STUDIES IN TRANSFORMATION-BASED ADAPTATION

Venkatesh Nagesha *Larry Gillick*

Dragon Systems, Inc.
Newton, MA 02160

ABSTRACT

This paper studies the use of transformation-based speaker adaptation in improving the performance of large vocabulary continuous speech recognition systems. We present a formulation of the adaptation procedure that is simpler than existing methods. Our experiments demonstrate that speaker normalization continues to be important even after significant amounts of speaker adaptation. An automatic clustering algorithm is compared to human expertise in sorting output distributions into collections that share the same transformation. We quantify improvements over standard Bayesian (by maximum a posteriori or MAP) adaptation in terms of (a) speed of adaptation, and (b) robustness to transcription errors. Finally, we discuss the use of speaker transformations in the training process.

1. INTRODUCTION

Much of the promise of large vocabulary continuous speech recognition (LVCSR) remains unfulfilled because the current speaker independent (SI) systems do not meet speed/accuracy requirements. One approach to improving performance is to limit system usage to one person and provide customized models for each user. This is usually accomplished by tuning a SI system based on speech material (adaptation data) acquired from a new user. Desirable characteristics for speaker adaptation methods include (a) ability to adapt to a new speaker rapidly and effectively, (b) computational efficiency, and (c) robustness against deviations from the assumed framework. While several strategies for speaker adaptation have been studied [3, 4], an approach that uses linear regression principles to transform the acoustic models seems particularly promising [1, 2, 5].

A simple strategy for speaker adaptation is to update the model parameters of a speech unit (e.g. phoneme) every time the new speaker utters the unit. The main difficulties with such an approach are that large numbers of observations are needed from each new speaker so as to (a) cover the entire collection of speech units and update them, and (b) obtain reasonable estimates of the model parameters for each unit. Given enough (and reliable) speech material, this method often leads to high performance systems. Although variants of this method abound, they are generically called Bayesian or maximum a posteriori (MAP) in the literature for they can be interpreted as a way to combine apriori information (SI models) and observed data [4]. Having noted the requirements of the MAP algorithm, an effective strategy may be to build more global strategies for updating model parameters, i.e., use observations from a speech unit to update several similar-sounding speech units. One approach is to use the decision-tree (that is used to cluster triphones) to update multiple leaves simultaneously [6]. A second (and more global) approach is to use transformation techniques to simultaneously update a group of model parameters. The transformation is derived from linear regression principles; and involves fitting a linear model between observations from the new speaker and the SI acoustic models [2, 5].

The goal of our paper is to investigate a number of issues in the use of transformation-based speaker adaptation for improving the performance of LVCSR systems. The paper is organized as follows. We begin by reviewing our formulation of the regression model and the overall system configuration. Our formulation is similar to [5] but is simpler in that (a) it is based on sufficient statistics rather than raw observations, and (b) it requires only one matrix inversion per transform. One might wonder whether it is still useful to perform speaker normalization [8] in addition to extensive speaker adaptation. Experiments in section 3 indicate that it provides an additional 5–7% improvement. When large amounts of adaptation data are available, it is possible to train more than one transformation. The problem, then, is to decide which speech units share the same transformation. Section 4 addresses the issue of whether this decision is best made by human linguistic expertise or by an automatic clustering algorithm. Since we update a collection of probability density functions (PDFs), rather than an individual PDF (as in MAP), we might expect that the procedure (a) is more robust to estimation errors arising from inaccurate transcriptions of the adaptation data, and (b) provides faster adaptation than MAP. Section 5 addresses this issue by comparing the two methods for supervised and unsupervised adaptation. Finally, we examine the use of speaker transformations in the training process. In typical model-building approaches, the baseline (SI) model attempts to represent a large population of speakers. While we attempt to retain within-speaker variations relevant to speech events and reduce across-speaker fluctuations (via channel normalization, linear discriminant analysis, frequency warping, etc.), we are not entirely successful. The successful use of transformation techniques indicates that simple linear models provide a reasonable means of describing individual speaker characteristics. It raises the issue of applying

similar mappings during training. If baseline acoustic models (representing any speaker) can be transformed to match a new speaker, perhaps we could apply another (*the dual*) transformation to the speaker's data so that all the training speakers "look alike." The idea then is that data from each training speaker will be converted to a *normalized* form, and SI models (built in the usual way from such normalized data) upon transformation will provide good models for each new speaker.

