Word-Level Confidence Estimation for Automatic Speech Recognition

Andrew O. Hatch
ahatch@icsi.berkeley.edu

April 11, 2002

1 Introduction

The following master's thesis documents a 15-month research project at the International Computer Science Institute in the field of word-level recognition and confidence estimation for automatic speech recognition (ASR). Among the primary objectives of this research are 1) to derive an effective means of computing confidence scores for competing word-level hypotheses based on phoneme-level probability estimates and 2) to obtain more accurate word and sentence-level scores by combining scores from various ASR systems. Both of these efforts have focused on developing new techniques for processing and applying acoustic-based information extracted from hybrid ASR systems (i.e. systems that employ artificial neural networks (ANNs) in a hidden-Markov model (HMM) recognition system). The research described in this report is therefore primarily directed at the field of acoustic modeling. Some of the specific topics and themes covered include posterior vs. scaled-likelihood-based confidence estimation, computation of scaled likelihoods using adaptive vs. static state priors, soft-target vs. hard-target ANN training, and forward-backward reestimation of state posteriors.

Although the original goal of this research was to develop ASR technology for conversational speech, most of the early experiments described in this report focus on "noisy digits" tasks (i.e. digit strings spoken in the presence of various types of background noise). As described in section 3, these experiments address the issue of how to convert posterior state probability estimates (i.e. $p(q_i \mid x)$) where q_i is some phone state and x is the input acoustics) into scaled likelihoods of the form, $p(x \mid q_i)/p(x)$, or equivalently, $p(q_i \mid x)/p(q_i)$. Based on performance tests of confidence scores computed from scaled likelihood estimates, the experiments demonstrate the utility of adapting state priors (i.e. estimates of $p(q_i)$) to the test data of the current speaker and the current noise environment. The experiments of section 3 also point to the superiority of scaled-likelihood-based confidence measures over confidence measures based solely on state posteriors.

These early results on digit-strings helped lay the groundwork for subsequent experiments on conversational speech, which comprise the bulk of this thesis. One of the more interesting early results on conversational speech demonstrates the utility of using forward-backward recursions to reestimate local state probability estimates prior to computing word and sentence-level scores (this topic is discussed in detail in section 4).

In accordance with the stated goals of this research, various realizations of the ICSI system were later integrated with a likelihood-based system provided courtesy of SRI International. These experiments ultimately produced an integrated ASR system that achieved a word error rate of 33.7% on the Switchboard task. Note that these test results constitute an absolute improvement of approximately 0.7% over the best performance recorded by either system in isolation. Further information on these experiments may be found in section 5.

2 The ICSI Speech Recognition System

In most contemporary ASR systems, the process of recognizing speech can be broken down into the following three stages:

1. feature extraction
2. phone classification
3. decoding

2.1 Feature Extraction

The purpose of the first stage is to extract a set of *features* from the speech signal that capture information about the signal's phonetic content. Typically, these features are computed in the form of n-dimensional vectors, where each vector corresponds to a given frame (i.e. a short-time, fixed-length segment of the speech signal). In the ideal case, these feature vectors provide a compact, coherent representation of all of the relevant speech information contained in the signal, while filtering out non-speech information. Feature extraction may thus be viewed as a form of signal compression, where the interest is in retaining speech information, not in maintaining the audio quality of the original signal.

The feature sets used in this research were obtained by means of an analysis technique called "RASTA Perceptual Linear Prediction" (i.e. RASTA-PLP). As with most feature extraction paradigms in speech recognition, RASTA-PLP involves applying short-time cepstral analysis to the incoming speech signal. The technique also involves various filtering steps. Further details on RASTA-PLP can be found in [2].

2.2 Phone Classification

After extracting a set of n features for each frame of speech data, the recognition process moves on to the phone classification stage, where the input acoustics

are either mapped into acoustic state *likelihoods* (i.e. estimates of $p(x \mid q_i)$ or state *posteriors* of the form, $p(q_i \mid x)$. In the former case, a gaussian mixture model (GMM) system is typically used to produce estimates of the state likelihoods. These likelihoods are then applied to an HMM-based decoding system which determines the most probable word hypotheses for a given utterance. Alternatively, some ASR systems employ discriminative models such as artificial neural networks (ANNs) to compute estimates of $p(q_i \mid x)$. These estimates are then converted into "scaled likelihoods" of the form $p(q_i \mid x)/p(q)$ prior to being processed in a similar manner by an HMM-based decoder.

As indicated earlier, the ICSI system uses an ANN to establish a mapping between input feature vectors and estimates of the posterior probabilities for each state. Since these probability estimates are "conditioned" on the input features, we typically refer to the outputs of the ANN as *posterior state probabilities* or simply as *posteriors*. For this report, we will use the notation, $p(q_k^n \mid x^n)$, to represent the posterior probability of state q_k occuring at time n, conditioned on the input acoustics (i.e. the input feature vector x at frame n).

Note that the various states, $q_1, q_2, \ldots, q_M$, defined by the ICSI acoustic model represent linguistic units called *phones*.[1] These phones correspond to particular sub-word speech sounds (e.g. /k/, /ae/, and /t/ as in the word, "cat"). A total of 56 phone classes are defined by the ICSI system, although not every phone is used in every speech task. Most digit-string tasks, for instance, require only 26 phones.

The architecture used for all ANN estimators described in this thesis consists of three layers of nodes: an input layer, a hidden layer, and an output layer. The input layer accepts a total of $CW \times M$ input values, where M is the length of each feature vector, and CW is the given "context window" (i.e. the number of consecutive feature vectors applied to the ANN). These input values are mapped through a fully-connected system of non-linearities to the hidden layer, which is similarly mapped to 56 output nodes representing the output posteriors for each state. At the hidden and output layers, the value of a given node, j, is computed as $f(w_{0j} + \sum_i w_{ij} x_{ij})$, where $x_{1j}, x_{2j}, \ldots, x_{Nj}$ represents the set of all nodes that are connected to the input of j and $w_{0j}, w_{1j}, w_{2j}, \ldots, w_{Nj}$ represents a corresponding set of weights. A smooth nonlinearity such as a sigmoid or softmax function is typically chosen for $f(\cdot)$. To train the network, an error criterion (e.g. the mean squared error of the output posteriors with respect to the training labels) is differentiated with respect to the internal weights, and the weights are updated accordingly. After each update, the *frame accuracy* of the ANN is tested on a held-out corpus of *cross-validation* data to prevent overtraining. This simply involves determining the percentage of the phone labels that match the top ranked phone for each frame (as determined by the outputs of the ANN). Once the gains in frame accuracy begin to level off or decrease, the step-size of each weight update is reduced until the training process is ultimately stopped. Further details on the training and design of ANNs can be found in [4] and [3].

[1] Many speech recognition systems define multiple states for each phone.

2.3 Decoding

Once the input features have been mapped into posterior probabilities by the ANN, the recognition process moves on to the decoding stage, where the input speech is matched with various possible word hypotheses. In most ASR systems, the decoding process is performed by using a set of hidden Markov models (HMMs) to determine the likelihoods of various word and sentence-level hypotheses given the outputs of the acoustic model. The purpose of these HMMs is to model the a priori probabilities of state transitions through an utterance based on observations made from actual speech. For instance, an HMM might be trained to encode the probability that phone state /ae/ transitions to /t/, conditioned on the duration of /ae/ and on the fact that the previous phone state was /b/. Given a sufficiently rich set of transition probabilities, an HMM may be used to estimate a priori probabilities of the form $p(Q \mid M_h)$, where $Q = q_j^n, q_k^{n+1}, \ldots, q_l^N$ represents an entire state sequence or "path" through an utterance, and M_h represents the given HMM. These a priori path likelihoods are then used in conjunction with other statistics, including language models and the outputs of the acoustic model to arrive at the probability $p(W \mid X, M)$ of a word sequence W given the input acoustics, X, and the overall ASR system, M. Many of the specifics of this procedure will be revealed in the coming sections—many others, however, are beyond the scope of this research, and will only be mentioned in passing. For now, we will limit the discussion by simply assuming that a means exists for estimating $p(W \mid X, M)$ for any word sequence W. For further details on decoding, the reader is referred to [6] and [1].

Note that in practice, it may not always be possible to conduct an exhaustive search of the word hypothesis space, particularly for unconstrained, large-vocabulary speech tasks where there are thousands of competing sentence-level hypotheses. For large tasks such as these, a number of pruning techniques are often employed to limit the search process (more details on search may be found in [6] and [1]).

2.4 Additional Background: The Forward and Backward Recursions

In this section, we introduce some basic theory related to 1st-order hidden Markov models. Assuming that state transitions in a given model follow a Markov process (i.e. a process where $p(q_{k_{n+1}}^{n+1} \mid q_{k_n}^n, q_{k_{n-1}}^{n-1}, \ldots, q_{k_{n-N}}^{n-N}) = p(q_{k_{n+1}}^{n+1} \mid q_{k_n}^n)$ for any N), we may define the following two recursions:

$$\begin{aligned} \alpha_n(\ell) &= p(X_1^n, q_\ell^n) \\ &= \left[\sum_k \alpha_{n-1}(k) p(q_\ell^n \mid q_k^{n-1}) \right] p(x^n \mid q_\ell^n) \end{aligned}$$

and,

$$\beta_n(\ell) = p(X_{n+1}^N \mid q_\ell^n, X_1^n, M)$$

$$= \sum_k \beta_{n+1}(k) p(q_k^{n+1} \mid q_\ell^n) p(x^{n+1} \mid q_k^{n+1})$$

Here, X_n^N represents the set of all observations from frame n to frame N. The above equations define the basic statistical properties of an HMM system. Given a set of local state likelihood estimates (i.e. $p(x^n \mid q_i^n)$) and a corresponding set of a priori state transition probabilities (i.e. $p(q_\ell^n \mid q_k^{n-1})$), we can efficiently distribute information throughout a given model by means of a *forward* recursion (represented by the α terms shown above), and an analogous *backward* recursion which is represented by β. The α and β terms may also be combined to obtain "global" posterior estimates of the form, $p(q_k^n \mid X_1^N)$, as shown in the following equation (note that the term, "forward-backward posterior" is also used throughout this thesis to describe $p(q_k^n \mid X_1^N)$).

$$\begin{aligned} p(q_k^n \mid X_1^N) &= \frac{p(X_1^N, q_k^n)}{p(X_1^N)} \\ &= \frac{\alpha_n(k)\beta_n(k)}{\sum_\ell \alpha_n(\ell)\beta_n(\ell)} \end{aligned}$$

Since the computation of $p(q_k^n \mid X_1^N)$ is invariant to any scaling at the frame-level, we note that the $p(x \mid q_i)$ terms in the forward and backward recursions may be replaced with scaled likelihoods of the form $p(q_i \mid x)/p(q_i)$. Thus, the forward and backward recursions allow us to convert local posterior estimates obtained from an ANN into global posteriors of the form $p(q_i^n \mid X_1^N)$. This approach to reestimating state posteriors forms the basis for much of the research described in the following sections.

3 Experiments on the Aurora Noisy Digits Task

As stated in the preceding introduction, many of the early experiments in this project were performed on the Aurora corpus—a speech task consisting of digit-strings spoken in the presence of noise. The purpose of these experiments was to compare different methods of applying state posteriors computed by the ANN to the task of estimating word-level confidence. More specifically, the experiments investigated different methods of estimating state priors (i.e. $p(q_i)$) for the purposes of converting posteriors into scaled likelihoods (i.e. $p(q_i \mid x)/p(q_i)$). An existing model for estimating acoustic-based confidence was then used to transform these scaled likelihoods into word-level confidence scores, which were later tested and compared.

The methodology used in the Aurora experiments for computing acoustic-based confidence measures follows directly from the work of Williams *et al.* in [7]. Given an ANN-based recognition system that produces estimates of $p(q_k^n \mid x^n)$, where q_k^n represents the occupation of state (i.e. phone) k at time n and x^n represents the corresponding input acoustics, Williams *et al.* define the

following phone-level acoustic confidence measure:

$$nPP(q_k) = \frac{1}{D} \sum_{n=n_s}^{n_e} \log(p(q_k^n \mid x^n)) \tag{1}$$

Here, n_s and n_e denote the start and end times of phone q_k according to a particular word-level or phone-level hypothesis, and D represents the hypothesized phone duration. The above expression thus computes the logarithm of the *normalized posterior probability* (i.e. $nPP(\cdot)$) of phone q_k given a hypothesized start and end time.

