INTERNATIONAL COMPUTER SCIENCE INSTITUTE

1947 Center St. • Suite 600 • Berkeley, California 94704-1198 • (510) 666-2900 • FAX (510) 666-2956

SCALING UP: LEARNING LARGE-SCALE RECOGNITION METHODS FROM SMALL-SCALE RECOGNITION TASKS

Nelson Morgan, Barry Y Chen, Qifeng Zhu, Andreas Stolcke

TR-03-002

September 2003

Abstract

Despite the common wisdom that lessons learned from small experimental speech recognition tasks often do not scale to larger tasks, many important algorithms used in larger tasks were first developed with small systems applied to small tasks. In this paper we report experiments with the OGI Numbers task that led to the adoption of a number of engineering decisions for the design of an acoustic front end. We then describe a three-stage process of scaling to the larger conversational telephone speech (CTS) task. Much of the front end design required no change at all for the more difficult task, yielding significant improvements over our baseline front end.

1 Background

Small tasks (such as digit or number recognition) are commonly used in the development of novel algorithms. This is true even when larger tasks are the ultimate goal, simply because the larger tasks can require enormous computational resources for training or recognition, thus severely limiting the range of possible experiments that can be run. However, it is well known that huge improvements can sometimes be shown for small tasks that do not carry over to the larger tasks. Sometimes this is simply due to the fact that larger, more complex systems are generally used for the larger tasks, and other mechanisms are already in place to remove many of the grosser errors. Another complicating factor is that diagnostics can be more complete for small tasks, so that specific word or subword confusions can sometimes be focused on and handled; this micro-analysis can seldom be so helpful for large tasks. Finally, when experiments are quick to run, it is often tempting to conduct a large number of experiments with almost random settings, taking the best values as the "learned" parameters for an approach. Indeed this makes sense for the setting of some parameters given a development test set that has similar character to that of the final evaluation. However, for an ultimate evaluation on a completely different task, this may not be appropriate.

For all of these reasons, new techniques that have been shown to work on a small task may or may not improve performance for large-scale recognition. Nonetheless, it is also true that most techniques that have become "classic" methods for large-scale recognition were originally developed on much smaller tasks. Front end designs are the obvious example (PLP [5] or Mel cepstral analyses, as well as the various kinds of cepstral normalizations), but this has been observed in other areas as well (for instance the application of dynamic programming to speech recognition, both in deterministic and stochastic forms). This then raises the question: how can one determine whether a particular course of investigation using small-task evaluations is likely to bear fruit for a larger problem?

This paper reports on a set of experiments that may provide some insight on this question. Our goal was to improve conversational telephone speech (CTS) recognition by modifying the acoustic front end. We found that approaches developed for the recognition of natural numbers scaled quite well to two different levels of CTS complexity: recognition of utterances primarily consisting of the 500 most frequent words in Switchboard, and large vocabulary recognition of Switchboard conversations. We will describe the methods and results for these experiments, and will draw tentative conclusions about the nature of scalability for speech recognition methods.

2 Augmenting Conventional Features

For decades, the feature extraction component of speech recognition engines has consisted of some form of local spectral envelope estimation, typically with some simple transformation; current typical front ends consist largely of the Mel cepstrum or PLP computed from an analysis window of roughly 25 or 30 ms surrounding a central signal point, stepped along every 10 ms. A number of alternatives have been developed in recent years. One such approach, tandem acoustic modelling [6, 4, 3] uses a multilayer perceptron (MLP) to first discriminatively transform multiple feature vectors (typically PLP from 9 frames) before

using them as observations for Gaussian mixtures hidden Markov models (GMHMM). Thus, the neural network, which could be called a "feature net", incorporates around 100 ms of speech. We refer to the resulting feature net features as PLP/MLP features. Others have also tried incorporating longer temporal information yielding significant improvements in ASR performance (e.g., [10]).

The MLP is typically trained using phonetic targets. This approach works very well in matched training and test conditions, often achieving lower word error rates than systems without the discriminant nonlinear transformation provided by the MLP. However, in the case of mismatched training and testing conditions, researchers working on the Aurora task found it preferable to augment the original features with the feature net outputs, essentially using the concatenation of the original features and the PLP/MLP features as the front end for the GMHMM [1]. A similar approach was used in [13] where standard features were augmented by a complimentary source of information (in this case, estimates of formants from a mixture of Gaussians).

