AUTOMATIC SPEECH RECOGNITION USING ACOUSTIC CONFIDENCE CONDITIONED LANGUAGE MODELS

Richard C. Rose and Giuseppe Riccardi
AT&T Labs-Research, Shannon Laboratory
180 Park Avenue, Florham Park NJ 07932-0971
email: rose,dsp3@research.att.com

ABSTRACT

A modified decoding algorithm for automatic speech recognition (ASR) will be described which facilitates a closer coupling between the acoustic and language modeling components of a speech recognition system. This closer coupling is obtained by extracting word level measures of acoustic confidence during decoding, and making coded representations of these confidence measures available to the ASR network during decoding. A simulation of this decoding strategy is implemented using a word lattice rescoring paradigm. A joint acoustic–language model will be described where linguistic context is augmented to include the encoded values of acoustic confidence. Finally, the performance of the word lattice based implementation of the decoding algorithm will be evaluated on a large vocabulary natural language understanding task.

1 INTRODUCTION

This paper addresses two issues in acoustic and language modeling for automatic speech recognition. The first issue that the paper attempts to address is the lack of integration between the acoustic and language modeling component of ASR systems. Language models are generally trained from perfect utterance transcriptions even though search algorithms in ASR should propagate paths in a network by optimizing the joint acoustic and language model probabilities. The second issue involves problems with assigning local confidence measures to words in a hypothesized string of words that is produced by a speech recognizer. It is well known that these locally derived confidence measures can be misleading in circumstances where the word is correctly decoded as a result of the language model even when the acoustic evidence is weak.

To deal with these issues, a procedure is proposed for incorporating acoustic confidence measures derived during ASR decoding directly in the language model. The techniques described here rely on model based techniques for extracting word level acoustic confidence measures that were originally presented in [lleida-96]. Furthermore, the general notion of incorporating acoustic confidence measures in stochastic language models (LMs) has been discussed in [Rose98]. The novel aspect of the work described in this paper is the implementation and evaluation of a decoding algorithm which makes continually updated measures of confidence available to the network during decoding. In our previous work, language models were trained to incorporate prior distributions of acoustic confidence, allowing separate states to exist in the LM for representing each of a number of levels of confidence [6]. Section 2 of the paper describes a word lattice rescoring approach to simulating a decoder which makes direct use of dynamically derived acoustic confidence measures. The parameterization and training of the language model used in this integrated decoder is discussed in Section 3. Finally, the LVCSR task and the experimental results reported for the integrated decoder on that task are given in Sections 4 and 5 respectively.

2 DECODING STRATEGY

This section describes the proposed decoding strategy used for obtaining closer integration of acoustic and language models during recognition. First, the basic structure of a stochastic language model that incorporates coded measures of acoustic confidence is briefly reviewed. Second, a decoder that allows these dynamically computed acoustic confidence measures to be passed to the language model in a single recognition pass is discussed. Finally, an implementation of the integrated decoder based on a word lattice rescoring paradigm is described and evaluated.

2.1 Model Integration

A stochastic LM is generally defined over the elements of a K length word sequence, $W = w_1, \ldots, w_K$, for an utterance where $w_i \in V$, and V is the lexicon for the task. The approach that is taken in this work involves augmenting the lexicon of symbols to include symbolic representations of acoustic confidence [1]. Word level acoustic confidence measures can be extracted and encoded as $c_i \in [0, \ldots, Q-1]$, a discrete, Q level encoding of the acoustic confidence for word w_i. If these two information sources are synchronous, then the acoustic and lexical information can be coupled so that decoding can be based on their joint distribution. Thus, the word string can be replaced by a symbol–pair sequence so that an utterance will be represented by $W, C = (w_1, c_1), (w_2, c_2), \ldots, (w_K, c_K)$. For example, the acoustic and lexical context for word w_i in a trigram LM would be augmented from $\{w_{i-1}, w_{i-2}\}$ to $\{(w_{i-1}, c_{i-1}), (w_{i-2}, c_{i-2})\}$. Since the overall effect of this language model is to condition language model probabilities on estimated measures of acoustic confidence, it is referred to below as an acoustic confidence conditioned (ACC) language model.

