GRADIENT STEEPNESS METRICS USING EXTENDED BAUM-WELCH TRANSFORMATIONS FOR UNIVERSAL PATTERN RECOGNITION TASKS

Tara N. Sainath

MIT Computer Science and
Artificial Intelligence Laboratory
32 Vassar St. Cambridge, MA 02139
tsainath@mit.edu

Dimitri Kanevsky and Bhuvana Ramabhadran

IBM T. J. Watson Research Center
Yorktown, NY 10598, U.S.A.
{kanevsky, bhuvana}@us.ibm.com

ABSTRACT

In many pattern recognition tasks, given some input data and a family of models, the "best" model is defined as the one which maximizes the likelihood of the data given the model. Extended Baum-Welch (EBW) transformations are most commonly used as a discriminative technique for estimating parameters of Gaussian mixtures. In this paper, we use the EBW transformations to derive a novel gradient steepness measurement to find which model best explains the data. We use this gradient measurement to derive a variety of EBW metrics to explain model fit to the data. We apply these EBW metrics to audio segmentation via Hidden Markov Models (HMMs) and show that our gradient steepness measurement is robust across different EBW metrics and model complexities.

Index Terms— Pattern recognition, gradient methods.

1. INTRODUCTION

Pattern recognition [1] is important in a variety of applications, including speech recognition, audio classification, speaker verification and audio information retrieval. In a general pattern recognition task, given some input data and a family of models, the goal is to evaluate which model best explains the data. Typically, an objective function, for example a likelihood probability, is computed to measure how well the model characterizes the data. Recently, a new approach for evaluating model fitness to data has been explored which is based on the principle of how much effort is required to change one model into another given some evaluation data. For example, the Earth Mover's Distance (EMD) ([2]) evaluates model fitness to data by calculating the minimal cost needed to transform one distribution to another. In addition, feature space Gaussianization [3] computes a distance between models in an original and transformed feature space.

In this paper, we look to evaluate model fitness by using a gradient steepness measurement. Given some data, a set of models and an objective function, we can update (train) each of the models by finding the best step along the gradient of the objective function. During such an update, each of the models changes such that models that fit the data best change the least and have the flattest gradient. Therefore the best fitting model has the flattest gradient slope.

One of the popular training methods used to estimate updated models, which we explore in this work, is the Extended Baum-Welch (EBW) transformations. EBW transformations have been used extensively in the speech recognition community as a discriminative training technique to estimate model parameters of Gaussian mixtures. For example, in [4] the EBW transformations were used for Maximum Mutual Information (MMI) training of large vocabulary speech recognition systems.

We have explored using the EBW gradient steepness measurement in a few pattern recognition applications. In [5] the likelihood ratio test, typically used for audio segmentation tasks, was redefined with the EBW gradient steepness criteria, while in [6] we explored using EBW for audio classification. In addition, in [7] the EBW metric was used in Hidden Markov Models (HMMs) and showed improvements over the likelihood metric for phonetic recognition.

In this work, we present the gradient steepness metric from a general pattern processing perspective. First, we continue to expand on previous work [5], [6], [7]), now looking at a large vocabulary task, and use the gradient metrics to introduce a variety of novel EBW methods which can describe model fitness to data. We show that the EBW gradient measurement is robust across the different EBW metrics and model complexities and appears to be a general technique to explain the quality of a model used to represent the data. While the EBW metrics presented can be used for general pattern processing applications, our experiments focus on using these metrics for speech/non-speech segmentation of broadcast news via HMMs, a state of the art method for segmentation [8]. Since HMMs are so widely used in speech recognition, success of our gradient steepness measure in HMMs will introduce a new decoding metric.

The following section provides background on the EBW transformations and general gradient measurement, followed by the EBW metrics in Section 3. Section 4 presents the experiments performed, followed by a discussion of these results in Section 5. Finally, Section 6 concludes the paper and discusses future work.