Another promising approach has been to combine the PLP/MLP features with features derived from the outputs of MLPs incorporating long-time log critical band energy trajectories (500 ms - 1 s) [7, 8]. The set of these MLPs forms the TRAPS system, named as such because the system learns discriminative TempoRAl Patterns (TRAPS) in speech. MLPs in the TRAPS system are also trained with phonetic targets. These features are complementary to the 100 ms span of the feature net MLP, and we have observed that systems using the combination of the two feature sets perform better than systems using either feature type alone.

The approaches listed above were developed on small tasks, i.e. connected digits, continuous numbers, and TIMIT phone recognition, where the training and test set was small in terms of vocabulary as well as data size. We tested systems that incorporated the novel approaches listed above in tasks of varying complexity. We used conventional front end features (PLP in this case) augmented with the combination of PLP/MLP features and TRAPS features. We used three different temporal resolutions. The original PLP features were derived from short term spectral analysis (25 ms time slices every 10 ms), the PLP/MLP features used 9 frames of PLP features (100ms), and the TRAPS features used 51 frames of log critical band energies (500ms).

In all of the experiments we performed, our baseline feature vector consisted of 12th order PLP coefficients plus energy computed over a 25 ms frame window every 10 ms. 1st and 2nd order deltas were calculated and appended together to yield a 39 dimensional baseline feature. We also used mean and variance normalization per conversation side.

As contrast, we augment the baseline PLP features with a combination of the two probability-based feature streams: PLP/MLP features and TRAPS features. For the first stream, we trained discriminative feature net MLPs using 9 consecutive frames of the baseline PLP features as inputs and 47 phoneme targets generated from forced alignments using the SRI recognizer. For the second stream, the first stage TRAPS MLPs took PCA transformed log critical band energy trajectories formed by taking 51 consecutive frames of log critical band energies every 10ms. These critical band MLPs were trained with the same phoneme targets as in the feature net MLP. A merger MLP (trained with these same phoneme targets) combined the critical band MLPs' outputs and gave one estimate of phoneme posteriors every 10 ms.

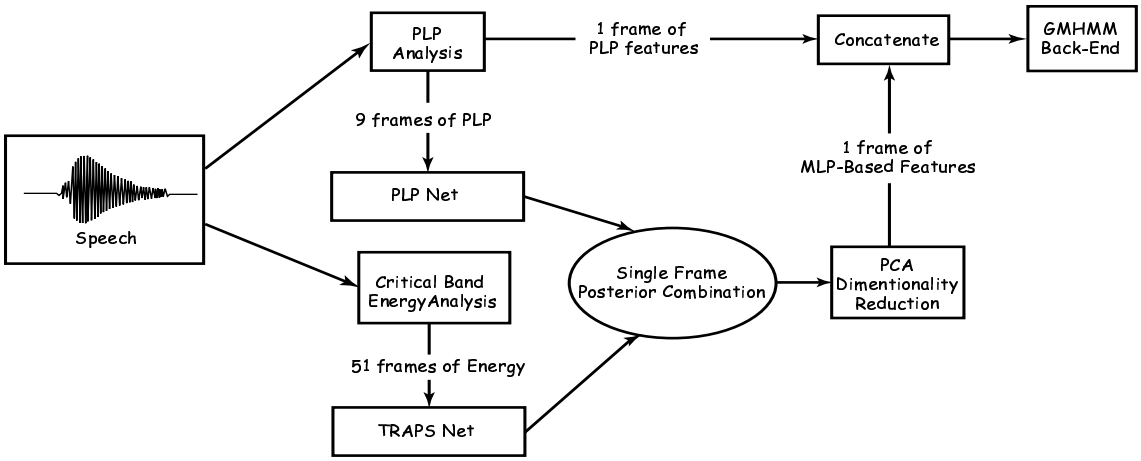

Figure 1: *Augmenting PLP Front End Features*

Since both the output of the TRAPS classifier and the feature net can be interpreted as posterior probabilities of the 47 phonemes, we could combine them using frame-wise posterior probability combination techniques [9, 11] (described briefly below). After combination, we took the log of the posterior vector to make it more Gaussian, and then orthogonalized and reduced the dimensionality of the posterior vector using PCA. The resulting variables were then appended to the original PLP cepstra to form the extended feature vector. Figure 1 summarizes this augmentation process.

