SINGLE-PASS ADAPTED TRAINING WITH ALL-PASS TRANSFORMS

John McDonough and William Byrne

Center for Language and Speech Processing
The Johns Hopkins University
{jmcd,byrne}@jhu.edu

ABSTRACT

In recent work, the *all-pass transform* (APT) was proposed as the basis of a speaker adaptation scheme intended for use with a large vocabulary speech recognition system. It was shown that APT-based adaptation reduces to a linear transformation of cepstral means, much like the better known maximum likelihood linear regression (MLLR), but is specified by far fewer free parameters. Due to its linearity, APT-based adaptation can be used in conjunction with speaker-adapted training (SAT), an algorithm for performing maximum likelihood estimation of the parameters of an HMM when speaker adaptation is to be employed during both training and test. In this work, we propose a refinement of SAT called single-pass adapted training (SPAT) which achieves the same improvement in system performance as SAT but requires much less computation for HMM training. In a set of speech recognition experiments conducted on the Switchboard Corpus, we report a word error rate reduction of 5.3% absolute using a single, global APT.

1. INTRODUCTION

In *speaker adaptation*, we attempt to transform the cepstral means of a hidden Markov model (HMM) so as to better match the characteristics of some speech from a particular speaker. Speaker adaptation is typically undertaken to reduce the error rate of a large vocabulary conversational speech recognition (LVCSR) system. Certainly one of the most effective speaker adaptation methods is maximum likelihood linear regression (MLLR), wherein a transformation matrix is estimated using some speaker-dependent enrollment data, and then applied to the cepstral means of an HMM via a simple matrix-vector multiplication [4].

Speaker normalization is closely related to speaker adaptation, inasmuch as it attempts to transform the short-time *features* of a given speaker's speech so as to better match a speaker independent (SI) model. In prior work [6] we explored the use of the bilinear transform (BLT), and a generalization thereof dubbed the all-pass transform (APT), as a means of formulating practical speaker normalization schemes. We noted that the BLT and APT can be represented as linear transformations in the cepstral domain; this linearity conduces to robust estimation of the requisite speaker dependent transformation parameters. In other work [7] we proposed the use of the BLT and APT as the basis for a speaker adaptation scheme. This formulation is attractive because the APT devolves to a linear transformation of cepstral means, as does MLLR, but is specified by far fewer free parameters, and thus can be reliably estimated even with very limited enrollment data.

Speaker-adapted training (SAT) is an algorithm for performing maximum likelihood estimation of the parameters of an HMM when speaker adaptation is to be employed during both training and test [5]. SAT can be used with any speaker adaptation scheme employing a linear transformation of cepstral means, including both MLLR and the APT-based formulation. While effective at reducing word error rate, the computational expense of SAT can become prohibitive, especially when used with an incremental training procedure such as that favored by HTK, the Hidden Markov Model Toolkit [11]. This work is concerned with a novel modification of the SAT algorithm, which we refer to as Single-Pass Adapted Training (SPAT), and its combination with APT-based speaker adaptation. SPAT achieves the same reduction in word error rate achievable with SAT, but requires only marginally more computation than conventional speaker-independent HMM training.

This material is based upon work supported by the National Science Foundation under Grant No. #IIS-9732388, and carried out at the 1998 Workshop on Language Engineering, Center for Language and Speech Processing, The Johns Hopkins University. Any opinions, findings, conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation or the Johns Hopkins University.

2. THEORY AND ALGORITHMS

Here we summarize the theoretical development on which the APT and SAT are based. We then briefly discuss the implementation of APT parameter estimation and SAT within the Homewood Extensions, a set of C++ classes built atop HTK.

All-Pass Transforms

The bilinear transform is the the simplest member of the class of all-pass transforms; it can be expressed as

$$Q(z) = \frac{z-\alpha}{1-\alpha z} \tag{1}$$

where α and $|\alpha| < 1$. A more general APT can be defined [8] as

$$\begin{aligned} Q(z) &= \underbrace{\frac{z-\alpha}{1-\alpha z}}_{A(z;\alpha)} \underbrace{\frac{z-\beta}{1-\beta^* z}\frac{z-\beta^*}{1-\beta z}}_{B(z;\beta)} \underbrace{\frac{1-\gamma^* z}{z-\gamma}\frac{1-\gamma z}{z-\gamma^*}}_{G(z;\gamma)} \\ &= A(z;\alpha)\,B(z;\beta)\,G(z;\gamma) \end{aligned} \tag{2}$$