To obtain confidence scores at the word-level, the phone-level confidence scores for each phone in a word hypothesis may be combined according to

$$nPP(w_k) = \frac{1}{L} \sum_{l=1}^{L} nPP(q_{k_l}), \tag{2}$$

where L denotes the number of phones in word w_k and $q_{k_1}, q_{k_2}, \ldots, q_{k_L}$ denotes the hypothesized phone sequence. To justify the basic form of this model, we appeal to the intuition that the sum of a group of N consecutive log posteriors in a phone or word-level hypothesis will tend to bear a direct relationship to the correctness of the given hypothesis. We should note, however, that this intuition assumes that the posteriors themselves provide reasonably informative estimates of the relative correctness of a given frame—that is, we assume that the posteriors provide meaningful probability estimates. One caveat to this approach, however, is that the sum of a sequence of log posteriors decreases with increasing N (note that this results from the fact that each log posterior is negative). Thus, to allow for comparisons between hypotheses of differing lengths, the resulting confidence measures must be normalized with respect to duration, as is done in equations 1 and 2.

3.0.1 The Aurora Corpus

As with most speech corpora, the Aurora corpus is divided into 2 data sets: one for training and one for testing, each recorded by a disjoint group of speakers. Both data sets consist of strings of 1 to 7 randomly-selected digits read in the presence of 1 of 4 different noise types: hall, babble, car, and train. The recording conditions for the data sets are further categorized by 1 of 7 noise levels including "clean" and every decibel level ranging from 20 to -5 (in decrements of 5 dB). Thus, a total of 28 different noise types and levels are represented in the Aurora corpus. Each of the 28 noise type/level subcategories accounts for 1001 test utterances and 300 training utterances—the complete training and test sets therefore comprise a total 8440 and 28028 utterances, respectively.

3.0.2 Acoustic Model Training

To train an acoustic model for the Aurora task, a set of 9 RASTA-PLP features were computed for each 16 millisecond frame of training data. An ANN system

	WER (%)			
experiment	*Hall*	*Babble*	*Train*	*Car*
Clean	3.7	3.9	3.1	3.4
SNR20	5.0	5.7	4.1	4.6
SNR15	7.7	10.0	5.5	6.0
SNR10	15.8	24.2	11.0	9.0
SNR5	43.8	51.7	25.4	29.2
SNR0	86.7	75.7	56.0	65.9
SNR-5	93.0	88.7	86.7	87.0

Table 1: Recognition Results for the Aurora Noisy Digits Task

consisting of 81 input units, 480 hidden units, and 56 output units was then trained to the corresponding phone labels (i.e. hard targets) of a random ordering of the 1st 7640 utterances of the training set. The remaining 800 utterances were reserved to provide a cross-validation corpus for the ANN training process.

Using the resulting ANN system, state posterior estimates were computed for each frame of the test corpus. A decoding was then performed on the full test-set, yielding a single "best-guess" hypothesis and corresponding phone-level time-alignment for each utterance. The word-error rates (WERs) computed for each of the test sets are summarized in table 1. Note that the average WER over 20 test sets (all test sets with the exception of "clean" and -5 dB) for this system is 27.2%.[2]

3.0.3 Experimental Procedure

Given the decoded list of word-hypotheses, the posterior estimates of the ANN, and estimated phone alignments corresponding to each word hypothesis, the formulae given in equations 1 and 2 were used to compute confidence scores for each word hypothesis. The effectiveness of the resulting confidence scores was then evaluated by dividing the word hypotheses into two groups: correct words and incorrect words, and then computing a detection error tradeoff (DET) curve as shown in figure 1. The DET curve simply provides an indication of how separable the correct and incorrect hypotheses are based on their confidence scores. To resolve a single point on the DET curve, we set an arbitrary threshold τ and assume that every word having a confidence score greater than τ will be "retained," while every word having a confidence score less than τ will be "rejected." Thus, for every threshold τ, we can compute percentages for the two types of errors encountered in confidence estimation—the percentage of correct words rejected and the percentage of incorrect words retained. If we vary the threshold over the entire range of confidence scores, we can form a complete DET curve like the one shown in figure 1. In the field of confidence estimation, the hope is, of course, that the percentages of correct words rejected

[2]This is considerably worse than the state-of-the-art for the Aurora task—in [5], Sharma *et al* report a system that performs at an average WER of roughly 6%.

and incorrect words retained will tend to be low for most points on the DET curve. Thus, given the axes shown in figure 1, the better-performing confidence measures will have DET curves that are closer to the lower left-hand corner of the figure where the percentages of errors approach zero.

To provide a quantitative means of comparing different confidence measures, it is often useful to evaluate error rates at specific points on a DET curve rather than trying to interpret an entire graph. For this thesis, we will be particularly concerned with the equal-error rates (EERs) of DET curves. The EER of a DET curve is simply defined as the error rate at which the percentage of correct words rejected is equal to the percentage of incorrect words retained.

3.0.4 Model Derivation

In the process of experimenting with various methods of computing confidence scores, it was discovered that a significant reduction in equal-error rate (EER) could be achieved by replacing the posterior terms (i.e. $p(q_k^n \mid x^n)$) in equation 1 with scaled likelihoods of the form $\frac{1}{\alpha_n} \cdot p(q_k^n \mid x^n)/p(q_k)$. Here, the constant α_n normalizes all of the scaled likelihoods within frame n to sum to one. Equations 3 and 4 summarize the modified confidence model:

$$nPP(q_k) = \frac{1}{D} \sum_{n=n_s}^{n_e} \log \left(\frac{1}{\alpha_n} \cdot \frac{p(q_k^n \mid x^n)}{p(q_k)} \right) \tag{3}$$

$$\alpha_n = \sum_k \frac{p(q_k^n \mid x^n)}{p(q_k)} \tag{4}$$

We can build a theoretical basis for this model by referring to the HMM theory developed in section 2.4. Let us first assume that the prior probability $p(Q)$ of any path $Q = q_{k_1}^n, q_{k_2}^{n+1}, \ldots, q_{k_N}^N$ within an M-state model is equal to ψ^N, where $\psi = 1/M$. Thus, all state priors, $p(q_k)$, and all state transition probabilities, $p(q_{k_{n+1}}^{n+1} \mid q_{k_n}^n)$, are uniform, and the prior probability of any path Q is dependent only on its duration, N. Based on this assumption, let us now compute the joint probability, $p(Q_k^{n_s,n_e}, X^{n_s,n_e})$, where $Q_k^{n_s,n_e} = q_{k_n}^{n_s}, q_{k_n}^{n_s+1}, \ldots, q_{k_n}^{n_e}$ represents a given state sequence and X^{n_s,n_e} represents the input acoustics from frame n_s to frame n_e.

$$\begin{aligned} p(Q_k^{n_s,n_e}, X^{n_s,n_e}) &= p(Q_k^{n_s,n_e}) \cdot p(X^{n_s,n_e} \mid Q_k^{n_s,n_e}) \\ &= \psi^{n_e-n_s+1} \cdot \prod_{n=n_s}^{n_e} p(x^n \mid q_{k_n}^n) \\ &= k \cdot \psi^{n_e-n_s+1} \cdot \prod_{n=n_s}^{n_e} \frac{1}{\alpha_n} \cdot \frac{p(q_{k_n}^n \mid x^n)}{p(q_{k_n})} \end{aligned}$$

Here, k simply accounts for the scaling of the likelihood terms. Next, we compute $p(X^{n_s,n_e})$ by summing $p(Q_k^{n_s,n_e}, X^{n_s,n_e})$ over all paths that start at n_s

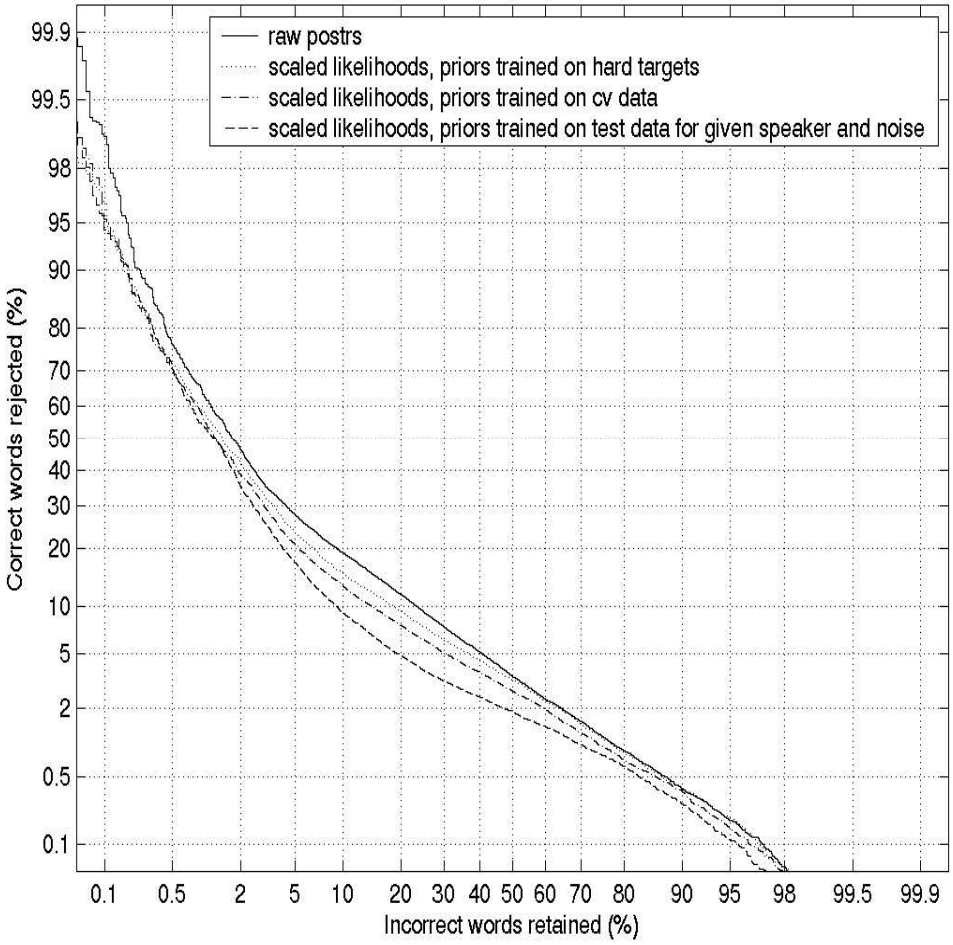

Figure 1: DET Curves for the Aurora Noisy Digits Task (complete test set)

and end at n_e:

$$\begin{aligned} p(X^{n_s,n_e}) &= \sum_{Q_k^{n_s,n_e}} p(Q_k^{n_s,n_e}, X^{n_s,n_e}) \\ &= k \cdot \psi^{n_e-n_s+1} \cdot \sum_{Q_k^{n_s,n_e}} \prod_{n=n_s}^{n_e} \frac{1}{\alpha_n} \cdot \frac{p(q_{k_n}^n \mid x^n)}{p(q_{k_n})} \end{aligned}$$

Note that the terms at the right side of the above equation sum to 1. Thus, $p(X^{n_s,n_e})$ reduces to

$$p(X^{n_s,n_e}) = k \cdot \psi^{n_e-n_s+1}$$

The log of $p(Q_k^{n_s,n_e} \mid X^{n_s,n_e})$ for a particular phone-level hypothesis $Q_k = q_k^{n_s}, q_k^{n_s+1}, \ldots, q_k^{n_e}$ may then be computed as

$$\begin{aligned} \log p(Q_k^{n_s,n_e} \mid X^{n_s,n_e}) &= \log p(Q_k^{n_s,n_e}, X^{n_s,n_e}) - \log p(X^{n_s,n_e}) \\ &= \sum_{n=n_s}^{n_e} \log \left(\frac{1}{\alpha_n} \cdot \frac{p(q_k^n \mid x^n)}{p(q_k)} \right) \end{aligned}$$

At this point, we've computed the desired log posterior, $\log p(Q_k^{n_s,n_e} \mid X^{n_s,n_e})$, which we use as a basis for computing confidence scores for phone-level hypotheses. However, since the value of $\log p(Q_k^{n_s,n_e} \mid X^{n_s,n_e})$ will tend to be lower for longer hypotheses, we are still left with the problem of maintaining comparability between confidence scores for phone hypotheses of different lengths. To correct for biases introduced by differences in duration, we simply normalize $\log p(Q_k^{n_s,n_e} \mid X^{n_s,n_e})$ with respect to D:

$$nPP(q_k) = \frac{1}{D} \cdot \log p(Q_k^{n_s,n_e} \mid X^{n_s,n_e}) = \frac{1}{D} \sum_{n=n_s}^{n_e} \log \left(\frac{1}{\alpha_n} \cdot \frac{p(q_k^n \mid x^n)}{p(q_k)} \right)$$

Note that this gives us the same expression shown in equation 3. Thus, for the case where all state priors and all state transition probabilities are assumed to be uniform, we have shown that the phone-level confidence model of equations 3 and 4 (prior to duration normalization) follows directly from the framework of HMM theory.