The interaction between the acoustic decoder and the language model (LM) in a single pass implementation of this system is illustrated by the block diagram in Figure 1. In the system in Figure 1, acoustic confidence scores are computed directly in the continuous speech recognition decoder, encoded as discrete symbols, and

passed to the ACC language model. The estimation of these confidence scores is normally performed as part of a two pass procedure. A hypothesized word string is generated by a CSR decoder in the first stage, and a confidence measure is computed in the second stage. While there have been many different techniques proposed for estimating word level confidence measures in automatic speech recognition [5], an acoustic likelihood ratio based approach is taken in this work. The acoustic confidence scores themselves are derived from the ratio of "target" hypothesis hidden Markov model and alternate hypothesis hidden Markov model likelihoods [5]. The configuration in Figure 1 assumes that the CSR decoder can produce word based acoustic confidence scores directly without the need for a dedicated acoustic utterance verification (UV) subsystem. This is enabled by a single–pass CSR decoder / utterance verification system which is based on a CSR decoder designed to directly optimize a likelihood ratio criterion [2].

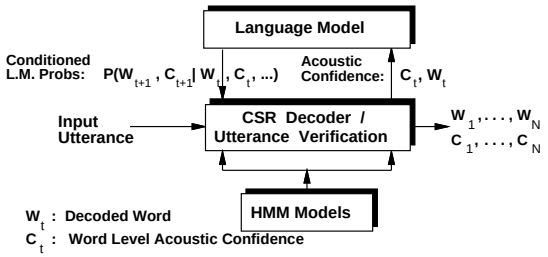

Figure 1: Block diagram describing the interaction between the network and the acoustic decoder for the integrated decoder using a single pass CSR decoder / UV subsystem.

The n–best candidate rescoring implementation of a decoder based on the ACC language model is shown in the block diagram in Figure 2. In this implementation, a list of n–best word candidates are extracted from word lattices produced by a "baseline" continuous speech recognizer and passed to the UV subsystem so that word level acoustic confidence scores can be assigned to the words in the n–best hypothesized strings. The stochastic language model for the baseline CSR is a phrase–based bigram that is defined strictly over the elements of the word lexicon V and does not incorporate any notion of acoustic confidence. The UV subsystem relies on dedicated acoustic models that are estimated to maximize an average likelihood ratio criterion [5]. These UV or confidence labeled word strings are then rescored using the ACC language model. The best hypothesized word string candidate is chosen as the string with the highest combined ACC language model score and acoustic UV score. The advantage of the implementation shown in Figure 2 is that it is possible to evaluate the effects on performance of any of the subsystems in the Figure in isolation. An experimental study where this evaluation is performed is presented in Section 5.

2.2 Decoder Implementation

The n–best rescoring implementation of the decoder involved three steps. The first step involved generating a word lattice for each utterance in the test set. The word lattices took the form of weighted finite state acceptors whose arcs were associated with word symbols, w_i, and weights equal to the combined acoustic and baseline language model costs [3]. The N highest scoring paths were extracted from the word lattices giving N strings with

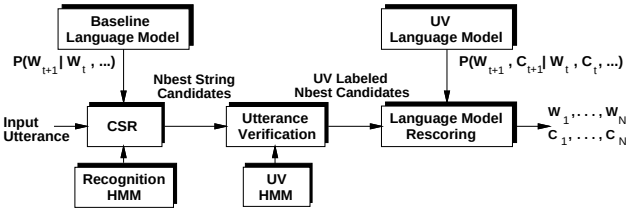

Figure 2: Implementation of the integrated decoder using a word lattice rescoring paradigm.

baseline log probabilities

$$\log P_{BL} = \sum_{k=1}^{K} \mathcal{K} \log P(w_k|w_{k-1}) + \log A(w_k) \ , \qquad (1)$$

where $A(w_k)$ corresponds to the acoustic score computed for word w_k during recognition and $\mathcal{K}$ is a grammar scale factor used to adjust the relative weight of language and acoustic model probabilities during recognition. The language model component of the path score in Equation 1 was removed before ACC language model rescoring was performed.