2. PROBLEM FORMULATION

Given a collection of new data and a set of baseline acoustic models, we run Baum-Welch adaptation (i.e., the EM algorithm) to produce, for each mixture component j with a K–dimensional mean vector, $\mathbf{x}[j]$, (a) an average ($\mathbf{y}[j]$) of all frames (probabilistically) assigned to the component, and (b) the sum of fractional frames assigned ($N[j]$). We can think of $\mathbf{y}[j]$ and $N[j]$ as being sufficient statistics derived from frames of observed data, $\mathbf{z}[n]$. Let $\mathcal{C}$ be a collection of components for which the new data is assumed to share the same transformation. We assume

$$\mathbf{y}[j] = \mathbf{A}\mathbf{x}[j] + \mathbf{e}[j] \ , \ \text{for all } j \in \mathcal{C} \ ,$$

where $\mathbf{e}[j] \sim \mathcal{N}(\mathbf{0}, \mathbf{I}\sigma^2/N[j])$ describes the regression error.[1] The scaling by $N[j]$ reflects the averaging done to calculate the statistic, $\mathbf{y}[j]$. Our assumption that the error covariance is a scaled identity matrix stems from our use of linear discriminant analysis, which forces the average within-class covariance of the feature vector to have that form. The maximum likelihood estimate (MLE) for the regression matrix, $\mathbf{A}$, is obtained by solving a weighted least squares problem, and is given by

$$\hat{\mathbf{A}}^T = \left[\sum_{j=1}^{N_c} N[j]\,\mathbf{x}[j]\,\mathbf{x}^T[j]\right]^{-1} \left[\sum_{j=1}^{N_c} N[j]\,\mathbf{x}[j]\,\mathbf{y}^T[j]\right] \ , \quad (1)$$

where N_c is the number of components in the collection $\mathcal{C}$. It requires a single $K \times K$ matrix inversion, where K is the number of features in the observation stream. We considered two other possibilities for modeling the regression error covariance. For an arbitrary (but known) covariance, $\mathbf{\Sigma}[j] = \mathcal{E}\left(\mathbf{e}[j]\,\mathbf{e}^T[j]\right)$, the MLE is

$$\text{vec}\left(\hat{\mathbf{A}}^T\right) = \left[\sum_{j=1}^{N_c} \left(\mathbf{x}[j]\,\mathbf{x}^T[j]\right) \otimes \mathbf{\Sigma}^{-1}[j]\right]^{-1} \quad (2)$$

$$\text{vec}\left(\sum_{j=1}^{N_c} \mathbf{x}[j]\mathbf{\Sigma}^{-1}[j]\mathbf{y}^T[j]\right) , \quad (3)$$

where $\otimes$ denotes the Kronecker product and $\text{vec}(\mathbf{T})$ denotes the column-by-column concatenation of a matrix $\mathbf{T}$. This requires inverting a single matrix of size $K^2 \times K^2$. For diagonal covariances, this solution can be simplified [5], and would involve the inversion of K matrices, each of size $K \times K$. Both assumptions were deemed unattractive because (a) they are computationally expensive and (b) a priori calculations of the required *average within-speaker covariances,* $\mathbf{\Sigma}[j]$, are cumbersome. In particular, the *model variances used in [5] are biased* in that they include within-speaker and across-speaker variances. At present none of the methods include the (linear) modeling error covariance which will dominate the error descriptions when there is a large-enough sample size.