In what follows, we refer to these augmented features as PLP+*combomethod*(*Streams*) features, where *combomethod* can be one of three frame-wise posterior combination methods: the average of the posteriors combination (AVG); the average of log posteriors combination (AVGLog), and finally, the inverse entropy weighted combination (INVENT) [11]. *Streams* refers to the PLP/MLP feature stream and TRAPS feature stream. The first two combination methods essentially assume that each MLP feature stream is equally important, while the entropy-based combination assumes that the MLP feature with lower entropy is more important than an MLP feature with high entropy. This is intuitively correct, since a low entropy posterior distribution (such as would occur with a high single peak) implies strong confidence in class identity. Generally, the combined posterior can be written as:

$$\tilde{P}(q_k|x) = \omega_1 \tilde{P}_1(q_k|x^1) + \omega_2 \tilde{P}_1(q_k|x^2) \quad (1)$$

where $\tilde{P}_1(q_k|x^1)$ and $\tilde{P}_2(q_k|x^2)$ are the posteriors (or log posterior in the log posterior average) of different MLPs for the same frame. For the average combination, ω_1=ω_2=0.5. For entropy-based posterior combination, ω is the inverse entropy computed over one frame for an MLP output and normalized so that the sum of all weights is one. A threshold of 1 is applied on frame entropy. If the entropy for a frame from an MLP is greater than 1, it is set to a large value (e.g., 10000) so that the weight for the stream is a very small number.

The features described above served as the front end features for our recognition experiments. We used a stripped down version of SRI's Hub-5 conversational speech transcription system for our HMM backend. In particular, the backend that we used was similar to the

first pass of the system described in [12], using a bigram language model and within-word triphone acoustic models.

3 Stage 1: Numbers Task

As noted previously, all the basic techniques employed here were originally developed using quite small tasks. In particular, prior to the experiments reported here, the MLP-based feature transformations, the temporal features called TRAPS, and the methods used to combine them were all trained and tested on a number of smaller tasks including the OGI Numbers task (the Numbers95 corpus). In these earlier Numbers experiments, Numbers data was used for both training and test, and so the scalability of the typically impressive results was unknown.

On the other hand, simply taking the features and applying them to a large task risked failure without obvious diagnostic potential. Consequently, we designed a three-stage approach to the scaling process. Our initial step was to train on a combination of CTS data and read speech, and then test on OGI Numbers.

3.1 Numbers Task Description

The training set for this stage was an 18.7-hour subset of the old "short" SRI Hub training set. 48% of the training data was male and 52% female. 4.4 hours of this training set comes from English CallHome, 2.7 hours from Hand Transcribed Switchboard, 2.0 hours from Switchboard Credit Card Corpus, and 9.6 hours from Macrophone (read speech).

We divided the entire OGI Numbers corpus into three sets. One was used for system parameter tuning, one for development testing, and another for final testing. We report our results on the test set which contains 1.3 hours of speech (2519 utterances and 9699 word tokens).

3.2 Results on Numbers Task

Using the training set defined above, we trained triphone gender-independent HMMs using the SRI speech recognition system. We also trained a gender-independent PLP/MLP feature net as well as a gender-independent TRAPS system. Although the recognition task was numbers, the HMMs were trained for broader coverage. Thus we hoped that the conclusions reached with this training data might generalize better to other tasks. The testing dictionary contained thirty words for numbers and two words for hesitation, and we used a simple bigram language model trained on our Numbers tuning set.

For all of the experimental systems in this paper, truncation of the PCA output (that is, eliminating some low-variance components) was critical to performance. Keeping the top 17 dimensions seemed to be the optimal length on all of our tuning data. The truncated PCA output was then appended with PLP features as the augmented feature. It is likely that the small variance components may not have been useful because PLP was already fairly good for phonetic classification (particularly for low-noise speech). In contrast, when MLP-based features were used alone as the front end feature, often no truncation was needed after PCA to achieve the best performance.

System	Numbers Test Set WER	Relative Reduction WER
PLP Baseline	4.0%	-
PLP+AVG(*Streams*)	3.3%	17.5%
PLP+AVGLog(*Streams*)	3.2%	20.0%
PLP+INVENT(*Streams*)	3.3%	17.5%

Table 1: Word error rate (WER) and relative reduction of WER on Numbers using different combination approaches. *Streams* denotes the PLP/MLP feature stream and the TRAPS feature stream.