2. EXTENDED BAUM-WELCH TRANSFORMATIONS

2.1. Derivation of EBW Transformations

The EBW procedure involves continuous transformations that can be described as follows. Assume that frame x_i is drawn from Gaussian mixture model (GMM) λ^k, with each component $j \in k$ parameterized by the following mean and variance parameters $\lambda_j^k = \{\mu_j^k, \sigma_j^k\}$, and weight w_j^k. Thus GMM λ^k includes all the parameters of the individual components, in other words $\lambda^k = \{\lambda_1^k, \ldots \lambda_N^k\}$ and weights $w^k = \{w_1^k, \ldots w_N^k\}$. Let us define the probability of frame x_i given mixture component j as $p(x_i|\lambda_j^k) = z_{ij}^k = \mathcal{N}(\mu_j^k, (\sigma_j^k)^2)$ and similarly $z_i^k = \sum_{j=1}^N w_j^k z_{ij}^k$. Let $F(z_{ij}^k)$ be some objective function over z_{ij}^k and $c_{ij}^k = z_{ij}^k \frac{\delta}{\delta z_{ij}^k} F(z_{ij}^k)$. Given this function and initial model parameters λ_j^k, the EBW transformations provide for-

mulas to re-estimate model parameters $\lambda_j^k(\epsilon) = \{\mu_j^k(\epsilon), \Sigma_j^k(\epsilon)\}$ as:

$$\hat{\mu}_j^k = \hat{\mu}_j^k(\epsilon) = \frac{\sum_{i=1}^{M} c_{ij}^k x_i \epsilon + \mu_j^k}{\sum_{i=1}^{M} c_{ij}^k \epsilon + 1} \quad (1)$$

$$(\hat{\Sigma}_j^k) = \hat{\Sigma}_j^k(\epsilon)^2 = \frac{\sum_{i=1}^{M} c_{ij}^k x_i x_i^T \epsilon + \left(\mu_j^k \mu_j^{kT} + (\Sigma_j^k)\right)}{\sum_{i=1}^{M} c_{ij}^k \epsilon + 1} - (\hat{\mu}_j^k)\mu_j^{kT} \quad (2)$$

Here ϵ is a small constant such that $F(z_{ij}^k)$ increases per iteration, that is $F(\hat{z}_{ij}^k) \geq F(z_{ij}^k)$. [9]

2.2. EBW Fitness Curve

As shown in Figure 1, given an initial model, $\lambda(0)$, for our data and an objective function F_Λ, we can estimate a new model, $\lambda(\epsilon_0)$, for our data using the EBW tranformations by finding the best step along the gradient of the objective function. We can think of the gradient slope as measuring how much we have to adapt an initial model to fit the data. In what follows we introduce a general gradient steepness concept between data and model that generalizes the definition of gradient steepness and its relation to the EBW transformations.

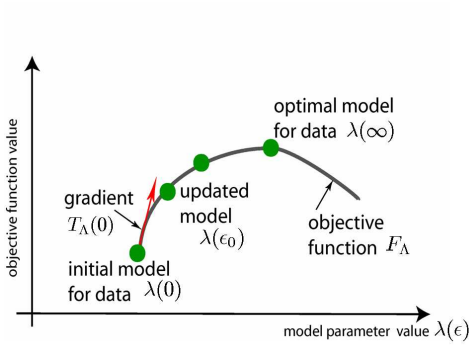

Fig. 1. EBW Model Update Graph

Definition 1 *FITNESS CURVE*

$$\textit{Let } \Lambda = \{\lambda(\epsilon)\} = \{(\hat{\mu}(\epsilon)), \hat{\Sigma}(\epsilon)) \subset R^{n^2+n}, 0 \leq \epsilon \leq \infty\} \quad (3)$$

denote a parametric curve in $(n^2 + n)$-dimensional vector space R^{n^2+n}, where ϵ changes between 0 and ∞ and points $\lambda(\epsilon)$ on this curve Λ are transformations of means and variance as defined in (1) and (2). The parameter ϵ controls the rate at which we estimate our updated model. If ϵ is very small then training is very slow (but stable). However, if ϵ is too large model re-estimation may not increase the objective function on each iteration. [9]