where β and γ are complex, and $|\alpha|, |\beta|, |\gamma| < 1$. Although we shall restrict our attention to (1–2) for the present, it should be

noted that an even more general APT can be expressed as

$$Q(z) = A(z;\alpha) \prod_{i=1}^{N_p} B(z;\beta_i)\, G(z;\gamma_i) \tag{3}$$

In what follows we shall be manipulating the *Laurent series* expansion

$$Q(z) = \sum_{n=-\infty}^{\infty} q[n]\, z^n$$

of an APT, where q is the relevant sequence of series coefficients. As an aid to this development, denote the *Cauchy product* of two sequences a and b as $c = a * b$, where the components of c are given by

$$c[n] = \sum_{k=-\infty}^{\infty} a[k]\, b[n-k] \tag{4}$$

In [8], expressions are derived for the Laurent series expansions of the factors $A(z;\alpha)$, $B(z;\beta)$ and $G(z;\gamma)$ appearing in (2). Denoting the coefficient sequences associated with these expansions as a, b, and g respectively, it can be shown $q = a * b * g$. Moreover, letting $q^{(m)}$ denote the coefficient sequence of $Q^m(z)$ for all integer $m \geq 0$ and $q^{(0)}$ the *unit sample sequence*:

$$q^{(0)}[n] = \begin{cases} 1; & \text{for } n = 0 \\ 0; & \text{otherwise} \end{cases}$$

it follows $q^{(m)} = q^{(m-1)} * q$ for all $m \geq 2$.

Consider the k^{th} cepstral mean μ_k of a hidden Markov model, and define the k^{th} *transformed mean* as $\hat{\mu}_k = A\,\mu_k$. Here, the components $\{a_{n,m}\}$ of the transformation matrix $A = A(\alpha,\beta,\gamma)$ are given by

$$a_{nm} = \begin{cases} q^{(m)}[0], & \text{for } n = 0, m \geq 0 \\ 0, & \text{for } n > 0, m = 0 \\ \left(q^{(m)}[n] + q^{(m)}[-n]\right), & \text{for } n, m > 0 \end{cases} \tag{5}$$

Prior to speech recognition, the parameters specifying $Q(z)$ must be estimated for each speaker in a test or training set. This is most easily accomplished through recourse to the EM algorithm [2], whose application entails the estimation of an *auxiliary function* and its subsequent maximization with respect to the relevant transform parameters. Consider a hidden Markov model composed of thousands of individual states; with each state is associated a probability density function composed of several Gaussian components. Let $c_{ki}^{(s)}$ denote the *posterior probability* that the cepstral feature $x_i^{(s)}$ was drawn from the k^{th} Gaussian component, and let $c_k^{(s)} = \sum_i c_{ki}^{(s)}$ denote the total occupancy count for this component over all frames in a set $\{x_i^{(s)}\}$ of enrollment data; the several $c_{ki}^{(s)}$ can be calculated via the well-known forward-backward algorithm [1, § 12]. Assuming all Gaussian components have diagonal covariance matrices of the form

$$D_k = \text{diag}\left(\sigma_{k0}^2\,,\ \sigma_{k1}^2\,,\ \sigma_{k2}^2\,,\ \ldots\,,\ \sigma_{k,L-1}^2\right) \tag{6}$$

the requisite auxiliary function for the simple BLT can be expressed as [8]

$$\mathcal{G}(\alpha) = \sum_k c_k^{(s)} \sum_n \frac{1}{\sigma_{kn}^2} \left(\tilde{\mu}_{kn} - \tfrac{1}{2}\hat{\mu}_{kn}\right) \hat{\mu}_{kn} \tag{7}$$

where the k^{th} speaker-dependent (SD) mean is given by

$$\tilde{\mu}_k^{(s)} = \frac{1}{c_k^{(s)}} \sum_i c_{ki}^{(s)} x_i^{(s)} \tag{8}$$

As there is no closed form solution for that α maximizing (7), a numerical optimization algorithm must be brought to bear. For the BLT, parameter optimization devolves to a simple linear search; good results have been obtained with *Brent's method* [10, Section 10.2]. Estimation of optimal parameters for general all-pass transforms can be accomplished with *Newton's method* [3, §4.4]; expressions for the gradient and Hessian required by Newton's method are developed in [8].

Speaker-Adapted Training

Speaker-Adapted Training (SAT) is an algorithm for performing maximum likelihood (ML) estimation of HMM parameters when speaker adaptation is to be used during both test and training [5]. At this point, the SAT re-estimation formulae are fairly well known; we briefly summarize them here to add coherence to the discussion to follow.