The second step was to apply the utterance verification techniques described in Section 2.1 to assign word level acoustic confidence scores, $S(w_k)$, to each word in each of the N candidate strings. These confidence scores were then compared to a decision threshold so that a binary coded representation of the confidence score, c_k, could be associated with each word

$$c_k = \begin{cases} 1 & S(w_k) > \tau \\ 0 & \text{otherwise} \ . \end{cases} \qquad (2)$$

This results in a word/confidence sequence (w_k, c_k) $k = 1, \ldots, K$ for each string hypothesis of an utterance and a finite state acceptor whose arcs contained symbol pairs (w_k, c_k).

Finally, the third step involved in the n–best rescoring implementation was the actual language model rescoring. The ACC language model, which is described in more detail in Section 3, also took the form of a stochastic finite state acceptor (FSA). The symbols on the arcs correspond to the pairs (w_k, c_k) and the arc weights are the language model costs estimated as described in Section 3. The ACC language model rescoring was implemented for each utterance by composing the ACC language model FSA with the FSA representation of the n–best candidate utterance.

Two different cases are investigated for assigning acoustic scores. In one case, the baseline word level acoustic likelihoods, $A(w_k)$, were used as arc weights resulting in path log probabilities computed as

$$\log P_{ACC} = \sum_{k=1}^{K} \mathcal{K} \log P((w_k, c_k)|(w_{k-1}, c_{k-1})) + \log A(w_k) \ .$$

In the second case, the word level acoustic confidence scores, $S(w_k)$, were assigned as arc weights resulting in total path probabilities computed as

$$\log P_{UV} = \sum_{k=1}^{K} \mathcal{K} \log P((w_k, c_k)|(w_{k-1}, c_{k-1})) + \log S(w_k) \ .$$

Paths are propagated based on acoustic confidence scores and language model states are also dependent on representations of estimated acoustic confidence.

3 LANGUAGE MODELING USING HIDDEN UV STATES

A new approach to training language models and integrating acoustic word confidence scores was proposed in [6]. The integration of acoustic level confidence with the language model augments the word n-gram event space, which defines linguistic context, with encoded values of acoustic confidence $C = c_1, c_2, \ldots, c_K$. While a traditional n-gram language model computes the probability over all possible word sequences, W, via the back–off mechanism, the joint acoustic-language model has to account for the acoustic events as well. The joint probability of the sequence W, C is decomposed as:

$$P(W,C) = \prod_i P(w_i, c_i | w_1, c_1, w_2, c_2, \ldots, w_{i-1}, c_{i-1}) \tag{3}$$

In previous work, the probability estimates for the acoustic-language event space were based on the observed symbol pair training sequences [6]. Even though that model was able to back–off to lower order n-gram pairs, it was not able to recognize the pair (w_i, c_i) if c_i had never occurred in conjunction with w_i. In general when language modeling using multiple discrete information channels, the space of events is given by the Cartesian product of each channel event space. In order to assign a probability to all possible pairs of sequences W, C we have augmented the language model finite state space with Q hidden UV states for each word. As a result, any pair of sequences W, C will be recognized by our stochastic finite state machines and the probability $P(W, C)$ will be approximated from the training set pairs as:

$$\begin{aligned} P(w_i, c_i | w_1, c_1, w_2, c_2, \ldots, w_{i-1}, c_{i-1}) = \\ P(w_i | w_1, \ldots, w_{i-1}) \\ P(c_i | w_1, c_1, w_2, c_2, \ldots, w_{i-1}, c_{i-1}, w_i) \end{aligned} \tag{4}$$

In equation 4, the first term is the word n-gram and the second is trained using a Maximum Likelihood (ML) criterion from the training set word/UV-score pairs. Since some of the frequency counts in the ML estimate of $P(c_i | w_1, c_1, w_2, c_2, \ldots, w_{i-1}, c_{i-1}, w_i)$ will be zero, we applied standard discount techniques to the probability mass [4]. The approximation in Equation 4 is similar to the hidden variable modeling used for class-based language models. Thus, we have used a hierarchical training procedure for learning the stochastic finite state machine on the paired sequences W, C [4]:

- **Modeling Source W:** In this step a finite state model λ_L is designed so each string W is recognized and $P(W)$ is assigned to each W $(P(w_i | w_1, \ldots, w_{i-1}))$.

- **Training Hidden States for C:** Each state in the model λ_L is augmented with Q hidden states and discounted ML training is applied to compute the probability distribution of the hidden states.