A comparison between the performance of the original and the modified confidence estimation models is provided in figure 1. Here, the baseline DET curve corresponding to the original posterior-based confidence model of equation 1 is shown under the "raw postrs" label. The other curves in the figure correspond to confidence measures obtained from the scaled-likelihood model of equation 3. Each of these curves employs a different estimate of the state prior, $p(q_k)$, for the computation of word-level confidence. The estimate of $p(q_k)$ corresponding to the "scaled likelihoods, priors trained on hard targets" curve is simply computed as the average frequency of each phone class based on the hard targets of

the training data. Thus, we compute $p(q_k)$ as

$$p(q_k) = \frac{1}{N} \sum_{n=1}^{N} \mathbf{1}(q_k^n) \tag{5}$$

where $\mathbf{1}(q_k^n)$ is either one or zero, depending on whether or not state q_k is occupied at time n. This "hard targets" curve is particularly significant because it employs the method by which priors are typically computed in ANN-based speech recognition systems. The most compelling justification for this method of estimating priors has to do with the fact that virtually every aspect of embedded ASR training may ultimately be traced back to the original phone labels assigned to the training data. Hence, it is often argued that the hard labels provide the most appropriate source of "ground truth" for estimating state priors.

On the other hand, one might argue that if a posterior estimator, $p(q_k^n \mid x^n)$ is assumed to be accurate, then $p(q_k)$ should be consistent with that estimator. Based on this reasoning, we could try computing $p(q_k)$ as the time-average of $p(q_k^n \mid x^n)$ over some set of N frames, as shown below:

$$p(q_k) = \frac{1}{N} \sum_{n=1}^{N} p(q_k^n \mid x^n) \tag{6}$$

To provide a theoretical basis for this model, we can begin by expressing $p(q_k)$ as a marginal probability:

$$p(q_k) = \int_x p(q_k \mid x) p(x) dx \tag{7}$$

Since $p(q_k \mid x)$ is a function of the random variable, x, equation 7 simply states that $p(q_k) = E[p(q_k \mid x)]$. Thus, a natural estimate for $p(q_k)$ can be obtained by time-averaging $p(q_k \mid x)$, as long as the group of posteriors over which we are averaging is adequately representative of the true distribution of $p(q_k)$. In this paper, we will not concern ourselves with formally testing whether or not the latter condition is satisfied for any specific group of posteriors. However, we acknowledge that equation 6 should only be used in cases where the group of sample posteriors can reasonably by assumed to represent $p(q_k)$.

This alternative formulation introduces additional questions about which data set(s) to use for averaging state posteriors. One possible source for the posteriors in equation 6 is the cross validation data. This is the data source used to train priors for the "scaled likelihoods, priors trained on cv data" curve in figure 1. We note, however, that equation 6 also allows for the adaptation of $p(q_k)$ to posteriors taken from the test data itself. In the Aurora corpus, the test sets are divided into 28 subsets—4 noise types multiplied by 7 noise levels. Thus, it's natural to train priors on posteriors that match the given noise conditions. Given that we are also provided knowledge of speaker identities, the Aurora task allows for the training of a set of priors, $p(q_{k,s,n,\ell})$, on posteriors obtained from the test data of a particular speaker, s, a given noise type, n,

experiment	*EER (%)*
raw postrs	14.86
scaled likelihoods, priors trained on hard targets	12.90
scaled likelihoods, priors trained on cv data	11.64
scaled likelihoods, priors trained on test data for given speaker and noise	9.60

Table 2: Confidence Estimation Results for the Aurora Noisy Digits Task

and a given noise level, ℓ. When we apply the resulting priors, $p(q_{k,s,n,\ell})$, to equation 6 to compute confidence scores for speaker, s, under noise conditions, n and ℓ, we obtain the DET curve labeled "scaled likelihoods, priors trained on test data for given speaker and noise" in figure 1.

3.0.5 Results

The corresponding EER for each of the four curves is shown in table 2. As shown in figure 1 and table 2, all three of the curves computed from the modified confidence model of equations 3 and 4 outperform the curve computed from the original posterior-based model. Using the "raw postrs" curve as a baseline, the scaled likelihood-based curves yield relative reductions in EER of 13.19%, 21.67%, and 35.40%, respectively.

As for the performance of the various estimates of $p(q_k)$ used in the experiment, figure 1 and table 2 show that the cross-validation priors outclass the hard target priors by a margin of 9.77%, in terms of relative EER. The best showing, however, is put forth by the adaptive priors which outperform the priors obtained from hard targets and cross validation data by relative margins of 25.58% and 17.53%, respectively. These results lend credence to the estimation paradigm of equation 6, where state priors are computed as the time average of posterior estimates obtained from the ANN. Moreover, the results point to the utility of adapting priors to the particular conditions of the input speech, including adapting to the current speaker and the current noise environment.

To put these results in the proper context, we should note, however, that real world speech tasks do not always allow us to make assumptions about the identity of the current speaker or noise environment. Thus, the results obtained from the adaptive priors for the Aurora task might be viewed as adaptation under ideal (or at least favorable) conditions. For the purposes of implementing real world adaptive ASR systems, it's encouraging to note, however, that the adaptive priors were trained on relatively small lengths of test data. In the Aurora test set, only 9 or 10 utterances are recorded for each speaker in each of the 28 noise conditions. Given that these utterances tend to be no longer than 6 seconds in length, we might conclude that for limited vocabulary tasks like Aurora, effective adaptation of prior state probabilities may be possible on time scales of no longer than one minute.

4 Experiments on the Switchboard Task

Following the Aurora experiments, a set of tests were conducted on the Switchboard corpus, which consists of conversational speech recorded over telephone lines. The primary objective of these experiments was to apply the forward-backward recursion outlined in section 2.4 to the task of reestimating the raw state posteriors produced by the ANN. To recap the procedure, the forward-backward recursion uses prior state transition probabilities of the form $p(q_k^{n+1} \mid q_j^n)$ in conjunction with scaled likelihoods (i.e. $p(x^n \mid q_k^n)$) to compute path probabilities through an utterance. By summing the likelihoods of all paths that arrive at a given state, q_k, at a given time, n, the recursion produces estimates of the so-called "global" or "forward-backward" posterior, $p(q_k^n \mid X, M)$. Normally, forward-backward posteriors are only used to determine soft time-alignments for the purposes of embedded training (more on these topics can be found in [6] and [1]). However, one might imagine that a posterior of the form $p(q_k^n \mid X, M)$ might be useful in other applications as well, such as the estimation of word-level confidence. Indeed, it's worth noting that forward-backward posteriors are "conditioned" on more information than raw posterior estimates (i.e. $p(q_k^n \mid x^n)$)—thus, it seems reasonable to assume that confidence estimates computed from forward-backward posteriors may yield improved performance over estimates computed directly from the outputs of the ANN. In the following report, we describe several experiments that support this intuition.

4.0.6 HMM Specifications

To build an HMM architecture for the forward-backward procedure, we trained a set of first-order phone transition probabilities from the pronunciations and unigram probabilities of each word contained in a 32000 word lexicon. Transition probabilities derived in this way may be represented as $p(q_k^n \mid q_\ell^{n-1}, \tilde{q}_\ell^n)$ where $\tilde{q}_\ell^n$ denotes the event of q_ℓ *not* being occupied at time n (in other words, $(q_\ell^{n-1}, \tilde{q}_\ell^n)$ denotes an exit out of state q_ℓ at time n). We also trained duration models for each phone from the labels of the training data. Each duration model consists of a simple left-to-right architecture of N states, $q_{k_1}, q_{k_2}, \ldots, q_{k_N}$, where if $n < N$, q_{k_n} can either exit or transition to the next state, $q_{k_{n+1}}$. Only the last state, q_{k_N}, is permitted to have a non-zero self-loop probability. Given this architecture, the duration models are trained by associating q_{k_n} where $n < N$ with the nth consecutive frame of phone q_k in a forced Viterbi alignment. The final state, q_{k_N}, is associated with each consecutive frame of phone q_k in the forced alignment that follows q_{k_N}. Using this procedure, any distribution of phone durations derived from the cross-validation data can be precisely modelled given that N is sufficiently large.

Once derived, the duration models and first-order phone transitions were merged into a single HMM by applying Bayes rule:

$$p(q_{k_1}^n \mid q_{\ell_m}^{n-1}) = p(q_k^n \mid q_\ell^{n-1}, \tilde{q}_\ell^n) \times p(\tilde{q}_\ell^n \mid q_{\ell_m}^{n-1})$$

Here, the two terms on the right-hand side of the equation are obtained from

the phone transition models and from the phone duration models, respectively.

For the switchboard experiments, the number of substates per phone in each duration model was set to $N = 5$. We also defined two parameters, ϵ and ρ, that were used to smooth the probabilities of each state transition defined by the HMM:

$$\hat{p}(q_{k_1}^n \mid q_{\ell_m}^{n-1}) = (p(q_{k_1}^n \mid q_{\ell_m}^{n-1}) + \epsilon)^\rho \tag{8}$$

Here, $\hat{p}$ represents a probability after smoothing is applied. Since these smoothed values do not necessarily retain the usual properties of probability estimates (e.g. they do not necessarily sum to one), we will refer to quantities of the form, $\hat{p}(q_{k_1}^n \mid q_{\ell_m}^{n-1})$, as transition *likelihoods*. Although these transition likelihoods do not form a consistent set of conditional probabilities, we may still use the $\hat{p}(q_{k_1}^n \mid q_{\ell_m}^{n-1})$ terms to estimate path probabilties by renormalizing the likelihoods of all possible paths, $Q_1, Q_2, \ldots, Q_N$, so that $\sum_i p(Q_i) = 1$. Note, however, that the forward-backward recursion performs this renormalization implicitly—thus, no further consideration must be given to computing path probabilties from the $\hat{p}(q_{k_1}^n \mid q_{\ell_m}^{n-1})$ terms.

Given the model described above for the transition likelihoods of the HMM system, the forward-backward recursion was performed on state posteriors to produce estimates of $p(q_k^n \mid X_1^N, M)$. These forward-backward posteriors were then substituted into the confidence models of section 3 to compute confidence measures for a corresponding set of word-level hypotheses. For each of the Switchboard experiments, these hypotheses were obtained from an independent ASR system provided by SRI International. Note that the WER of the SRI system was measured to be 29.71% on the final Switchboard test corpus. Prior to actually testing the performance of confidence measures derived from forward-backward posteriors, the smoothing parameters, ϵ and ρ, were optimized on a held-out corpus of 1143 cross-validation utterances. For each confidence model tested in this section, the optimal values of the smoothing parameters were found to be approximately the following: $\epsilon = 0.01$ and $\rho = 0.55$. Note, however, that smoothing was only applied to state transitions that are permitted by the architecture of the duration model (i.e. no smoothing was applied to undefined transitions, such as $p(q_{k_4}^n \mid q_{\ell_3}^{n-1})$). Thus, the probability of all undefined state transitions remains zero in the smoothed probability model.

4.0.7 System Training and Experimental Setup

The ANN system used in the forward-backward experiments consists of two separate ANNs—one trained on 68903 utterances of female speech and the other trained on 57067 utterances of male speech. Both the male and the female ANN were trained to hard targets using 13 PLP features computed for every 10 millisecond frame of speech data. A context window of 17 frames was employed on the nets—thus, a total of $17 \times 13 = 221$ input units were used for training. Both of the ANN systems were implemented with a hidden layer of 8000 units and an output layer of 56 units (one for every phone state).

To compute frame-level posteriors, the male and the female ANNs were used individually to produce 2 sets of posterior estimates for each utterance. A simple

form of gender detection was then performed by comparing the relative entropies of the resulting male and female posteriors (more on this subject can be found in [6]). The gender having the lower of the two entropy statistics was then chosen as the gender hypothesis for the given utterance, and the corresponding posteriors were retained for further use. Thus, if a set of male posteriors for a given utterance have a lower entropy than a set female posteriors for the same utterance, only the male posteriors will be used for the purposes of computing confidence scores, performing recognition, etc.

As for the speech corpora used in these experiments, the Switchboard Devtest 2000 set was used to tune the ϵ and ρ parameters listed above, while the Switchboard Eval2000 set was used for testing. These corpora consist of 1143 and 4466 utterances, respectively, each comprised of between 100 and 2000 frames of conversational speech recorded by a single speaker. As with the Aurora task, each utterance is indexed with a label corresponding to the current speaker's identity. Thus, the Switchboard task allows for the computation of speaker-specific priors of the form, $p(q_{k,s})$, where k is the state label and s is the current speaker.