2.1. System Description

A portion (about 24 hours) of the SI37 (25 WSJ1 speakers) training data was selected to build the baseline (speaker independent) models. A separate portion of SI37 consisting of 12 WSJ0 speakers was chosen as the adaptation corpus. Each of these 12 speakers has up to 600 sentences (nearly 80 mins) available for adaptation. The WSJ0 (Nov. 92) speaker-dependent 5K dev test material was used as the test corpus and has about 40 sentences from each of the 12 adaptation speakers. Our acoustic modeling is based on decision-tree clustering of triphones and uses mixture Gaussian distributions to represent leaves in the tree [7, 8]. An acoustic front-end produces 36 features (based on channel-normalized mel cepstra) per 10 ms frame and linear discriminant analysis reduces them to $K = 24$ features/frame. Decision-tree clustering and a combination of the k–means and EM algorithms are used to produce the final acoustic models. Experiments using two sets of gender-independent acoustic models were conducted. System I uses 6000 leaves in the decision tree with 4 Gaussians to model the PDF for each leaf. System II uses a slightly smaller set of acoustic models (nearly 4800 leaves and 4 Gaussians per leaf). The recognition engine uses a time-synchronous decoder with Viterbi beam-pruning for the detailed match and a tree-search for the rapid match. We used a DARPA standard WSJ 5K vocabulary with word bigrams for the language model. The performance of the baseline models with no adaptation data is 15.3% for System I and 16.2% for System II. The error rates are higher than our state of the art systems primarily because we have used limited amounts of training data, our acoustic models are small and gender-independent, and we have used bigram language models.

3. INTERACTION OF SPEAKER ADAPTATION AND SPEAKER NORMALIZATION

We first address the issue of combining adaptation with speaker normalization by frequency warping [8] *since both techniques attempt to exploit speaker-specific variability* in the data. Recall that [5] does not use speaker normalization. The word error rates for System I are given in Table 1, speaker normalization appears to yield about 12% improvement before adaptation, and 5%–7% improvement after adaptation.

[1] In theory, the regression error consists of two terms, (a) $\boldsymbol{\epsilon}[j]$, the sampling error in $\mathbf{y}[j]$ whose variance is inversely proportional to $N[j]$ and can be reduced to zero, and (b) $\boldsymbol{\delta}[j]$, the error incurred by the linear model assumption, which does not depend on the amount of adaptation data. In particular, $\boldsymbol{\delta}[j]$ is difficult to characterize and takes on greater significance as the sample size increases. We did not account for $\boldsymbol{\delta}[j]$ in this study but hope to address this in the future.

Spkr. Norm. data	NO	YES
mins. of adapt. data	error rate	error rate
0.0	15.3%	13.4%
6.5	12.9%	12.1%
13.0	12.1%	11.6%
26.0	12.1%	11.4%
78.0	12.0%	11.4%

Table 1. Combining speaker normalization and rapid adaptation, experiments on System I using 30 transforms

4. CHOOSING THE COLLECTIONS $\mathcal{C}$

We have investigated two methods for determining the set of components that share a common transformation. The first is to use linguistic information about phonetic similarities to cluster individual mixture components at the state level. The second method is data-driven whereby we cluster component means in the baseline models using a k–means type algorithm. This method tends to cluster components of highly dissimilar phonemes that happen to look similar (in a Euclidean-distance sense), but, unlike the first method, large numbers of collections (say > 100) are easy to build. The word-error rates for System II are given in Table 2, a hand-selected collection of $N_c = 30$ transformations seem to perform as well or slightly better than 30, 50, and 100 collections obtained from the automatic clustering approach. We hope to combine these ideas by including penalties in our clustering algorithm so that dissimilar phonemes are discouraged from sharing a transformation.

mins. of	$N_c = 30$		$N_c = 50$		$N_c = 100$
adapt.	(i)	(ii)	(i)	(ii)	(ii)
6.5	12.9%	13.1%	13.1%	12.8%	18.3%
13.0	12.2%	13.0%	12.3%	12.3%	12.6%
19.5	12.3%	12.6%	12.2%	12.2%	12.3%
39.0	12.2%	12.5%	12.6%	12.3%	12.2%
78.0	11.9%	12.8%	11.8%	12.3%	12.0%

Table 2. Approaches to choosing collections $\mathcal{C}$ using (i) knowledge-based, and (ii) data-driven strategies