At the end of the hierarchical training, the joint acoustic/language model λ_{L-UV} is represented as a stochastic transducer [6] and used for UV-labeled word lattice rescoring.

4 AUTOMATED CALL ROUTING TASK

The utterances used for the experimental study described in this paper were taken from spoken transactions between customers and human telephone operators over the public switched telephone network. The utterances correspond to customers responses to the open-ended prompt "How may I help you?" [1]. The first utterance from the customer in this transaction was transcribed and labeled according to one of fifteen call–types. The call–types themselves correspond to a set of actions relating to the routing of the incoming call. Examples of these call–types include *collect*, *calling card*, and *third party billing*, with an additional "*other*" type to handle calls that do not correspond to those that have been defined. A set of 670 utterances were used as a test set. The utterances were an average of 5.3 seconds in duration, with the longest utterance being 52.7 seconds. There is an average of nineteen words per utterance with an out–of–vocabulary rate in the test data at the utterance level of thirty percent. The baseline ASR performance on this test set is 60.7% word accuracy (WAC). When an n–best decoding algorithm is used with the number of candidates $N = 100$, the best possible WAC that can be obtained is 67.8%. This performance figure represents an upper bound on the possible performance that might be obtained by rescoring string candidates with some additional information source as was described in Section 2.

5 EXPERIMENTAL RESULTS

An experimental study was performed to evaluate the effect of the modified decoding procedure presented in Section 2 on the large vocabulary speech recognition (LVCSR) task described in Section 4. Two different aspects of the system were evaluated. First, the performance of the underlying word–based utterance verification procedures were measured. Second, the effect of rescoring the n–best hypothesized string candidates extracted from word lattices produced by the baseline speech recognizer using the ACC language model was evaluated. A value of $N = 100$ candidates was used for these experiments.

A modified word accuracy performance measure was used to characterize the effect of the ACC language model rescoring the test set. The measure exploits the fact that each word in the hypothesized string which is produced by the rescoring procedure is associated with some indication of confidence. The language model produces a discrete label, c_i, associated with each word, w_i, and the acoustic UV procedure associates a continuous valued confidence score, s_i, with each word. The performance measure characterizes the word accuracy (WAC) of the hypothesized word strings after "low confidence" words have been rejected. Low confidence implies that $c_i = 0$ when relying on discrete confidence labels for word-level confidence or $s_i < T$ where T is a decision threshold for continuous valued acoustic confidence. This will be referred to below as "thresholded" word accuracy, or TWAC.

The acoustic utterance verification performance for this task is plotted as a receiver operating characteristic (ROC) curve in Figure 3. The figure also displays the histograms of word level confidence scores obtained for the correctly decoded and incorrectly decoded words in the test set. The ROC curve in Figure 3 was generated by sweeping over a range of values of the decision

threshold, T. The vertical axis represents the probability of detecting a correctly decoded word hypothesis and the horizontal axis represents the probability of false acceptance of an incorrectly decoded word hypothesis. This curve was computed by assigning UV confidence measures to the words in the hypothesized strings generated for the one-best baseline ASR system described in Section 4. It is important to note that the characteristic curve is dependent on the operating point of the baseline speech recognition system.

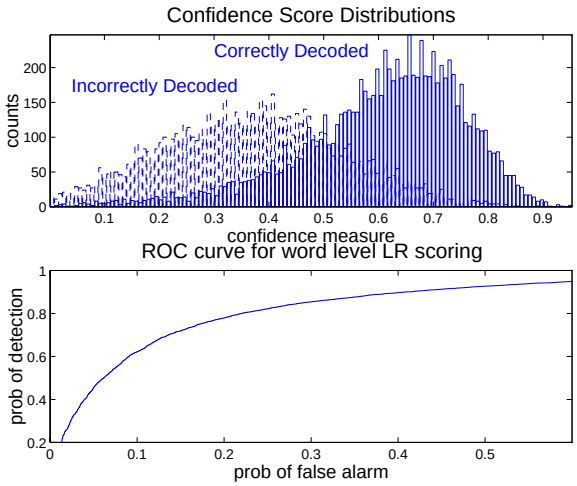

Figure 3: a. Histograms of word level confidence score distributions for correctly decoded and incorrectly decoded words. b. Receiver operating characteristic curve describing the detection performance of the word-level acoustic confidence measure.