4.0.8 Results

Figure 2 shows DET curves corresponding to various models of word-level confidence computed from either raw posteriors or forward-backward posteriors. The various curves are computed from either the posterior or the scaled-likelihood based models of equations 1 and 3, and are labeled accordingly. As implied by the legend of figure 2, the curves are further distinguished by the prior estimates used in the computation of the forward-backward posteriors. In all cases, the priors are either estimated from the raw posteriors or from the forward-backward posteriors of the test data for the given speaker. Thus, each of the prior estimates is of the form $p(q_{k,s})$, where k is the state label and s is the current speaker. A fourth distinction between the curves involves the particular forced alignment procedure used to compute the optimal path of each word and phone-level hypothesis. For each of the curves, the state paths are computed from a forced Viterbi alignment using one of two possible estimates of the scaled likelihoods, $p(q_k^n \mid x^n)/p(q_k)$ (more information on forced Viterbi alignment can be found in [6] and [1]). In the "align I" case, these scaled likelihoods are computed from raw posteriors, while forward-backward posteriors are used in "align II." Note that both align I and the align II employ priors estimated from the hard-targets of the training data, as in equation 5.

The corresponding EER for each DET curve is listed in table 3. As expected, the results in table 3 obtained from the scaled-likelihood-based confidence model of equation 3 outperform those obtained from the posterior-based model of equation 1 (when holding constant for alignment). These results agree with the findings of the Aurora experiments and further support the effectiveness of confidence models based on scaled-likelihoods. The EERs of table 3 also suggest that raw posteriors are a better source for computing priors than forward-backward posteriors. This is an interesting result, since the best overall

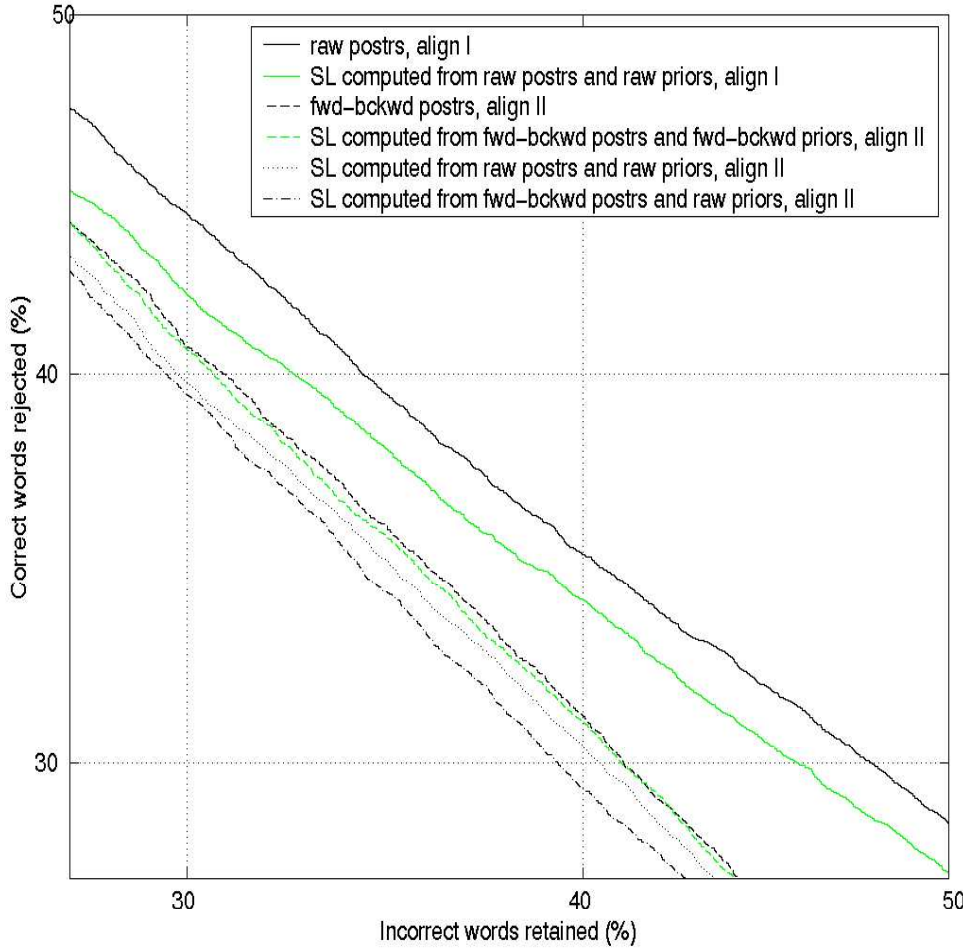

Figure 2: DET Curves for the Switchboard Task. Here, "SL" represents "scaled likelihoods." "align I" and "align II" refer to alignments computed from two different estimates of the scaled likelihoods. In the "align I" case, the scaled likelihoods are computed from raw posteriors while forward-backward posteriors are used for "align II" (note that both sets of scaled likelihoods are computed by dividing the given posteriors by priors obtained from the hard-targets of the training data). "raw priors" refers to adaptive priors computed from the raw posteriors, and "fwd-bckwd priors" refers to adaptive priors computed from the forward-backward posteriors.

experiment	*EER (%)*
raw postrs, align I	37.52
SL computed from raw postrs and raw priors, align I	36.64
fwd-bckwd postrs, align II	35.57
SL computed from fwd-bckwd postrs and fwd-bckwd priors, align II	35.47
SL computed from raw postrs and raw priors, align II	35.14
SL computed from fwd-bckwd postrs and raw priors, align II	34.68

Table 3: Confidence Estimation Results for the Switchboard Task. Here, "SL" represents "scaled likelihoods." "align I" and "align II" refer to alignments computed from two different estimates of the scaled likelihoods. In the "align I" case, the scaled likelihoods are computed from raw posteriors while forward-backward posteriors are used for "align II" (note that both sets of scaled likelihoods are computed by dividing the given posteriors by priors obtained from the hard-targets of the training data). "raw priors" refers to adaptive priors computed from the raw posteriors, and "fwd-bckwd priors" refers to adaptive priors computed from the forward-backward posteriors.

EER of 34.68% is actually obtained by using raw priors and *forward-backward* posteriors to compute scaled likelihoods. The "SL computed from fwd-bckwd postrs and fwd-bckwd priors" curve, by comparison, performs worse by an absolute margin of 0.79%. Thus, it appears that while the forward-backward recursion may yield improved posterior estimates, these posteriors do not lend themselves to computing state priors. One might conclude that the posteriors obtained from the forward-backward recursion are biased in some way. Indeed, the state transition probabilities used to derive the forward-backward posteriors are themselves biased by the smoothing parameters, ϵ and ρ. The possibility that the forward-backward posteriors are also biased is therefore not unexpected.

Based on a comparison of the results obtained for the "align I" and the "align II" curves, we note that much of the gain obtained from computing forward-backward posteriors may actually be attributed to improved time-alignments. In particular, we note that "SL computed from raw posteriors and raw priors" performs significantly better with the "align II" scheme than with "align I." Although a thorough investigation into time-alignment is beyond the scope of this paper, it is important to point out that much of the discrepancy between align I and align II has to do with the number of "empty" alignments (i.e. utterances for which no alignment was found) for the two schemes. In the align I case, 5028 out of a total of 38652 words in the test set were designated as empty alignments, whereas only 662 words were left empty for align II. Each of the words corresponding to an "empty" alignment were simply assigned the lowest confidence score of any word in the entire corpus. Thus, the number of empty words in an alignment has a significant impact on confidence estimation.

5 System Combination Experiments on the Switchboard Task

As described in section 1, one of the primary thrusts of this project is to combine components of independently trained ASR systems to improve recognition performance and confidence estimation. The previous section describes a set of experiments along this vein (recall that the word hypotheses in the preceding section were obtained from the SRI system). In this section, we extend this line of research by incorporating the ICSI recognizer into the process of ranking competing sentence-level hypotheses. This approach differs from that of the preceding section in that the ICSI recognizer actually factors into the recognition process. Thus, the goal in this section is not only to yield improved confidence estimates for existing word hypotheses, but also to improve the accuracy of the word hypotheses themselves. As in previous sections, we will ignore the details of the decoding process (i.e. the process of actually determining an "n-best list" of the most likely sentence-level hypotheses for a given utterance). Instead, we will focus on the task of computing scores for each competing sentence among an n-best list of the top candidates. The following section will show how these scores can then be applied to the following two tasks: 1) extracting the most likely sentence-level hypothesis and 2) computing confidence measures for each hypothesized word.

5.0.9 Model Definition

To derive a model for evaluating the relative likelihoods of the top N sentence-level hypotheses for a given utterance, we begin by computing the probability of sentence S conditioned on the input acoustics X:

$$p(S \mid X, M) = \frac{p(S \mid M_\ell) p(X \mid S, M_a)}{p(X \mid M)}$$

Here, we define $M = \{M_\ell, M_a\}$ where M_ℓ represents a given language model and M_a represents a given acoustic model. The terms $p(S \mid M_\ell)$ and $p(X \mid S, M_a)$ shown above represent probability estimates or "scores" obtained from the language and acoustic models, respectively. For the purposes of this discussion, we will again ignore the details of how these scores are computed and simply assume that they are available.

In the above equation, we note that $p(X \mid M)$ is the same for each sentence-level hypothesis, S, for the given utterance. Since we are only interested in the likelihood of S relative to the other sentences in the n-best list, we can ignore the $p(X \mid M)$ term and simply compute sentence-level scores in terms of the joint probability $p(S, X \mid M)$. After applying smoothing parameters to both the acoustic and language model terms in the above equation and taking the logarithm of $p(S, X \mid M)$, we arrive at the following linear model for scoring a sentence-level hypothesis, S:

$$PP(S) = \log p(S, X \mid M) = \rho_\ell \log p(S \mid M_\ell) + \rho_a \log p(X \mid S, M_a)$$

Here, the ρ_ℓ and ρ_a terms are fixed smoothing parameters that weight the contributions of the language and acoustic model terms. For the experiments described in this section, the above model is extended to include log probability scores obtained from multiple sources. More specifically, we adopt a model that employs acoustic model scores obtained from a GMM system supplied by SRI international and from the ANN system of ICSI. Both acoustic scores are computed by summing log likelihoods for each frame in a forced alignment to the given sentence, S. Since the GMM system directly estimates likelihoods of the form $p(x^n \mid q_i^n)$, the acoustic model score $AM_{GMM}(S)$ can be computed as follows:

$$AM_{GMM}(S) = \sum_{n=1}^{N} \log p(x^n \mid q_{k_n}^n) \tag{9}$$

Here, $q_{k_1}, q_{k_2}, \ldots, q_{k_N}$ represents a "best path" through the utterance for the given hypothesis, S.

As shown in the previous sections, the likelihoods obtained from the ANN system are composed of separate posterior and prior terms. Thus, the acoustic model score for the ANN system can be broken down into two separate scores, $AM_{ANN,post}(S)$ and $AM_{ANN,prior}$:

$$AM_{ANN,post}(S) = \sum_{n=1}^{N} \log p(q_{k_n}^n \mid x^n) \tag{10}$$

$$AM_{ANN,prior}(S) = \sum_{n=1}^{N} \log p(q_{k_n}) \tag{11}$$

Again, $q_{k_1}, q_{k_2}, \ldots, q_{k_N}$ represents the most likely state sequence for the given sentence, S. Note that the GMM system and the ANN use different states—thus, each system must compute its own best path. Given these scores, the complete sentence-level log likelihood, $PP(S)$, is computed as follows:

$$PP(S) = \rho_{a,GMM} AM_{GMM}(S) + \rho_\ell LM(S) + \rho_{postr} AM_{postr}(S) - \rho_{prior} AM_{prior}(S) \tag{12}$$

The sentences contained in each n-best list are then ranked according to their sentence-level scores, and the top-ranking sentence is chosen as the final hypothesis.

5.0.10 Experiment I

In the following discussion, we describe several experiments that test the word recognition performance of the combined ASR system defined by equation 12. The goal of the first of these experiments, labeled "Experiment I" is to compare the performance of various estimators of $p(q_i)$ for the purposes of computing $AM_{prior}(S)$. As was done in section 3, the priors are computed by time-averaging one of the following three sources: the posteriors of the cross-validation data, the hard targets of the training data, and the posteriors of the

experiment	WER (%)	$\rho_{a,GMM}$	ρ_ℓ	ρ_{postr}	ρ_{prior}
HT priors	34.0	1	10.45	0.135	0.195
CV priors	33.7	1	12.75	0.11	0.175
adaptive priors	34.0	1	9.55	0.09	0.14

Table 4: WER Results for Experiment I (all scores)

experiment	WER (%)	ρ_{postr}	ρ_{prior}
HT priors	42.4	1	2.49
CV priors	42.1	1	2.481
adaptive priors	42.1	1	1.974

Table 5: WER Results for Experiment I (ANN only)

given speaker's test data. In each case, the posteriors are derived from the same hard-target trained ANN used in section 4. After computing the various acoustic and language model scores of equation 12, the smoothing parameters for each of these scores were tuned on the cross-validation corpus to yield an optimized WER. Thus, for each setting of the tuning parameters, each n-best list in the cross validation corpus was rescored according to the model of equation 12. A WER score for the given parameterization was then determined based on the correctness of the top sentence-level hypotheses.