We incorporated PLP/MLP and TRAPS features by frame-wise posterior combination. The combined features were then reduced in dimension to 17 using PCA and concatenated to the baseline PLP features to create an augmented feature vector of dimension 56. As noted previously, we used several frame-wise posterior combination methods: the average of posteriors PLP+AVG(*Streams*), the average of log posteriors PLP+AVGLog(*Streams*), and the inverse entropy weighted combination PLP+INVENT(*Streams*) (see Table 1). All three performed roughly the same, achieving a 17-20% relative reduction in word error rate.

These experiments showed that the combination of the three features can improve the recognition performance over using any feature alone. On the other hand, all the approaches to posterior combination were roughly equivalent in this case. These preliminary conclusions would later be tested on tasks of increasing complexity.

4 Stage 2: The 500 Word CTS Task

Our methods continued to work well on the small vocabulary continuous numbers task even when we did not train explicitly on continuous numbers. Before applying our approaches to the full vocabulary Switchboard task, we considered a second stage task, that of recognizing the 500 most common words[1] in Switchboard I. There were several advantages to using this intermediate task. First, since the recognition vocabulary consisted of common words from Switchboard, it was likely that error rate reduction would apply to the larger task as well. Second, there were many examples of these 500 words in the training data, so less training data was required than would be needed for the full task. This in turn sped training time accordingly. Lastly, decoding complexity in this task was smaller, which also sped experimental turn-around time.

4.1 500 Words Task Description

For training, we created a subset of the "short" training set used at SRI for CTS system development, which we referred to as the Random Utterances of Short Hub or the RUSH set.

[1]This task was proposed by our colleague George Doddington.

This RUSH set consisted of utterances from 217 female and 205 male speakers, which was the same number of speakers as the short CTS training set, but which contained one third of the total number of utterances. The female speech consisted of 0.92 hours from English CallHome, 10.63 hours from Switchboard I with transcriptions from Mississippi State [2], and 0.69 hours from the Switchboard Cellular Database. The male speech consisted of 0.19 hours from English CallHome, 10.08 hours from Switchboard I, 0.59 hours from Switchboard Cellular, and 0.06 hours from the Switchboard Credit Card Corpus.

The 500 word test set was a subset of the 2001 Hub-5 evaluation data. Given the 500 most common words in Switchboard I, we chose utterances [2] from the 2001 evaluation data in which 90% or more of the words in the utterance were on the word list. In other words, we allowed at most 10% of the words in an utterance to be out of vocabulary (OOV) words. 49.6% of the utterances in the 2001 evaluation data met this requirement, and the total OOV rate was 3.2%. We then partitioned this set into a tuning set (0.97 hours, 8242 total word tokens) and a test set (1.42 hours, 11845 total word tokens). We used the tuning set to tune system parameters like word transition weight and language model scaling, and we determined word error rates on the test set. The language model used in both the 500 word task as well as the full vocabulary task was the first-pass bigram language model used by SRI for the large vocabulary evaluations in 2000.

4.2 Results on Top 500 Words Task

Using the baseline PLP features, we trained gender dependent triphone HMMs on the 23 hour RUSH training set, and then tested this system on the 500 word test set achieving a 43.8% word error rate (see Table 2 which shows the word error rates of our various systems on the top 500 word test set). As seen in the table, the word error rate was reduced 10% relative by augmenting the baseline features with the combined PLP/MLP and TRAPS features. In this case, we trained gender dependent PLP/MLP feature nets and TRAPS systems.

System	500 Word Test Set WER	Relative Reduction WER
PLP Baseline	43.8%	-
PLP+AVG(*Streams*)	39.4%	10.0%
PLP+AVGLog(*Streams*)	39.5%	9.8%
PLP+INVENT(*Streams*)	39.2%	10.5%

Table 2: Word error rate (WER) and relative reduction of WER on the top 500 word test set of systems trained on the RUSH set using different combination approaches. *Streams* denotes the PLP/MLP feature stream and the TRAPS feature stream.