Let $\Lambda_k = (\mu_k, D_k)$ denote the parameters of the k^{th} Gaussian component, where μ_k is the SI mean and D_k the SI diagonal covariance. Let $A^{(s)}$ denote the global transformation matrix associated with speaker s, where, by assumption, all components of $A^{(s)}$ are determined by a much smaller number of APT parameters. Also let $\tilde{\mu}_k^{(s)}$ denote the SD mean for speaker s as in (8). The optimal SI mean is then given by

$$\mu_k = \left(\boldsymbol{A}^T \boldsymbol{D}_k \boldsymbol{A}\right)^{-1} \left(\boldsymbol{A}^T \boldsymbol{D}_k \tilde{\boldsymbol{\mu}}_k\right) \tag{9}$$

where

$$\boldsymbol{A} = \begin{bmatrix} A_k^{(1)} \\ A_k^{(2)} \\ \vdots \\ A_k^{(S)} \end{bmatrix}, \quad \boldsymbol{D}_k = \begin{bmatrix} c_k^{(1)} D_k^{-1} \\ c_k^{(2)} D_k^{-1} \\ \vdots \\ c_k^{(S)} D_k^{-1} \end{bmatrix}, \quad \text{and} \quad \tilde{\boldsymbol{\mu}}_k = \begin{bmatrix} \tilde{\mu}_k^{(1)} \\ \tilde{\mu}_k^{(2)} \\ \vdots \\ \tilde{\mu}_k^{(S)} \end{bmatrix}$$

The necessary matrix products can be written more compactly as

$$\boldsymbol{A}^T \boldsymbol{D}_k \boldsymbol{A} = \sum_s c_k^{(s)} A^{(s)T} D_k^{-1} A^{(s)} \tag{10}$$

$$\boldsymbol{A}^T \boldsymbol{D}_k \tilde{\boldsymbol{\mu}}_k = \sum_s c_k^{(s)} A^{(s)T} D_k^{-1} \tilde{\mu}_k^{(s)} \tag{11}$$

By way of re-estimating the diagonal covariance, let us decompose D_k as in (6); the optimal value of σ_{kl}^2 can then be expressed as

$$\sigma_{kl}^2 = \frac{1}{c_k} \sum_s c_k^{(s)} \left[\tilde{\sigma}_{kl}^{(s)\,2} + \left(\tilde{\mu}_{kl}^{(s)} - \hat{\mu}_{kl}^{(s)}\right)^2\right]$$

where $c_k = \sum_s c_k^{(s)}$ and the variance $\tilde{\sigma}_{kl}^{(s)\,2}$ for speaker s is given by

$$\tilde{\sigma}_{kl}^{(s)\,2} = \frac{1}{c_k^{(s)}} \sum_i c_{ki}^{(s)} \left(x_{il}^{(s)} - \tilde{\mu}_{kl}^{(s)}\right)^2 \tag{12}$$

Single-Pass Adapted Training

SAT can be applied to any speaker adaptation scheme based on a linear transformation of the original cepstral means—a property of both APT adaptation as well as the better-known MLLR [4]. When used with the latter, the SAT model is typically initialized with the final, multiple-mixture HMM obtained from conventional training. This approach *cannot* be used in the case of APT adaptation, as the speaker-dependent transforms estimated in this fashion will be indistinguishable from the identity, and no improvement in system performance will be achieved. This is a consequence of the highly constrained nature of the APT, as compared to the full-matrix transformation typically used in MLLR. Reasonable APT parameters can be estimated by beginning with a conventionally-trained HMM containing a *single* mixture for each state, accumulating speaker-dependent forward-backward statistics, then optimizing the auxiliary function in (7). The determinative factor is not that the HMM is composed of many Gaussian mixture components, but rather that each state is apportioned a single mixture, as this implies that each frame in the training set can, in some sense, only be aligned to a single Gaussian density.

After training the single-mixture SAT model, it is possible to simply split all Gaussian densities and continue with more forward-backward passes, split all densities, etc. This procedure is in keeping with the incremental approach to HMM training advocated by HTK and may yield results as good as the best, but is very time and resource consuming. A more efficient solution is provided by the novel Single-Pass Adapted Training (SPAT) strategy outlined here:

0. Use the HTK incremental training procedure to obtain a conventional, multiple-mixture, state-clustered SI model.
1. Perform several iterations of regular SAT beginning with the single-mixture, state-clustered triphone system generated as an intermediate result of Step 0. Keep the SD adaptation parameters for all training set speakers.
2. Beginning with the final, multiple-mixture SI model from Step 0, do a forward-backward pass on all utterances in the training set and dump SD statistics. Note that no speaker adaptation is performed on the SI model prior to forward-backward alignment.
3. Using the SD adaptation parameters from Step 1 and the SD forward-backward statistics from Step 2, perform a regular SAT combination step.
4. Perform several additional iterations of normal SAT beginning with the model obtained from Step 3 and SD adaptation parameters from Step 1.