For "experiment I," two separate parameterizations were tested. In the first of these, which we will call "all scores," the language model weight, ρ_ℓ was tuned to a resolution of 0.05, while ρ_{prior} and ρ_{postr} were tuned to resolutions of 0.005. Meanwhile, $\rho_{a,GMM}$ was held constant at 1. Note that all parameters were tuned so as to allow for every possible combination of values within the given tuning resolution. In the second parameterization, which we will call "ANN only," $\rho_{a,GMM}$ and ρ_ℓ were set to 0, and ρ_{postr} and ρ_{prior} were tuned to resolutions of 0.001. The following table summarizes the final WER test results obtained from these experiments. Note that the baseline WER of the SRI system is 34.4%.

As shown in table 4, the inclusion of the posterior and prior scores from the ANN system yields up to a 0.7% absolute improvement over the 34.4% baseline of the the SRI system. Based on these results, we may infer that the information provided by the two probability estimators—the GMM and the ANN—is at least somewhat complimentary. Whether or not this complimentary nature is inherent to the two models in general is uncertain. However, the above results provide a great deal of support for the concept of using integrated ANN and GMM estimators to rescore sentence-level hypotheses.

Table 4 also underscores the primary theme of the previous sections—namely, that priors computed by time-averaging posteriors tend to outperform the standard hard-target priors used in most ANN systems. However, contrary to the results found on Aurora, table 4 shows that the CV priors actually outperform the adaptive priors on the "all scores" test. Although the difference in perfor-

mance here is significant (0.3%), we note that subsequent results obtained in the following section show adaptive priors performing at the same WER as CV priors. A similar equivalence in performance between adaptive and CV priors was also predicted by the results of the tuning corpus. Thus, we consider the relatively lackluster showing of the adaptive priors in table 4 to be an unexpected (and possibly anomalous) test result.

Regardless of what factors may have contributed to the WER scores shown above, we note that in general, the adaptive priors appear to be much more effective for the Aurora task than for Switchboard. This discrepancy in gains may be partially attributed to the difficulty of the Switchboard task—in general, performance improvements obtained on Switchboard tend to fairly small. However, it's also possible that adaptive priors tend to be particulary useful for speech tasks that involve various types of background noise (as is the case with Aurora). Indeed, it's worth noting that in the Aurora experiments, the priors were not only adapted to the given speaker but also to the given noise environment. Given the largely uniform acoustic conditions under which the Switchboard corpus was recorded, the experiments of this section do not lend themselves to the same sort of noise-specific adaptation performed on Aurora.

5.0.11 Experiment II

The second experiment of this section involved further testing the performance of $AM_{postr}(S)$ and $AM_{prior}(S)$ based on various new parameterizations and probability estimators. In particular, experiment II involves testing a new smoothing parameter, ρ_{pp}, which is incorporated into the estimation of the adaptive priors. The revised model for estimating these adaptive priors is defined below:

$$p(q_i) = \frac{1}{N} \sum_{n=1}^{N} \frac{p(q_i^n \mid x^n)^{\rho_{pp}}}{\sum_j p(q_j^n \mid x^n)^{\rho_{pp}}} \tag{13}$$

Here, the ρ_{pp} parameter simply smooths the individual posterior estimates used to compute the adaptive prior, $p(q_i)$. A significant feature of this revised model is that the smoothed posteriors are renormalized to sum to one. The above model is therefore consistent with the original adaptive model of equation 6 insofar as the renormalized, smoothed posteriors can themselves be thought of as posterior estimates.

"Experiment II" also provides a comparison between the original hard-target trained ANN used in Experiment I and a new ANN system trained on soft-targets. This "soft-target ANN" is trained on a "soft-alignment" of the scaled likelihoods of the training data (i.e. estimates of $p(q_i^n \mid x^n)/p(q_i)$) with the reference phone sequence. To derive the soft-alignments, we simply use the duration models of section 4 to compute forward-backward posteriors at each frame while conditioning on the phone sequence of the reference transcript. These forward-backward posteriors are then used as targets to train a "soft-target ANN" having the same general architecture and specifications of the original hard-target ANN. The only significant differences in training between the two ANNs (other

than the training targets) involves the choice of network non-linearities and error criteria. While the hard-target ANN was trained using softmax non-linearities, sigmoidal non-linearities were employed along with a cross-entropy error criterion for the soft-target ANN to ensure stability in training (we note that using softmax on the soft-target ANN resulted in training errors). Given that ANN training is a topic of secondary concern to this project, we will leave out any further discussion of why the softmax criterion failed to admit a training for the soft-target ANN. Suffice it to say that the reasons for this failure are unknown and may well lie with a fault in the software.

For "experiment II," the ρ_{pp} parameter was tuned from 1.0 to 0.5 by decrements of 0.1 for both the posteriors obtained from the hard-target ANN and the posteriors obtained from the soft-target ANN (i.e. the HT-ANN posteriors and the ST-ANN posteriors). These experiments generally showed some improvement in WER on the cross-validation set when ρ_{pp} was tuned below 1.0.

Two different alignments were derived for each sentence-level hypothesis to compute the acoustic model scores. These included the "HT-ANN alignment" obtained from scaled likelihoods computed from HT-ANN posteriors and the "ST-ANN alignment" computed similarly from the ST-ANN system. Thus, a total of 8 tests were performed on each combination of the following categories: "all scores" vs. "ANN only," "HT-ANN posteriors" vs. "ST-ANN posteriors," and "HT-ANN alignment" vs. "ST-ANN alignment." For the "ANN only" test, each posterior set performed best on the cross-validation corpus when used with its corresponding alignment. However, on the "all scores" test, the ST posteriors yielded a slightly lower WER when the ST-ANN alignment was replaced with the HT-ANN alignment to compute $AM_{postr}(S)$ (note, however, that a similar improvement was *not* observed when the HT-ANN posteriors were used in conjunction with the ST-ANN alignment). Given these tuning results, one might suppose that further gains can be achieved by incorporating scores from both posterior sets into the combined GMM + ANN system.

As a final experiment, the "all scores" test was conducted using forward-backward posteriors (i.e. $p(q_i \mid X_1^N)$) of the same form used in section 4 to compute $AM_{postr}(S)$. As before, the raw posteriors from the ANN were used to compute the adaptive priors. Given the best value of ρ_{pp} found in the previous tuning, the forward-backward smoothing parameter, ρ, was raised from 0 to 0.5 by increments of 0.1 while ϵ was fixed at 0.1. For each experiment, the WER was found to go up slightly as ρ was raised above 0. Thus, the best results for the forward-backward posteriors are obtained when the transition values of the forward-backward recursions are completely smoothed (i.e. $\rho = 0$). We note that this corresponds to the special case where the forward-backward posteriors are equal to the raw posteriors. To rationalize these results, we might postulate that in the combined ANN + GMM system, forward-backward reestimation obscures much of the novel information contained in the ANN posteriors.

A list of selected test results for the "HT-ANN alignment" of "experiment II" is provided in tables 6, and 7, along with the corresponding optimized parameter values.

experiment	WER (%)	ρ_{pp}	$\rho_{a,GMM}$	ρ_{ℓ}	ρ_{postr}	ρ_{prior}
HT-ANN posteriors	33.7	0.80	1	11.85	0.095	0.65
ST-ANN posteriors	33.7	0.90	1	12.1	0.08	0.135

Table 6: WER Results for Experiment II (all scores)

experiment	WER (%)	ρ_{pp}	ρ_{postr}	ρ_{prior}
HT-ANN posteriors	42.1	0.80	1	2.337
ST-ANN posteriors	42.8	0.40	1	2.75

Table 7: WER Results for Experiment II (ANN only)

As shown above, optimization of the ρ_{pp} parameter improves the WER of the HT-ANN posteriors (using adaptive priors) from 34.0% to 33.7% for the "All scores" test. This result matches the WER recorded for the CV priors in "experiment I." However, it's interesting to note that optimizing ρ_{pp} resulted in only small improvements on the tuning corpus (usually on the order of 0.05% or less)—thus, it seems unlikely that the ρ_{pp} parameter should yield an improvement as large as 0.3% on a test set for which it was not optimized. Given the peculiarity of these findings, we should again stress that the above results may be more indicative of anomalies in the test data than in any underlying improvement in our scoring system. Note that this hypothesis is also supported by table 7, which shows that optimizing ρ_{pp} results in no significant improvement on the "ANN only" test.

As for the experiments involving the soft target ANN, no significant change in WER is observed when the HT-ANN posteriors are replaced with ST-ANN posteriors for the "all scores" test. However, on the "ANN only" test, exchanging HT-ANN posteriors with ST-ANN posteriors actually degrades the recognition performance by 0.7%. In response to these findings, we should again emphasize that the soft-target training software used in this experiment behaved in a somewhat suspect manner, possibly due to an underlying fault. Given these circumstances, the ST-ANN results should be viewed as inconclusive.

5.0.12 Experiment III

Following the recognition experiments outlined in the previous sections, an additional experiment was conducted to test the confidence estimation performance of the combined ANN + GMM system. In this experiment, a standard dynamic progamming technique was used to group the individual word-level hypotheses of an n-best list into sets of competing hypotheses based on their position within the given sentence. For instance, given the sentences:

> Only a few sections left!
> No lonely Texans slept.

The word "Only" might be grouped with "No" or with "lonely" given that each of these words comes up at approximately the same time in the hypothesized

experiment	EER (%)
baseline (GMM only)	25.66
ANN + GMM system, HT priors	25.66
ANN + GMM system, CV priors	25.47
ANN + GMM system, adaptive priors without smoothing	25.52
ANN + GMM system, adaptive priors with smoothing	25.35

Table 8: Confidence Results for ANN + GMM System. Here, the baseline represents the case where only the acoustic model and the language model scores from the GMM-based system are used to perform sentence-level scoring. The "adaptive priors with smoothing" curve represents results obtained when the ρ_{pp} parameter is set at its optimized value of 0.8. In the "adaptive priors without smoothing" case, ρ_{pp} is set to 1.

sentences. Presumably, the word "left" would be grouped with "slept," since both of these words occur at the ends of the hypotheses. Thus, we consider "left" and "slept" to be *competing* words in the given n-best list. Assuming that we can sort each word-level hypothesis contained in a given n-best list into 1 of M competing groups (further details on this approach can be found in [8]), we may compute the word-level posterior, $p(w_i^k \mid X_1^N)$, for a given word w_i occuring in group k as follows:

$$p(w_i^k \mid X_1^N) = \frac{\sum_i exp(PP(\dot{S}_i))}{\sum_i exp(PP(S_j))} \tag{14}$$

Here, $\dot{S} = \dot{S}_1, \dot{S}_2, \ldots, \dot{S}_L$ represents the set of all sentences that contain w_i in group k, while $S = S_1, S_2, \ldots, S_N$ represents the complete set of all sentences contained in the n-best list. Given this formulation, a set of word-level posteriors was computed for the sentence-level scores obtained from the combined ANN + GMM system in "experiment I." An additional set of word posteriors was also derived from the "adaptive priors" scores of "experiment II" where the ρ_{pp} parameter was set to its optimized value of 0.8. We will refer to this latter case as "adaptive priors with smoothing." Note that word-level posteriors were only computed for the top sentence-level hypotheses of the n-best lists given the *original* sentence ranking (i.e. prior to performing any rescoring). Thus, each of the DET curves obtained in this experiment are taken from the same set of word-level hypotheses. The results are shown in figure 3 and in table 8.

Note that we have included a baseline corresponding to the "GMM only" case (i.e. the case where only the $AM(S)_{GMM}$ and $LM(S)_{GMM}$ knowledge sources are used). As shown in table 8, the performance of the various curves follows a trend similar to that observed in the recognition experiments of the previous section. We note, however, that the best performance is obtained from the "adaptive priors with smoothing" curve, which outperforms the baseline by an absolute margin of 0.31%. Based on these results, it appears that the ρ_{pp} parameter may, indeed, be of some significance in improving the estimation of the state priors—at least for the purposes of computing word-level confidence.