All three combination methods performed roughly the same. Even though the more

[2] An utterance is defined to be a string of words separated by less than 0.3sec, and greater than 0.3 seconds of separation at the beginning and end.

complicated inverse entropy combination technique performed only slightly better than the simple average combination methods, both styles have their appeal. The averaging methods are certainly simple, and don't rely on any estimation method. On the other hand, the inverse entropy combination technique is potentially robust to poor classifier streams. We experienced this property for one of our later (CTS) experiments. Due to a bug in our procedures, we unintentionally combined a badly degraded TRAPS stream with the other features using both methods. When probabilities were multiplied or added without weights, the degraded stream hurt performance badly, as one might expect. On the other hand, the inverse entropy-weighting reduced the importance of the poor stream so that the overall performance essentially matched what we had for a feature that consisted of the baseline PLP features concatenated with the PLP/MLP feature alone. Thus, the entropy-based approach to combination appears to be more robust to unexpectedly poor streams. We expect that this property might be particularly useful for future efforts in which we might combine a larger number of streams where some streams may sometimes provide less useful information.

5 Stage 3: Full CTS Vocabulary

Having seen how our approaches scaled with increasing test set complexity, we applied these approaches to the third and last stage: full vocabulary CTS task.

5.1 Full CTS Task Description

We tried using our previously defined RUSH training set for this task and found it inadequate for training given the increase in vocabulary. In other words, error rates on Switchboard test sets were unacceptably high for the RUSH training set. Instead, we used SRI's entire "Short" CTS training set from which RUSH was derived. This set contained a total of 68.95 hours of CTS. 2.75 hours of English CallHome, 31.30 hours from Mississippi State transcribed Switchboard I, and 2.03 hours of Switchboard Cellular form the data from female speakers. The male speaker data came from 0.56 hours of English CallHome, 30.28 hours from Switchboard I, 1.83 hours from Switchboard Cellular, and 0.20 hours of Switchboard Credit Card Corpus. As in the 500 word task, we trained triphone gender dependent HMMs as well as gender dependent PLP/MLP feature nets and TRAPS systems.

For testing, we used the 2001 Hub-5 Switchboard evaluation set. This evaluation set contains a total of 6.33 hours of speech, 62890 total word tokens. For tuning our system parameters, we used a subset of the 2001 Hub-5 development set.

5.2 Results on Full CTS Task

The baseline system achieved a 43.8% word error rate on the Hub-5 evaluation 2001 set (see Table 3 which shows the word error rates of our various systems on the 2001 Hub-5 evaluation set). The augmented features reduced the error rate by about 7% relative. For this task, there was a small penalty for the AVGLog combination method in comparison to the other approaches.

System	Hub-5 EVAL2001 WER	Relative Reduction WER
PLP Baseline	43.8%	-
PLP+AVG(*Streams*)	40.5%	7.5%
PLP+AVGLog(*Streams*)	41.0%	6.4%
PLP+INVENT(*Streams*)	40.6%	7.3%

Table 3: Word error rate (WER) and relative reduction of WER on the 2001 Hub-5 evaluation set of systems trained on SRI's "Short" CTS training set using different combination approaches. *Streams* denotes the PLP/MLP feature stream and the TRAPS feature stream.

6 Discussion and Conclusions

The PLP/MLP and the TRAPS features, developed for a very small task, were then applied to successively larger problems. As we had hoped,

1. Word error rate was significantly reduced for the larger tasks as well, and
2. The combination methods, which gave equivalent performance for the smaller task, were also comparable on the larger tasks.

Regarding the first point, an absolute error rate reduction of over 3% on Switchboard is quite significant. However, the typical relative reduction in error is somewhat smaller for the larger tasks (ranging from 20% on the smallest task to 7% on the largest one). Thus, error rate reduction may scale, but the degree of improvement may not without further work using the larger task. Nonetheless, even 7% relative improvement is often of interest for larger tasks like CTS. For such tasks, sizeable improvements are typically only obtained by a combination of many small innovations.

The second observation seems to be unequivocally confirmed in these three stages of experiments - we observed no consistent (scalable) advantage to using any of the three chosen methods for combining posteriors as part of the process of generating probability-based front end features. On the other hand, as noted earlier, the inverse entropy method appears to be quite robust to catastrophic degradations of feature streams. We also should emphasize the limitation of this experiment, in which we were only combining two streams, both of which were fairly effective for phonetic discrimination. If we begin to use a significantly larger number of streams, some streams will be more likely to be ineffective at least some of the time, and a dynamic weighting method like the inverse entropy approach may show a clearer advantage. This view seems to be supported by earlier work at IDIAP [11]

For optimal performance, there are always detailed aspects of the analysis that should be adjusted (if possible) when scaling to a new task. For instance, as noted earlier, our experiments with Numbers suggested that dimensionality reduction via PCA was critical to combining the new features with the baseline PLP, and the optimal number of dimensions turned out to be 17. We would expect that this number would vary depending on many

factors, such as the informative nature of the streams, the nature of the models, and features of the recognition engine such as the exponentiation of the Gaussian component likelihoods that is done in the SRI system; this latter feature can be viewed as compensating for variations in the feature dimension.