5. ROBUST ADAPTATION

We study the overall issue of robustness in the adaptation procedure by comparing this procedure to standard MAP adaptation. Table 3 reports the error rates with and without supervision of the adaptation material. The transcripts, used in the unsupervised experiments, were produced by using acoustic models used in System I, and a DARPA standard WSJ 20K vocabulary with word bigrams for the language model. The error rate on the recognized transcript of the adaptation material is about 21.5%, of which about 4.5% errors are attributable to words not present in the 20K recognition vocabulary. Table 3 reports results on running one (T–1) and three (T–3) iterations of transformation-based adaptation, standard MAP adaptation, and one iteration of the transformation-based adaptation followed by a single pass of MAP adaptation. The lack of supervision in the adaptation data leads to a loss of about 0.6%–0.8% accuracy for the transformation-based adaptation, while the loss is about 1.4%–1.6% for MAP adaptation. It suggests that the transformation approach is less sensitive to transcription errors. Combining the two methods leads to slightly higher accuracy, and lower sensitivity to transcription errors, when compared to MAP adaptation, especially for small amounts of data. Also, notice that running 3 iterations of (1) produces a slight improvement over 1 iteration. Another typical aspect of the adaptation is that the error rate goes down rapidly for the first few minutes of data from a new speaker with little or no improvement as significantly more amounts of speech become available. Presumably, the inability of transformation-based adaptation to effectively utilize large amounts of data stems from modeling error incurred in sharing transformations over many mixture components. In contrast, MAP adaptation is slower and more gradual, and leads to better performance if large amounts of data are provided.

mins. of	supervised			
adapt. data	T–1	T–3	MAP	T–1 + MAP
6.5	12.9%	12.6%	13.7%	12.7%
13.0	12.2%	12.1%	12.1%	11.5%
67.0	12.0%	11.5%	10.4%	10.4%
78.0	11.9%	11.5%	10.4%	10.4%

mins. of	unsupervised*			
adapt. data	T–1	T–3	MAP	T–1 + MAP
6.5	13.7%	13.6%	15.1%	13.4%
13.0	13.1%	12.9%	13.7%	12.7%
67.0	12.6%	12.2%	11.9%	11.7%
78.0	12.7%	12.4%	12.0%	11.5%

Table 3. Word error rates for MAP and transformation approaches in System II (*transcription error rate $\sim 21.5\%$)

6. TRANSFORMATIONS DURING TRAINING

The standard SI training process involves (a) probabilistic assignment of each frame of training data to individual mixture components, and (b) estimation of the output PDF parameters. We now attempt to incorporate speaker transformations into the model building process and highlight changes in each of the two steps. The basic philosophy is as follows. Let $\mathbf{x}_j$ denote the "mean" of mixture component j such that when transformed by $\mathbf{A}_j[l]$ (the transformation matrix for speaker l and component j), $\mathbf{A}_j[l]\,\mathbf{x}_j$ is a good representation of observations from speaker l. We would then choose the means $\{\mathbf{x}_j\}$ such that, for each training speaker l with transformations $\{\mathbf{A}_j[l]\}$, the transformed means $\{\mathbf{A}_j[l]\,\mathbf{x}_j\}$ effectively model the acoustic space spanned by training data for speaker l.

We run the Baum-Welch algorithm on data from training speaker l, using current models for speaker l, to pro-

duce our sufficient statistics (mean vector $\mathbf{y}_j[l]$, variances $\left\{\sigma^2_{jk}[l]\right\}_{k=1}^{K}$, and fractional frame counts $N_j[l]$) for each mixture component j with mean vector $\mathbf{x}_j$. Let $\mathbf{A}[l]$ denote the transformation for training speaker l and for a set of components, $j \in \mathcal{C}$, i.e. $\mathbf{y}_j[l] = \mathbf{A}[l]\mathbf{x}_j + \mathbf{e}_j$, for all $j \in \mathcal{C}$, and $\mathbf{e}_j \sim \mathcal{N}(\mathbf{0}, \mathbf{I}\sigma^2/N_j[l])$ as in section 2. The MLE of the model means (for a fixed value of $\mathbf{A}[l]$) is obtained by solving a weighted least squares problem, and is given by

$$\hat{\mathbf{x}}_j = \left(\sum_{l=1}^{N_s} N_j[l]\mathbf{A}^T[l]\mathbf{A}[l]\right)^{-1} \left[\sum_{l=1}^{N_s} N_j[l]\mathbf{A}^T[l]\mathbf{y}_j[l]\right] . \quad (4)$$

The model variances and mixture weights are then updated using the standard reestimation procedure. An interesting simplification of (4) is to use

$$\tilde{\mathbf{x}}_j^{(i)} = \left(\sum_{l=1}^{N_s} N_j[l]\right)^{-1} \left[\sum_{l=1}^{N_s} N_j[l]\mathbf{A}^{-1}[l]\mathbf{y}_j[l]\right] , \quad (5)$$

which corresponds to applying the inverse transformation to each training speaker's mean and then applying the standard reestimation procedure. The mean update in (4) is similar to that in [1], one difference being in the use of the model variances. This is analogous to the similarity between our estimate of $\mathbf{A}$ in (1) and the estimator derived in [5] for the transformation matrix. The key issue is the use of model variances as regression error variances in [1, 5], while we use scaled identity matrices.