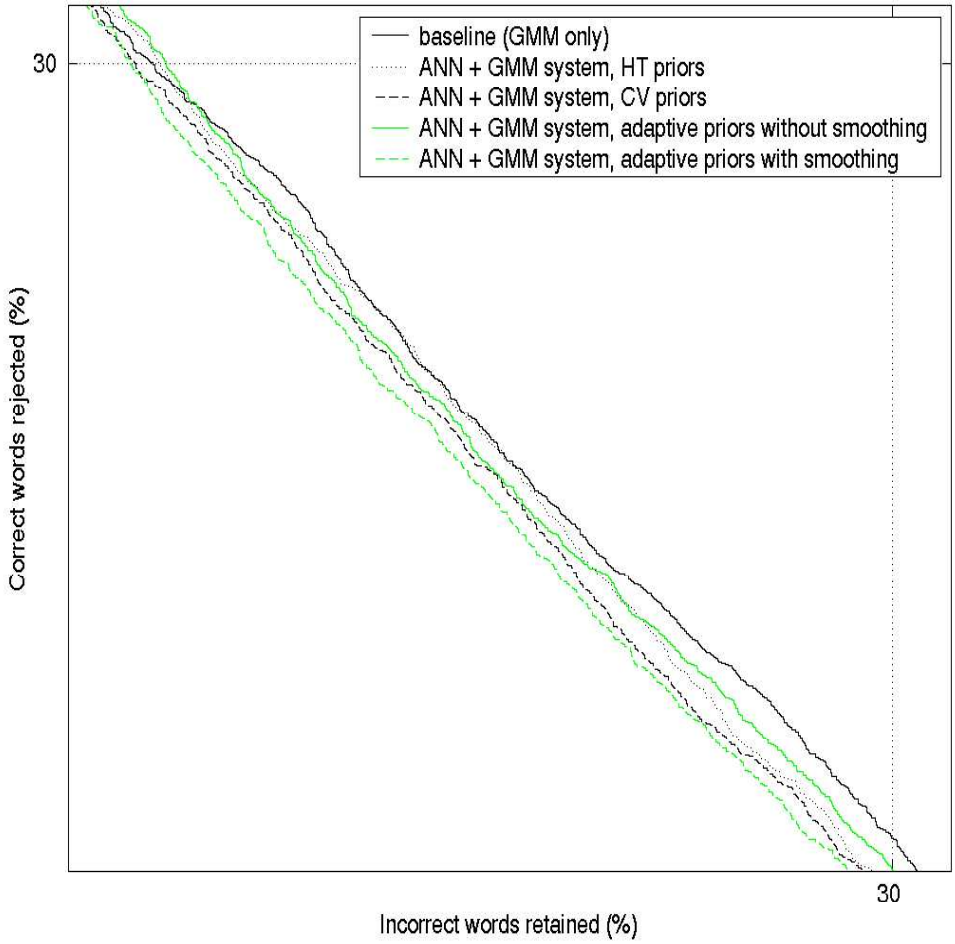

Figure 3: DET Curves for the Switchboard Task. Here, the baseline represents the case where only the acoustic-model and the language-model scores from the GMM-based system are used to perform sentence-level scoring. The "adaptive priors with smoothing" curve represents results obtained when the ρ_{pp} parameter is set at its optimized value of 0.8. In the "adaptive priors without smoothing" case, ρ_{pp} is set to 1.

6 Extensions

In this section, we reexamine some of the theoretical results uncovered in section 3 and outline a direction for future work in the area of speaker/noise adaptation. In particular, we will review the adaptive prior model of equation 6 and discuss how adaptive priors might be employed in a likelihood-based system (such as the GMM-based classifier used in the previous section). Before going into possible extensions of this work, let us briefly review the theory of section 3: Given a joint probability, $p(q_i, x) = p(q_i \mid x)p(x)$, we can compute the prior probability of state q_i by marginalizing over all x. Thus, the prior can be expressed as:

$$\pi_i = \int_x p(q_i \mid x)p(x)dx \tag{15}$$

Here, we use π_i to denote $p(q_i)$. Since $p(q_i \mid x)$ is a function of the random variable, x, it follows from equation 15 that $\pi_i = E[p(q_i \mid x)]$. Thus, a natural estimate for π_i can be obtained by time-averaging $p(q_i \mid x)$ (assuming, of course, that the group of posteriors over which we are averaging is adequately representative of π_i). We therefore arrive at the following model for estimating the state priors:

$$\pi_i = \frac{1}{N}\sum_{n=1}^{N} p(q_i^n \mid x^n) \tag{16}$$

As demonstrated in section 3, the model of equation 16 allows for the adaptation of state priors to a given speaker or noise environment. Given the success of this approach, particularly in experiments performed on the Aurora task (see section 3), one might wonder how the concept of adaptive priors can be incorporated into a likelihood-based system (i.e. a system that directly estimates $p(x \mid q_i)$ without first computing state posteriors). For such a system, we can express equation 16 in the following form:

$$\pi_i = \frac{1}{N}\sum_{n=1}^{N} \frac{\pi_i p(x^n \mid q_i^n)}{\sum_j \pi_j p(x^n \mid q_j^n)} \tag{17}$$

Here, we have simply substituted $p(q_i^n \mid x^n) = \frac{\pi_i p(x^n \mid q_i^n)}{\sum_i \pi_i p(x^n \mid q_i^n)}$ into the former model (note that this representation of $p(q_i^n \mid x^n)$ follows from Bayes rule). Unlike equation 16, the preceding model does not admit a unique solution for π_i. To verify this fact, we can easily show that for any set of likelihoods in an M-dimensional state space, the preceding model has M trivial solutions of the form: $\pi_i = \mathbf{1}(i = j)$ for some j. Other solutions with fewer zeros may also exist—however, in general, we are not guaranteed a solution where $\pi_i > 0 \ \forall \ i$. This fact should be somewhat disconcerting, particularly is we espouse the view that a "reasonable" solution for the priors should always exist (preferably a solution where all state-priors are non-zero). To ameliorate this concern, and to allow for speaker/noise adaptiation similar to that performed on the ANN system, we propose the following procedure: Given some best estimate,

$\boldsymbol{\pi}^* = (\pi_1^*, \pi_2^*, \ldots, \pi_M^*)$, of the state priors, we can weight the likelihoods for the various states to ensure consistency between the weighted likelihoods and $\boldsymbol{\pi}^*$. That is, given $\boldsymbol{\pi}^*$, we weight $p(x^n \mid q_i^n)$ for all n by some constant, α_i. These constants are chosen such that $\boldsymbol{\pi}^*$ is consistent with our weighted likelihoods (note that by "consistency," we mean that $\boldsymbol{\pi}^*$ forms a solution to equation 17 for the given set of likelihoods). Using this approach, we arrive at the following model for computing $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_M)$:

$$\pi_i^* = \frac{1}{N} \sum_{n=1}^{N} \frac{\pi_i^* \alpha_i p(x^n \mid q_i^n)}{\sum_j \pi_j^* \alpha_j p(x^n \mid q_j^n)} \tag{18}$$

The above model is identical to that of equation 17, except that the likelihoods have been weighted by α_i. As previously stated, our goal in introducing this model is to ensure consistency between a given set of likelihoods and our best estimate of the state priors. However, we are left with a number of questions: How do we find a "best estimate" of the priors? Given these priors, how do we solve for $\boldsymbol{\alpha}$? Does a solution for $\boldsymbol{\alpha}$ always exist, and if so, is it unique? On the latter point, we can show that a solution for $\boldsymbol{\alpha}$ does exist given any $\boldsymbol{\pi}^*$ and that this solution is unique when $\pi_i^* > 0 \ \forall \ i$. A proof of this result can be found in Appendix A. Given that we are guaranteed a unique solution for any "reasonable" $\boldsymbol{\pi}^*$ (i.e. any $\boldsymbol{\pi}^*$ where none of the prior state probabilities are zero), we may employ any standard iterative approach to solving for $\boldsymbol{\alpha}$ (e.g. gradient descent or Newton-Raphson). Thus, we will simply assume that a solution for $\boldsymbol{\alpha}$ exists and that this solution can be readily obtained. As for the question of how to determine a "best estimate" of the priors, we will leave this question open—however, the experiments described in this section should shed some light on this problem.

6.1 Experiments

To test the model of equation 18, we essentially repeated the Aurora confidence experiments of section 3 for various sets of scaled likelihoods. These scaled likelihoods were computed from three different estimates of the state priors, including HT priors (i.e. priors trained on the hard targets of the training data) and a set of "uniform priors" where $\pi_i = \frac{1}{M} \ \forall \ i$. To test the effectiveness of the $\boldsymbol{\alpha}$ terms when a "bad" estimate of the priors is chosen, we computed a third set of priors by inverting the HT priors and then renormalizing. These "inverse HT priors" are defined as follows:

$$\pi_{invHT,i} = \frac{\pi_{HT,i}^{-1}}{\sum_j \pi_{HT,j}^{-1}} \tag{19}$$

Here, $\pi_{HT,i}$ denotes an HT prior and $\pi_{invHT,i}$ denotes an inverse HT prior.

As was done in section 3, scaled likelihoods were computed for each type of prior. The set of scaled likelihoods for a given speaker and a given noise environment were then adapted to the particular prior by computing $\boldsymbol{\alpha}$as defined

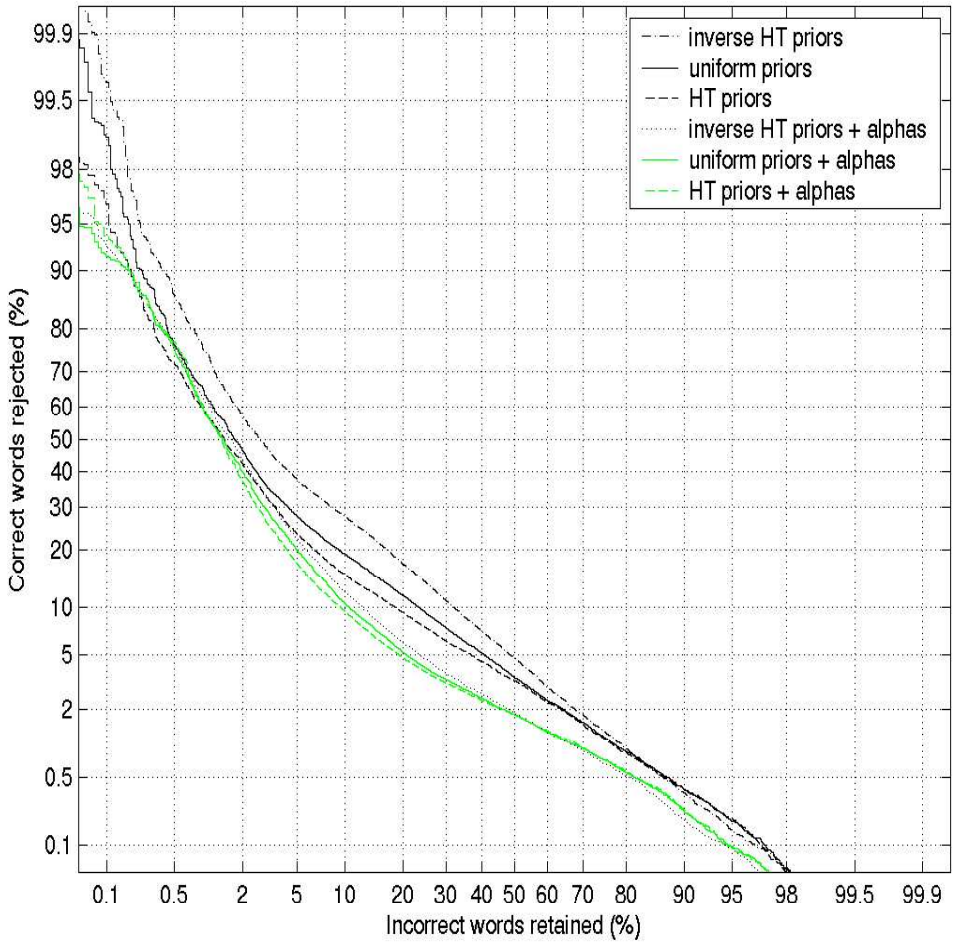

Figure 4: DET Curves for the Aurora Task. Here, "alphas" represents the inclusion of the α parameters of equation 18 to weight the likelihood terms.

experiment	*EER (%)* without $\boldsymbol{\alpha}$ terms	*EER (%)* with α terms
inverse HT priors	18.44	11.19
uniform priors	14.86	10.29
HT priors	12.90	9.73

Table 9: Confidence Results for the Aurora Noisy Digits Task. Here, "alphas" represents the inclusion of the $\boldsymbol{\alpha}$ parameters of equation 18 to weight the likelihood terms.

in equation 18. We then computed confidence measures for the scaled likelihoods for each of the three priors with and without their corresponding $\boldsymbol{\alpha}$ terms. Thus, confidence measures were computed for a total of 6 different sets of scaled likelihoods. The resulting DET curves and EERs for each of these experiments are shown in figure 4 and table 9, respectively.

Note from table 9 that the inclusion of the $\boldsymbol{\alpha}$ terms vastly improves the performance of each type of scaled likelihood. In the case of scaled likelihoods computed from HT priors, the EER obtained with the $\boldsymbol{\alpha}$ terms is 9.73%—only 0.13% worse than the EER recorded in section 3 for scaled likelihoods computed from adaptive priors. We also note that the latter set of scaled likelihoods (i.e. those computed from adaptive priors) implicitly satisfies equation 17. Thus, each of the best EER results obtained on Aurora have come from scaled likelihoods that are consistent with a particular "non-trivial" set of priors (i.e. a set of priors having no zero elements).