SPAT can also be combined with single-pass training to change the parameterization of the original SI model. This is accomplished in Step 2 by using the feature set on which the original model was trained to calculate posterior probabilities, but accumulating sufficient statistics with the new set of features; the latter are then used in the SAT combination stage, Step 3. If the APT is augmented with an additive bias, the best system performance is obtained when no cepstral mean subtraction (CMS) is applied to the features of the training and test sets. As the original SI model is unadapted, however, CMS is generally used during its training and test. The combined SPAT/single-pass re-training procedure is very useful in this case for changing from features normalized with CMS to features with no such normalization.

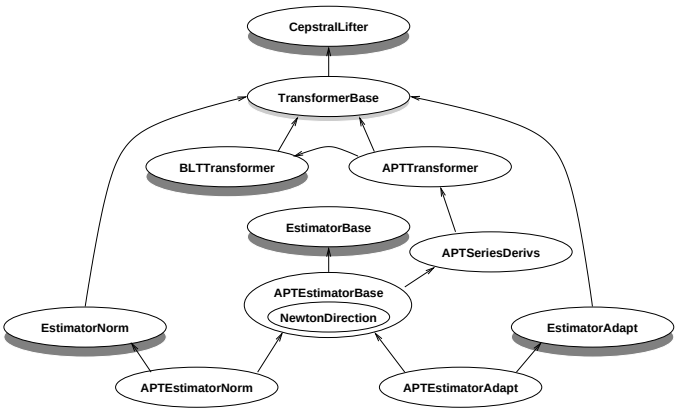

Figure 1: Class hierarchies for BLT and APT parameter estimation defined in the Homewood Extensions. Drop shadows indicate shared classes.

The Homewood Extensions

The Homewood Extensions (THE) are a set of C++ classes built atop HTK. In its present incarnation, THE performs speaker normalization and adaptation based on the APT—in particular, THE provides implementations for the APT parameter estimation and SAT algorithms discussed above. In the development of THE, great pains were taken to provide clear and concise implementations of all algorithms. Some appreciation of this fact can be obtained by examining Figure 1, which illustrates the class hierarchy for maximum likelihood (ML) estimation of the multiple-parameter APT defined in (2–3). ML estimation entails the use of a numerical optimization algorithm; the form of the relevant objective function depends on whether speaker normalization or adaptation is to be performed [9, §2]. To achieve the necessary specialization while ensuring no source code is duplicated, the class hierarchy includes distinct branches for normalization and adaptation, as shown in the figure.

Although developed for HTK, the Homewood Extensions were written so as to be readily portable to any speech recognition system based on a continuous density HMM. This was accomplished by *hiding* all explicit references to HTK behind class interfaces, and thereafter accessing the necessary components of HTK solely through these interfaces. THE is publicly available for all non-commerical use and can be found at `www.clsp.jhu.edu/~jmcd`.

3. SPEECH RECOGNITION EXPERIMENTS

The speech recognition experiments discussed below were conducted using training and test material extracted from the *Switchboard Corpus*. Of the complete Switchboard Corpus, approximately 140 hours of data are set aside for system training. In order to obtain fast turnaround, however, a subset of the full training set was identified and used in all speaker adaptation experiments. This subset, dubbed *MiniTrain*, is composed of approximately 200 conversations providing a total of 18.6 hours of speech material. Approximately 100 speakers of each gender participate in the MiniTrain conversations. The test set used in all experiments was comprised of 19 Switchboard conversations, for a total of 18,000 words.

System Description	% Word Error Rate	
	w/o Bias	with Bias
Baseline	48.9	48.9
BLT (1-param.)	45.0	44.4
APT (5-param.)	44.4	43.9
APT (9-param.)	44.3	43.6

Table 1: Results of lattice rescoring experiments comparing BLT- and APT-based speaker adaptation schemes both with and without an additive bias.