In summary, the iterative procedure used for retraining is as follows.

- For each training speaker l,
  1. apply transformation, $\mathbf{A}^{(i)}[l]$, to current models, $\mathcal{M}^{(i)}$, to produce models $\mathcal{M}^{(i)}[l]$ for speaker l
  2. run Baum-Welch adaptation using models $\mathcal{M}^{(i)}[l]$ on data from speaker l to produce means, $\mathbf{y}_j^{(i)}[l]$, variances, $\sigma^{2\,(i)}_{jk}$, and counts, $N_j^{(i)}[l]$, for all mixture components
  3. recompute transformation, $\mathbf{A}^{(i+1)}[l]$, as in (1), using $\mathcal{M}^{(i)}$ and means $\mathbf{y}_j^{(i)}[l]$
- Use per-speaker accumulations to recompute (i) the model means using equations (4) or (5), (ii) the variances and mixture weights using standard Baum-Welch updates, and produce new models, $\mathcal{M}^{(i+1)}$

The procedure is initialized ($i = 0$) with baseline SI models and identity matrix for each speaker's transformation. Since each Baum-Welch iteration (after the first) is run with speaker-adapted models, the variances obtained from this procedure are average within-speaker quantities.

We have performed some preliminary experiments with the models used in System II using the approximate solution, (5), and the results are shown in Table 4. Two iterations of the retraining procedure were performed, and results with one and three iterations during adaptation are included. Thirty transformations, derived from linguistic considerations, were used during training and adaptation.

mins. of adapt. data	no retraining		two iters. of (5)	
	T–1	T–3	T–1	T–3
19.5	12.4%	12.1%	11.7%	11.3%
26.0	12.4%	11.8%	11.6%	10.9%
39.0	12.2%	12.0%	11.3%	10.8%
52.0	12.0%	11.6%	11.4%	10.8%

Table 4. Word error rates for transformation approaches in System II with and without retraining

Retraining the models appears to provide nearly 7–10% extra reduction in the word error rate. In addition, recognition with the retrained models is about 10% faster; we attribute this to reduced model variances because our estimation procedure produces average within-speaker quantities.

7. CONCLUSION

The paper reports on several issues involving the use of transformation-based speaker adaptation for LVCSR systems. Experiments indicate that speaker normalization by frequency warping continues to provide additional improvement even after extensive speaker adaptation. We have shown that transformation-based adaptation compares favorably with standard MAP adaptation in a number of adaptation scenarios. The use of speaker transformation in the training process has been examined. We have also used some related ideas to improve our conversational speech system [7].

REFERENCES

[1] T. Anastasakos et al, "A compact model for speaker-adaptive training," *Proc. 1996 ICSLP,* pp. 1137–1140.

[2] S. Cox, "A speaker adaptation technique using linear regression," *Proc. 1995 ICASSP,* pp. 700–703.

[3] V. Digalakis and L. Neumeyer, "Speaker adaptation using combined transformation and Bayesian methods," *Proc. 1995 ICASSP,* pp. 680–683.

[4] J. L. Gauvain and C. H. Lee, "Maximum a posteriori estimation for multivariate Gaussian mixture observations of Markov chains," *IEEE Trans. Speech Audio Process.,* pp. 291–298, 1994.

[5] C. Leggetter and P. Woodland, "Maximum Likelihood linear regression for speaker adaptation of HMMs," *Computer Speech and Language,* pp. 171–186, 1995.

[6] D. B. Paul, "Extensions to phone-state decision-tree clustering," to appear in *1997 ICASSP*.

[7] B. Peskin et al., "Progress in Recognizing Conversational Telephone Speech," to appear in *1997 ICASSP*.

[8] S. Wegmann et al., "Speaker Normalization on Conversational Speech," *Proc. 1996 ICASSP,* pp. *I*.339–*I*.341.