Figure 4 displays the TWAC when rescoring 100 best string hypotheses for each utterance with the word level acoustic log likelihoods replaced by the word level acoustic confidence measures. The horizontal axis in Figure 4 corresponds to the percent of hypothesized words that were rejected, and the vertical axis represents the WAC for the accepted words. The baseline WAC corresponds to the TWAC=60.7% point on the vertical axis. Both of the curves in the figure were plotted for the case where the word level acoustic confidence measures were used as acoustic scores in the rescoring algorithm. The solid curve in Figure 4 displays the TWAC for rescoring of the acoustic confidence labeled string using the ACC language model. This scenario is equivalent to computing path probabilities, $\log P_{UV}(W, C)$, as shown in Section 2. The dashed curve in the figure displays the TWAC for rescoring of the acoustic confidence labeled string using the "baseline" language model which does not incorporate the coded confidence measures.

There are two observations that can be made from the plots in Figure 4. The first observation is that a small improvement over the baseline WAC was obtained by rescoring using the ACC language model. This is apparent from the 62.4% WAC which corresponds to zero percent word rejection in the figure. It was surprising that rescoring with the baseline language model resulted in a decrease in performance relative to the baseline. This may be a result of the fact that the average utterance length is almost 20 words and the fact that the upper bound on performance of 67.8% WAC is still very low. The second observation is that when a relatively low percentage of individual word hypotheses with low confidence scores are rejected, the resulting word

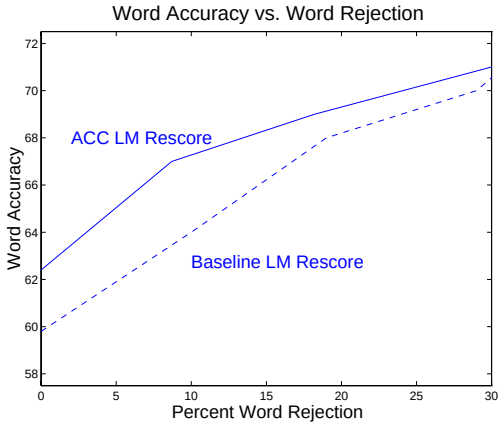

Figure 4: Threshold based word accuracy (TWAK) for rescoring of n–best UV labeled hypotheses.

strings containing high confidence words have a significantly increased WAC. This is an indication that the resulting confidence measures are relatively good predictors of whether words have been correctly decoded.

6 CONCLUSIONS

A decoding algorithm that incorporates representations of acoustic confidence in both the acoustic and language component of the decoder has been presented. The algorithm has been implemented as a procedure for rescoring n–best string candidates that are extracted from word lattices produced by a LVCSR system. The performance of the entire system was evaluated as the word accuracy obtained under various degrees of word rejection on an automated call routing task. It was found that best performance was obtained when the n–best rescoring mechanism relied on both word based acoustic confidence measures and acoustic confidence conditioned language models. Future work will focus both on better training procedures for the underlying acoustic and language models as well as on more efficient implementations for the integrated decoder algorithm described in this paper.

REFERENCES

[1] A.L.Gorin, G.Riccardi, and J.H.Wright. How may I help you? *Speech Communication*, 23:113–127, 1997.

[2] E. Lleida and R. C. Rose. Efficient decoding and training procedures for utterance verification in continuous speech recognition. *Proc. Int. Conf. on Acoust., Speech, and Sig. Processing*, May 1996.

[3] F. Pereira, M. Riley, and R. Sproat. Weighted rational transductions and their application to human language processing. *Proc. ARPA Workshop on Human Language Technology*, 1994.

[4] G. Riccardi, R. Pieraccini, and E. Bocchieri. Stochastic automata for language modeling. *Computer Speech and Language,*, pages 265–293, December 1996.

[5] R. C. Rose. Word spotting - extracting partial information from continuous speech utterances. In C. H. Lee, Frank K. Soong, and Kuldip K. Paliwal, editors, *Automatic Speech and Speaker Recognition - Advanced Topics*, pages 303–330. Kluwer, 1996.

[6] R. C. Rose, H. Yao, G. Riccardi, and J. Wright. Integration of utterance verification with statistical language. *Proc. ICASSP*, May 1998.