The PLP/MLP and TRAPS features that were used here were significantly different from the baseline features. In such cases, we would expect that major conclusions about their use would apply to a range of tasks, and this was largely supported by our experiments. On the other hand, it is clear that further optimization of performance can be achieved by work with the larger task. This is always true, but the ability to bring some of the performance improvements forward following on work with smaller tasks is extremely important for speeding the development of novel approaches. Our experience suggests that providing intermediate tasks as "stepping stones" can greatly aid the scaling of techniques from small to large tasks.

7 Acknowledgments

We would like to gratefully acknowledge all the people who helped provide various components for our system: Sunil Sivadas for all his help in providing the TRAPS features for the 500 word and full vocabulary tasks; Pratibha Jain, Hynek Hermansky, and Frantisek Grezel for the TRAPS features in the Numbers task; Hemant Misra for providing inverse entropy combination tools; Ozgur Cetin for creating the training set for the Numbers task; and George Doddington for finding the top 500 words in Switchboard and creating the top 500 word subset from the 2001 Hub-5 evaluation set. All of the other members of our EARS Novel Approaches team (at SRI, OGI, IDIAP, Columbia, and the University of Washington) also contributed intellectually to this work. Finally, this work was made possible by funding from the DARPA EARS Novel Approaches Grant: No. MDA972-02-1-0024.

References

[1] C. Benitez, L. Burget, B. Chen, S. Dupont, H. Garudadri, H. Hermansky, P. Jain, S. Kajarekar, and S. Sivadas. Robust ASR front-end using spectral-based and discriminant features: experiments on the Aurora tasks. In *Proc. Eurospeech-2001*, Denmark, September.

[2] N. Deshmukh, A. Ganapathiraju, A. Gleeson, J. Hamaker, and J. Picone. Resegmentation of Switchboard. In *Proc. ICSLP-1998*, pages 1543–1546, Sydney, Australia, November.

[3] D.P.W. Ellis and M.J. Reyes Gomez. Investigations into tandem acoustic modeling for the Aurora task. In *Proc. Eurospeech-2001, Special Event on Noise Robust Recognition*, Denmark, September.

[4] D.P.W. Ellis, R. Singh, and S. Sivadas. Tandem acoustic modeling in large-vocabulary recognition. In *Proc. ICASSP-2001*, Salt Lake City, May.

[5] H. Hermansky. Perceptual linear predictive (PLP) analysis for speech. *Journal of the Acoustical Society of America*, 87:1738–1752, April 1990.

[6] H. Hermansky, D.P.W. Ellis, and S. Sharma. Tandem connectionist feature stream extraction for conventional HMM systems. In *Proc. ICASSP-2000*, volume III, pages 1635–1638, Istanbul.

[7] H. Hermansky and P. Sharma, S.and Jain. Data-derived nonlinear mapping for feature extraction in HMM. In *Proc. ICASSP-2000*, Istanbul.

[8] H. Hermansky and S. Sharma. Temporal patterns (TRAPS) in ASR of noisy speech. In *Proc. ICASSP-1999*, Phoenix, March.

[9] A. Janin, D. Ellis, and N. Morgan. Multi-stream speech recognition: Ready for prime time? In *Proc Eurospeech-1999*, volume II, pages 591–594, Budapest.

[10] B. Milner. Inclusion of temporal information into features for speech recognition. In *Proc. ICSLP-1996*, pages 256–259.

[11] H. Misra, H. Bourlard, and V. Tyagi. New entropy based combination rules in HMM/ANN multi-stream ASR. In *Proc. ICASSP-2003*, Hong Kong.

[12] A. Stolcke, H. Bratt, J. Butzberger, H. Franco, V. R. Rao Gadde, M. Plauche, C. Richey, E. Shriberg, K. Sonmez, F. Weng, and J. Zheng. The SRI March 2000 Hub-5 conversational speech transcription system. In *Proc. NIST Speech Transcription Workshop*, College Park, MD, 2000.

[13] M. N. Stuttle and M. J. F. Gales. A mixture of Gaussians front end for speech recognition. In *Proc. Eurospeech-2001*, Denmark, September.