These findings lend support to the notion of establishing consistency between a given set of likelihoods and our best estimate of the state priors. However, the results of table 9 are still rather inconclusive, as we have not shown the consistency criterion of equation 17 to be significant in the general case. That is, these results do not preclude the possibility that a given likelihood estimator might yield outstanding test results without abiding by equation 17 for any non-trivial set of priors. We have also yet to explore the extent to which the notion of consistency is satisfied in existing likelihood estimators (i.e. do certain estimators implicitly define some non-trivial set of priors according to equation 17?)

What we have established, however, is a starting point for an interesting and potentially fruitful new area of research. Some of the more obvious applications of this notion of ensuring consistency between likelihoods and priors have already been discussed—namely speaker and noise adaptation. However, one might also imagine employing the theory of this section to extract new and potentially useful statistics for use in ASR systems. For instance, we might try using equations 17 or 18 to factor a given set of likelihoods into uniquely specified posterior and prior terms. These posteriors and priors could then be used as separate knowledge sources to perform sentence-level rescoring, as demonstrated in section 5. Here, the hope would be, of course, that improved WER results can be obtained when one of these statistics—either the posteriors or the priors—are

weighted more heavily than the other. Whether or not this or other techniques will lend themselves to improvements in recognition or confidence performance remains to be seen. However, we feel that the results obtained in this section are encouraging enough to warrant the continued investigation of these topics.

7 Conclusions

The findings of this thesis are largely summarized by the following three themes:

1. With few exceptions, we have shown that adaptive priors computed from the test data itself tend to outclass static prior estimates in evaluations of both recognition and confidence estimation. This adaptive estimation paradigm appears to work particularly well on noisy tasks such as Aurora, where the potential exists to not only adapt priors to the current speaker, but also to the current noise environment.

2. For the purposes of word-level confidence estimation, we have found that local posterior estimates obtained from an ANN can be improved by means of forward-backward reestimation. However, we have not shown this technique to yield any benefit on systems that employ both GMM-based and ANN-based acoustic models.

3. We have demonstrated the utility of combining sentence-level scores obtained from an ANN-based acoustic model with acoustic and language model scores obtained from a GMM-based system for the purposes of rescoring n-best lists. This technique has been shown to yield significant improvements on conversational speech tasks in evaluations of both word-level recognition and confidence estimation.

8 Acknowledgements

The author would like to thank the individuals who contributed to this research through their attention and guidance. Principal among these are Nelson Morgan, who acted as P.I. for this project and who served as my official advisor. Many thanks are also due to Andreas Stolcke, who provided access to the SRI speech recognition system and who helped guide my efforts at combining ASR systems. I would also like to express a heart-felt thank you to Eric Fosler Lussier and to Dan Ellis, both of whom played key roles in developing much of the underlying software and many of the systems used in this research. Finally, I would like to thank my sponsors at the Department of Defense for their generous funding of this research.

9 Appendix

Theorem 1 *Given a set of likelihoods of the form,* $p(x^n \mid q_i^n)$*, and a set of priors,* $\boldsymbol{\pi} = (\pi_0, \pi_1, \ldots, \pi_{M-1})^T$*, where* $p(x^n \mid q_i^n) > 0 \ \forall \ (i,n)$*,* $\sum_i \pi_i = 1$*,*

and $\pi_i > 0 \ \forall \ i$*, the following set of equations admits a unique solution for* $\boldsymbol{\alpha} = (\alpha_0, \alpha_1, \ldots, \alpha_{M-1})^T$ *under the constraint,* $\sum_i \alpha_i = 1$*:*

$$\pi_i = \frac{1}{N} \sum_{n=1}^{N} \frac{\pi_i \alpha_i p(x^n \mid q_i^n)}{\sum_j \pi_j \alpha_j p(x^n \mid q_j^n)} \tag{20}$$

proof To prove that $\boldsymbol{\alpha}$ is uniquely specified by $\boldsymbol{\pi}$ (under the above stated conditions), we begin by showing that at least one solution for $\boldsymbol{\alpha}$ exists given any valid $\boldsymbol{\pi}$ vector. To prove this, we will first show that $\boldsymbol{\pi}$, as defined above, is a continuous function of $\boldsymbol{\alpha}$. Using the shorthand, $p_i^n = p(x^n \mid q_i^n)$, we compute the full derivative of the model as follows:

$$0 = \sum_{n=1}^{N} \left[\frac{p_i^n d\alpha_i}{\sum_j \pi_j \alpha_j p_j^n} - \frac{\alpha_i p_i^n \sum_j \left[\alpha_j p_j^n d\pi_j + \pi_j p_j^n d\alpha_j \right]}{(\sum_j \pi_j \alpha_j p_j^n)^2} \right] \tag{21}$$

Let us tempoarily relax the constraint that $\sum_i \alpha_i = 1$. If we assume that $d\alpha_{j:j \neq i} = 0$, and that $d\alpha_i$ is finite and non-zero, then the preceding expression reduces to:

$$d\alpha_i \sum_{n=1}^{N} \left[\frac{p_i^n}{\sum_j \pi_j \alpha_j p_j^n} - \pi_i \alpha_i \left(\frac{p_i^n}{\sum_j \pi_j \alpha_j p_j^n} \right)^2 \right] =$$

$$\sum_{n=1}^{N} \frac{\alpha_i p_i^n \sum_j \alpha_j p_j^n d\pi_j}{(\sum_j \pi_j \alpha_j p_j^n)^2} \tag{22}$$

Let us now assume that one or more elements of $d\boldsymbol{\pi}$ are infinite, and hence, that $\boldsymbol{\pi}$ is *not* a continuous function of $\boldsymbol{\alpha}$. Then the right-hand side of the above equation must be equal to either ∞ or 0. However, we note that the bracketed expression on the left-hand side of the above equation must always be strictly positive (this follows from close inspection). Given that $d\alpha_i$ is finite and non-zero, the entire left-hand side must therefore also be finite and non-zero, which contradicts our previous assumption that one or more elements of $d\boldsymbol{\pi}$ are infinite. Thus, we have shown that all elements of $d\boldsymbol{\pi}$ must be finite under the given conditions, which implies that $\boldsymbol{\pi}$ is a continuous function of $\boldsymbol{\alpha}$.

Given this result, we assert the following: If at least one solution for $\boldsymbol{\alpha} = (\alpha_0, \alpha_1, \ldots, \alpha_{M-1})^T$ exists given any $\boldsymbol{\pi} = (\pi_0, \pi_1, \ldots, \pi_{M-1})^T$, then at least one solution for $\boldsymbol{\alpha} = (\alpha_0, \alpha_1, \ldots, \alpha_{M-1}, \alpha_M)^T$ must exist given any $\boldsymbol{\pi} = (\pi_0, \pi_1, \ldots, \pi_{M-1}, \pi_M)^T$. This follows from the fact that the sets of valid $\boldsymbol{\pi}$ vectors for all state-spaces of dimensionality $M-1$ form the endpoints of the set of all valid M-dimensional $\boldsymbol{\pi}$ vectors. To illustrate this point, consider a 3-dimensional state-space where $\boldsymbol{\pi} = (\pi_0, \pi_1, \pi_2)^T$. The set of all $\boldsymbol{\pi}$ vectors where $\sum_i \pi_i = 1$ and $\pi_i \geq 0 \ \forall \ i$ forms a simplex in $\Re^3$ whose endpoints lie along the lines, $\pi_0 + \pi_1 = 1$, $\pi_1 + \pi_2 = 1$, and $\pi_0 + \pi_2 = 1$. These lines define the set of all valid $\boldsymbol{\pi}$ in any 2-dimensional state-space (assuming that all π_i are constrained to be non-negative). Similarly, it can be shown that the sets of all valid $\boldsymbol{\pi}$ vectors of dimesionality $M-1$ form the endpoints for the set of all

valid M-dimensional $\boldsymbol{\pi}$ vectors. Since $\boldsymbol{\pi}$ is a continuous function of $\boldsymbol{\alpha}$, it follows that if at least one solution for $\boldsymbol{\alpha}$ exists given any $\boldsymbol{\pi}$ in an $(M-1)$-dimensional state-space, then at least one solution for $\boldsymbol{\alpha}$ must exist given any $\boldsymbol{\pi}$ in an M-dimensional state-space. We can therefore prove the existence of a solution for $\boldsymbol{\alpha}$ given any valid $\boldsymbol{\pi}$ vector by induction: Let us examine the special case where the state-space is one-dimensional. In this case, $\boldsymbol{\pi}$ is a scalar whose only valid value is $\pi = 1$. It can easily be shown that $\alpha = 1$ provides a solution for this prior. Thus, we have shown that a solution for $\boldsymbol{\alpha}$ exists for any valid $\boldsymbol{\pi}$ in a 1-dimensional state-space. By induction, it therefore follows that at least one solution for $\boldsymbol{\alpha}$ can be found for any valid $\boldsymbol{\pi}$ vector of dimensionality $M > 0$.

To prove that the solution for $\boldsymbol{\alpha}$ given any valid $\boldsymbol{\pi}$ vector is unique, we begin by defining the parameter vector, $\boldsymbol{\sigma} = (\sigma_0, \sigma_1, \ldots, \sigma_{M-1})^T$, where $\sigma_i = \pi_i \alpha_i$. Substituting into our original model gives us:

$$\pi_i = \frac{1}{N} \sum_{n=1}^{N} \frac{\sigma_i p(x^n \mid q_i^n)}{\sum_j \sigma_j p(x^n \mid q_j^n)} \tag{23}$$

We will now argue that the mapping between $\boldsymbol{\pi}$ and $\boldsymbol{\sigma}$ is one-to-one under the constraint that $\sum_i \sigma_i = 1$. Once we establish this, it is straightforward to show that a unique solution exists for $\boldsymbol{\alpha}$.

From previous arguments, we know that at least one solution exists for $\boldsymbol{\alpha}$, and hence, for $\boldsymbol{\sigma}$, given any valid $\boldsymbol{\pi}$ vector. Assuming that $\boldsymbol{\sigma}$ is defined on the same space as $\boldsymbol{\pi}$ (which it is, given that $\sum_i \sigma_i = 1$), we can prove that the mapping between $\boldsymbol{\pi}$ and $\boldsymbol{\sigma}$ is one-to-one by showing that $d\boldsymbol{\pi}^T d\boldsymbol{\sigma}$ must be either strictly positive or strictly negative if $d\sigma_i \neq 0$ for some i. Thus, let us begin by taking the full derivative of $\boldsymbol{\pi}$:

$$d\pi_i = \frac{1}{N} \sum_{n=1}^{N} \left[\frac{d\sigma_i p_i^n \sum_j \sigma_j p_j^n - \sigma_i p_i^n \sum_j d\sigma_j p_j^n}{(\sum_j \sigma_j p_j^n)^2} \right] \tag{24}$$

The preceding equation can be simplified by substituting $\hat{p}_i^n$ for $\frac{\sigma_i p_i^n}{\sum_j \sigma_j p_j^n}$. Multiplying by $d\sigma_i$ and summing over all i then yields the following expression for $d\boldsymbol{\pi}^T d\boldsymbol{\sigma}$:

$$d\boldsymbol{\pi}^T d\boldsymbol{\sigma} = \frac{1}{N} \sum_{n=1}^{N} \left[\sum_i \frac{(d\sigma_i)^2}{\sigma_i} \hat{p}_i^n - \sum_i d\sigma_i \hat{p}_i^n \sum_j \frac{d\sigma_j}{\sigma_j} \hat{p}_j^n \right] \tag{25}$$

Note from lemma 1 that the bracketed term in the above expression must be strictly positive if $d\sigma_i \neq 0$ for some i. Thus, we have shown that a one-to-one mapping exists between $\boldsymbol{\pi}$ and $\boldsymbol{\sigma}$ when we apply the constraint: $\sum_i \sigma_i = 1$. Moreover, since each $\boldsymbol{\pi}$ specifies a unique $\boldsymbol{\sigma}$, each $\boldsymbol{\pi}$ must also specify a unique $\boldsymbol{\alpha}$. Thus, we have proved the desired result.

