam width, the slope of this valley increases, from a more diagonal toward a more vertical direction. This indicates that the sensitivity of the word verification errors with respect to the acoustic weight is less than the language weight in a narrower beam width. In other words, due to the lower quality of the acoustic

\[\text{Fig. 3: Error surfaces for four different beam widths. The optimal parameters (\(\alpha, \beta\) threshold) for the “narrowest”, “narrow”, “wide”, and “widest” beam widths are (0.06, 0.6, 0.97), (0.02, 0.3, 0.68), (0.03, 0.3, 0.058) and (0.06, 0.7, 0.57), respectively.}\]

- **Fig. 1: Total error contours at different acoustic (\(\alpha\)) and language (\(\beta\)) model weights in coarse and fine grid.**

- **Fig. 2: Total errors obtained in full grid, steepest descent and downhill simplex searches at different beam widths.**

- **Fig. 3: Error surfaces for four different beam widths. The optimal parameters (\(\alpha, \beta\) threshold) for the “narrowest”, “narrow”, “wide”, and “widest” beam widths are (0.06, 0.6, 0.97), (0.02, 0.3, 0.68), (0.03, 0.3, 0.058) and (0.06, 0.7, 0.57), respectively.**
likelihoods in a narrow search beam, the value of acoustic weight is not very critical in minimizing the word verification errors. Nevertheless, these changes do not induce significant shift of the optimal region, which is still near the origin for all tested beam widths.

Fig 4 shows the CER and mAcc performance when parameters (α, β and threshold) determined from the development set is applied to the test set. The bar for 'optimal' in CER shows the performance obtained by searching the test set for optimal parameters in a posterior sense. Hence it marks the upper bound of the verification performance. As shown in the figure, the test results in CER (testing) improve significantly over the baseline and approach the upper bound for all beam widths. The performance improvement over the baseline increases with a narrower beam width, where more recognition errors are made. The maximum improvement of CER is at the narrowest beam width where 33.2% relative improvement is obtained (from 31.6% to 21.1% in CER) and the corresponding modified word accuracy improvement is 38.7% (from 62.6% to 86.8%).

For the observed distinctive preferred ratio across different search beam widths, it means that a proper ratio between the acoustic and language model is beneficial. When a narrower beam width (i.e., heavier pruning) is adopted, the slope of this preferred ratio increases, or a precise choice of acoustic model weight becomes less critical than that of language model.

Investigation on the search methods for determining the optimal parameters shows that the operating parameters can be determined efficiently based on our knowledge of the error surface characteristics learned from the full grid search. Experimental results show that both the steepest descent and downhill simplex methods are effective and can achieve statistically identical performance, comparing to the more computationally expensive full grid search.

6. Conclusions

GWPP consistently improves the word verification performance with the largest relative improvement of CER over the baseline performance at 33.2% and mAcc at 38.7%, where the narrowest search beam width is used and the word recognition performance is the poorest. The optimal acoustic and language model weights for minimizing the word verification errors can be found from several different search methods efficiently. It is also found that the word verification error surface with respect to the acoustic and language model weights shows a low-error valley along a fixed slope. With a narrower beam width, the slope of this valley is tilted upward (i.e., more vertical) when the importance of acoustic evidence diminishes with the heavy pruning in a narrow beam.

7. Acknowledgements

This research was supported in part by the National Institute of Information and Communications Technology.

8. References