The features used for speech recognition were composed of mel-frequency cepstral coefficients 1–12 along with first and second order difference coefficients derived from these. Parameters corresponding to short-time energy and its first and second order difference were also estimated, for a total feature length of 42. The mel-frequency cepstral coefficients were calculated using the waveform analysis tools provided with HTK. Cepstral mean subtraction was applied to the features of the test and training sets on a per utterance basis; no other feature normalization was applied.

All speech recognition experiments were conducted using a hidden Markov model trained with cross-word triphones. Each triphone in the model was composed of three states, and each state was composed of nine Gaussian components. The standard HTK implementation of the decision tree algorithm was used to generated the state clusters of the HMM. The final model was composed of approximately 3,000 distinct states.

Table 1 provides the results of an initial set of speech recognition experiments conducted to illustrate the effectiveness of the BLT- and APT-based speaker adaptation schemes, both with and without the inclusion of an additive bias. These results were obtained by rescoring a set of lattices using an appropriately adapted SI model, which was trained using the SPAT procedure. For each condition reported, the adaptation scheme employed for training was matched to that for test. In all cases, an entire conversation side—approximately 2.5 minutes of unsupervised enrollment data—was used in estimating the speaker-dependent adaptation parameters, a paradigm typically referred to as *transcription mode*. For the purpose of parameter estimation, the errorful transcripts obtained with the unadapted, baseline model were used for the requisite forward-backward passes. The initial lattices were also generated using the baseline system.

To complement the results reported above, we conducted two experiments with MLLR adaptation. Applying MLLR to the baseline SI model reduced WER from 48.9% to 45.6%. In a second experiment, we cascaded MLLR with APT adaptation: We first conducted a lattice rescoring with the best no-bias APT model, which gave a WER of 44.3%. We then used the putative transcriptions from this rescoring to perform unsupervised estimation of MLLR parameters. By using these parameters to transform the APT-adapted model, we were able to reduce WER from 44.3% to 42.3%. This result illustrates that the effects of APT and MLLR adaptation are largely, albeit not perfectly, additive.

4. CONCLUSIONS

Speaker-adapted training (SAT) is an algorithm for obtaining maximum likelihood estimates of the parameters of an HMM when speaker adaptation is used in both test and training. While effective at improving the performance of a large vocabulary speech recognition system, the computation required by SAT can become prohibitive, especially when used with the incremental build procedure preferred by HTK, the Hidden Markov Model Toolkit. In this work we have proposed a novel modification of SAT known as single-pass adapted training (SPAT), which is as effective at improving system performance but requires much less computation. Exploiting the linearity of the recently-introduced APT, we have formulated a speaker adaptation scheme by using APT-based adaptation with the SPAT procedure; this combination has achieved word error rate reductions of 5.3% absolute in a series of experiments conducted on conversational speech material from the Switchboard Corpus. The Homewood Extensions (THE) are a set of C++ classes implementing the speaker adaptation and training algorithms discussed in this work; THE is publicly available for all non-commercial use at `www.clsp.jhu.edu/~jmcd`. Future work will study the use of multiple regression classes for speaker adaptation, as well as the optimal assignment of the Gaussian components to these classes.

5. REFERENCES

[1] J. R. Deller, J. G. Proakis, and J. H. L. Hansen. *Discrete-Time Processing of Speech Signals*. Macmillan, New York, 1993.

[2] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihoood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society*, 39 B:1–38, 1977.

[3] P. E. Gill, W. Murray, and M. H. Wright. *Practical Optimization*. Academic Press, London, 1981.

[4] C. J. Leggetter and P. C. Woodland. Maximum likelihood linear regression for speaker adaptation of continuous density hidden markov models. *Computer Speech and Language*, pages 171–185, 1995.

[5] J. McDonough. On the estimation of optimal regression classes for speaker adaptation. *Computer Speech and Language*, submitted for publication.

[6] J. McDonough, W. Byrne, and X. Luo. Speaker normalization with all-pass transforms. In *Proc. ICSLP*, 1998.

[7] John McDonough and William Byrne. Speaker adaptation with all-pass transforms. In *Proc. ICASSP*, volume II, pages 1047–1050, 1999.

[8] John W. McDonough. Speaker normalization with all-pass transforms. Research Notes 28, Center for Language and Speech Processing, The Johns Hopkins University, 1998.

[9] John W. McDonough. The Homewood Extensions. Research Notes 39, Center for Language and Speech Processing, The Johns Hopkins University, 1999.

[10] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical Recipes in C*. Cambridge University Press, Cambridge, second edition, 1992.

[11] S. Young, J. Jansen, J. Odell, D. Ollason, and P. Woodland. *The HTK Book*. Entropic Software, Cambridge, 1997.