Lemma 1 *Given* $\boldsymbol{p} = (p_1, p_2, \ldots, p_M)^T$, $\boldsymbol{\sigma} = (\sigma_1, \sigma_2, \ldots, \sigma_M)^T$, *and* $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_M)^T$, *where* $\sum_i p_i = 1$, $p_i > 0 \ \forall \ i$, *and* $\alpha_i > 0 \ \forall \ i$, *the following*

inequality must hold for all $\boldsymbol{\sigma}$ *where* $\sigma_i \neq 0$ *for some* i.

$$J = \sum_i \frac{\sigma_i^2}{\alpha_i} p_i - \sum_i \sigma_i p_i \sum_i \frac{\sigma_i}{\alpha_i} p_i > 0 \tag{26}$$

proof Taking the second derivative of J with respect to $\boldsymbol{p}$, we obtain:

$$\frac{\partial^2 J}{\partial \boldsymbol{p} \partial \boldsymbol{p}^T} = -\boldsymbol{\sigma}(\boldsymbol{\sigma}./\boldsymbol{\alpha})^T - (\boldsymbol{\sigma}./\boldsymbol{\alpha})\boldsymbol{\sigma}^T \tag{27}$$

Here, we have employed the ./ operator to signify element-by-element division. We may now pre-multiply and post-multiply the above expression by $\boldsymbol{p}$ to yield:

$$\boldsymbol{p}^T \frac{\partial^2 J}{\partial \boldsymbol{p} \partial \boldsymbol{p}^T} \boldsymbol{p} = -2\boldsymbol{p}^T \boldsymbol{\sigma}(\boldsymbol{\sigma}./\boldsymbol{\alpha})^T \boldsymbol{p} \tag{28}$$

For the moment, let us constrain all elements of $\boldsymbol{\sigma}$ to be strictly non-negative. Clearly, $\boldsymbol{p}^T \frac{\partial^2 J}{\partial \boldsymbol{p} \partial \boldsymbol{p}^T} \boldsymbol{p} < 0$ in this case—assuming that $\sigma_i \neq 0$ for some i. Thus, J must be concave in $\boldsymbol{p}$ if $\sigma_i \geq 0 \; \forall \; i$ and $\sigma_i \neq 0$ for some i. Since J is a concave function (under the given conditions), the upper contour set, S, of J evaluated at $J = 0$ forms a convex set in $\boldsymbol{p}$. Moreover, since $\boldsymbol{p}^T \frac{\partial^2 J}{\partial \boldsymbol{p} \partial \boldsymbol{p}^T} \boldsymbol{p} < 0$, $J \geq 0$ for all points contained in S. We can easily show that $J = 0$ when evaluated at the "corners" of the valid $\boldsymbol{p}$ space (i.e. $J = 0$ when $p_i = \mathbf{1}(i = j)$ for some j). Thus, all points in the $\boldsymbol{p}$ space must lie within S. But, since J is concave in $\boldsymbol{p}$, it follows that $J > 0$ for all $\boldsymbol{p}$ where $p_i > 0 \; \forall \; i$ (assuming all elements of $\boldsymbol{\sigma}$ are non-negative).

We can now show that $J > 0$ in the general case by making the following observation: If we change the sign of 1 or more non-zero elements of $\boldsymbol{\sigma}$ from positive to negative, the value of J must increase (this can be verified by inspection). Thus, $J > 0 \; \forall \; \boldsymbol{\sigma}$ where $\sigma_i \neq 0$ for some i, which completes our proof.

Theorem 2 *Given a set of likelihoods,* $p(x^n \mid q_i^n)$*, where* $p(x^n \mid q_i^n) > 0 \; \forall \; (i, n)$*, and given the model,*

$$\pi_i = \frac{1}{N} \sum_{n=1}^{N} \frac{\pi_i \alpha_i p(x^n \mid q_i^n)}{\sum_j \pi_j \alpha_j p(x^n \mid q_j^n)},$$

we can prove the following: If and only if $\alpha_i p(x^n \mid q_i^n) \neq \alpha_j p(x^n \mid q_j^n)$ *for some* i*,* j*, and* n*, there exists no more than one non-trivial solution (i.e. a vector solution containing no zero elements) for the state priors,* $\boldsymbol{\pi} = (\pi_0, \pi_1, \ldots, \pi_{M-1})^T$*, given any parameter vector,* $\boldsymbol{\alpha} = (\alpha_0, \alpha_1, \ldots, \alpha_{M-1})^T$*, where* $\alpha_i > 0 \; \forall \; i$.

proof To prove the preceding theorem, we will first examine under which conditions $d\alpha_i = 0 \; \forall \; i$ when the state-space is 2-dimensional (that is, when

$\boldsymbol{\alpha} = (\alpha_0, \alpha_1)^T$ and $\boldsymbol{\pi} = (\pi_0, \pi_1)^T$). We begin by defining the shorthand, $p_i^n = p(x^n \mid q_i^n)$. Taking the full derivative of the model yields:

$$0 = \sum_{n=1}^{N} \left[\frac{p_i^n d\alpha_i}{\sum_j \pi_j \alpha_j p_j^n} - \frac{\alpha_i p_i^n \sum_j \left[\alpha_j p_j^n d\pi_j + \pi_j p_j^n d\alpha_j \right]}{(\sum_j \pi_j \alpha_j p_j^n)^2} \right] \tag{1}$$

Let us now assume the state-space to be 2-dimensional. If we set $d\alpha_0$ and $d\alpha_1$ to 0, equation 1 reduces to,

$$\boldsymbol{A} d\boldsymbol{\pi} = \mathbf{0}, \tag{2}$$

where,

$$\begin{aligned} \boldsymbol{A} &= \begin{bmatrix} \boldsymbol{A_{1,1}} & \boldsymbol{A_{1,2}} \\ \boldsymbol{A_{2,1}} & \boldsymbol{A_{2,2}} \end{bmatrix} \\ &= \begin{bmatrix} \sum_{n=1}^{N} \left(\frac{\alpha_0 p_0^n}{\sum_j \pi_j \alpha_j p_j^n} \right)^2 & \sum_{n=1}^{N} \frac{\alpha_0 p_0^n \alpha_1 p_1^n}{(\sum_j \pi_j \alpha_j p_j^n)^2} \\ \sum_{n=1}^{N} \frac{\alpha_0 p_0^n \alpha_1 p_1^n}{(\sum_j \pi_j \alpha_j p_j^n)^2} & \sum_{n=1}^{N} \left(\frac{\alpha_1 p_1^n}{\sum_j \pi_j \alpha_j p_j^n} \right)^2 \end{bmatrix}. \end{aligned}$$

Since $\sum_i d\pi_i = 0$, equation 2 requires 1 of the following conditions to be true if $d\alpha_i = 0 \; \forall \; i$: 1) $d\pi_i = 0 \; \forall \; i$, or 2) $\boldsymbol{A_{1,1}} = \boldsymbol{A_{1,2}}$ and $\boldsymbol{A_{2,1}} = \boldsymbol{A_{2,2}}$. By setting the various elements of $\boldsymbol{A}$ equal to one-another, we can show that the latter condition is equivalent to the following:

$$\sum_{n=1}^{N} \left(\frac{\alpha_0 p_0^n}{\sum_j \pi_j \alpha_j p_j^n} \right)^2 = \sum_{n=1}^{N} \left(\frac{\alpha_1 p_1^n}{\sum_j \pi_j \alpha_j p_j^n} \right)^2 = \sum_{n=1}^{N} \frac{\alpha_0 p_0^n \alpha_1 p_1^n}{(\sum_j \pi_j \alpha_j p_j^n)^2} \tag{3}$$

Note that the preceding expression can only be solved if $\alpha_0 p_0^n = \alpha_1 p_1^n \; \forall \, n$. Thus, for a 2-dimensional state-space, we have shown that all elements of $d\boldsymbol{\alpha}$ will be 0 if and only if 1) all elements of $d\boldsymbol{\pi}$ are 0 or 2) $\alpha_0 p_0^n = \alpha_1 p_1^n \; \forall \; n$. Equivalently, it follows that $\alpha_0 p_0^n \neq \alpha_1 p_1^n$ for some n is a necessary condition for establishing the uniqueness of any non-trivial solution of $\boldsymbol{\pi}$ given $\boldsymbol{\alpha}$.

To complete the proof for the 2-dimensional case, it will suffice to show that for any constant $d\boldsymbol{\pi}$ vector having at least one non-zero element, at least one element of $d\boldsymbol{\alpha}$ will be strictly positive or strictly negative. We will demonstrate this by first taking the full derivative of equation 1. This yields,

$$\begin{aligned} \frac{N}{\alpha_i} d^2\alpha_i - \frac{N}{\alpha_i^2} (d\alpha_i)^2 = &- \sum_{n=1}^{N} \frac{\hat{p}_i^n}{\pi_i^2} d\pi_i \sum_j \left[\frac{\hat{p}_j^n}{\pi_j} d\pi_j + \frac{\hat{p}_j^n}{\alpha_j} d\alpha_j \right] + \\ &\sum_{n=1}^{N} \frac{\hat{p}_i^n}{\pi_i} \sum_j \left[\frac{\hat{p}_j^n}{\pi_j} d^2\pi_j - \frac{\hat{p}_j^n}{\pi_j^2} (d\pi_j)^2 + \frac{\hat{p}_j^n}{\alpha_j} d^2\alpha_j - \frac{\hat{p}_j^n}{\alpha_j^2} (d\alpha_j)^2 \right], \end{aligned}$$

where we define the parameter, $\hat{p}_i^n = \frac{\pi_i \alpha_i p_i^n}{\sum_j \pi_j \alpha_j p_j^n}$. If we assume all elements of $d\boldsymbol{\pi}$ to be constant and finite, and if we constrain all elements of $\boldsymbol{\pi}$ and $\boldsymbol{\alpha}$

to be greater than 0, then it follows from equation 1 that all elements of $d\boldsymbol{\alpha}$ must be finite. Under these conditions, $-\infty < d^2\alpha_i < \infty \; \forall \; i$. Thus, $d\boldsymbol{\alpha}$ must be continuous if $d\boldsymbol{\pi}$ is constant. For the case of a 2-dimensional state-space, we have already shown that if the elements of $d\boldsymbol{\pi}$ are non-zero, the elements of $d\boldsymbol{\alpha}$ *cannot* be zero unless $\alpha_0 p_0^n = \alpha_1 p_1^n \; \forall \; n$. It therefore follows that if $\alpha_0 p_0^n \neq \alpha_1 p_1^n$ for some n, one of the elements of $d\boldsymbol{\alpha}$ must always be strictly positive and the other must always be strictly negative if the elements of $d\boldsymbol{\pi}$ are constant and non-zero. Thus, we have shown that for a 2-dimensional state-space, $\alpha_0 p_0^n \neq \alpha_1 p_1^n$ for some n is a necessary *and* a sufficient condition for ensuring the uniqueness of any non-trivial solution of $\boldsymbol{\pi}$ given $\boldsymbol{\alpha}$.

For a state-space of size $M > 2$, we note that two or more states may always be combined into a single state having a single set of likelihoods and a single $\boldsymbol{\pi}$ and $\boldsymbol{\alpha}$ term. Thus, any state-space of size $M > 2$ may be converted into 1 or more state-spaces of size $M = 2$. For every possible 2-dimensional state-space, the same rules as before must hold to ensure the uniqueness of non-trivial solutions. Namely, $\alpha_0 p_0^n \neq \alpha_1 p_1^n$ for some n. By applying these rules to every possible 2-dimensional state space derived from a state space of size $M > 2$, it can easily be shown that $\alpha_i p_i^n \neq \alpha_j p_j^n$ for some i, j, and n is a necessary *and* a sufficient condition for ensuring that no more than one non-trivial solution exists for $\boldsymbol{\pi}$ given $\boldsymbol{\alpha}$ (assuming, of course, that $\alpha_i > 0 \; \forall \; i$). This completes our proof.

References

[1] Gold, B., and Morgan, N., *Speech and Audio Signal Processing*, John Wiley and Sons, Inc., New York, N. Y., 2000.

[2] Hermansky, H., and Morgan, N., *RASTA processing of speech*, IEEE trans. Speech Audio Process. 2: 578-589, 1994.

[3] Kung, S. Y., *Digital Neural Networks*, Prentice Hall, Englewood Cliffs, N. J., 1993.

[4] Norvig, P., and Russell, S., *Artificial Intelligence: A Modern Approach*, Prentice Hall, Englewood Cliffs, N. J., 1995.

[5] Sharma, S., Ellis, D., Kajarekar, S., Jain, P., and Hermansky, H., *Feature Extraction Using Non-Linear Transformation for Robust Speech Recognition on the Aurora Database*, Proc. ICASSP-2000, Istanbul, II-1117-1120.

[6] Rabiner, L., and Juang, B.-H., *Fundamentals of Speech Recognition*, Prentice Hall, Englewood Cliffs, N. J., 1993.

[7] Williams, G., and Renals, S., *Confidence measures from local probability estimates*, Computer Speech and Language, vol. 13, no. 4, Oct. 1999, pp. 395-411.

[8] Mangu, L., Brill, E., and Stolcke, A., *Finding consensus in speech recognition: word error minimization and other applications of confusion networks*, Computer Speech and Language, vol. 13, no. 4, Oct. 1999, pp. 395-411.