$$\textit{Let us call: } F_\Lambda : [0, \infty] \rightarrow R, \epsilon \rightarrow F(\lambda(\epsilon)) \quad (4)$$

an EBW fitness curve for a model λ, data Y and function F. Also:

$$T_\Lambda(0) = lim_{\epsilon_0 \rightarrow 0} \frac{F_\Lambda(\hat{\lambda}(\epsilon_0)) - F_\Lambda(\lambda(0))}{\epsilon_0} \quad (5)$$

denote a tangent to the curve F_Λ at a point $\{0, F_\Lambda(\lambda(0))\}$, as indicated in Figure 1. Here $\lambda(0)$ represents the initial model and $\hat{\lambda}(\epsilon_0)$ the updated model estimated from the EBW transformations defined in (1) and (2).

Intuitively, the flatter the fitness curve F_Λ, the better the initial model $\lambda(0)$ fits the data Y. The flatness of the fitness curve F_Λ is represented by the tangent to the curve at point $\lambda(0)$. In other words, these tangents T to the fitness curve F_Λ at $\lambda(0)$ characterize the fitness of model $\lambda(0)$ to data Y. The smaller these tangents, the better the fitness. In [9], it was shown that T could be represented as sum of some squared terms and therefore is always non-negative.

Thus, the EBW transformations provide solutions to estimate an updated model, and also provide a measure of gradient steepness. Having a graphical idea of the EBW gradient steepness measurement, we can now derive our gradient measurement more formally. Using EBW transformations (1) and (2) such that $\lambda_j^k \rightarrow \hat{\lambda}_j^k(\epsilon)$ and $z_{ij}^k \rightarrow \hat{z}_{ij}^k$, [9] derives a linearization formula between $F(\hat{z}_{ij}^k)$ and $F(z_{ij}^k)$ for small ϵ as:

$$F(\hat{z}_{ij}^k) - F(z_{ij}^k) = T_\Lambda(0)\epsilon + o(\epsilon) \quad (6)$$

Here T measures the gradient required to adapt the initial model λ_j^k to data x_i, or equivalently how well the data is explained by the initial model λ_j^k. The larger the value of T indicates that the gradient to adapt the initial model to the data is steeper and $F(\hat{z}_{ij}^k)$ is much larger than $F(z_{ij}^k)$. Thus the data is much better explained by the updated model $\hat{\lambda}_j^k(\epsilon)$ compared to the initial model λ_j^k. In the next section, we derive our EBW gradient steepness metrics using both sides of Equation 6.

3. EBW GRADIENT STEEPNESS METRICS

Given a family of models $\Theta = \{\theta_1, \theta_2, \ldots, \theta_K\}$, the goal of a generic pattern recognition problem is to which model best describes data $x_i \in R^d$. Below we present the standard Gaussian Mixture Model (GMM) likelihood method used in pattern recognition tasks. Then we present our novel EBW metrics derived from our gradient steepness measurement discussed in Section 2.2.

3.1. GMM Likelihood

Assume that frame x_i is drawn from a GMM θ_k where z_{ij}^k is the likelihood of frame x_i given component $j \in k$ and w_j^k the *a priori* weight of component j. We define the log-likelihood of x_i given model θ_k by $F(z_i^k)$ as follows:

$$F(z_i^k) = p(x_i|\theta_k) = \log \sum_{j=1}^{N} w_j^k z_{ij}^k \quad (7)$$

Given an input sample x_i, we compute how well the data is represented by each model θ_k and choose the model θ^* which has the maximum likelihood. In other words: $\theta^* = \arg\max_{\theta_k} F(z_i^k)$. Sections 3.2-3.6 discuss different EBW metrics derived from our gradient steepness measurement.

3.2. EBW-T

Instead of calculating the likelihood of data x_i belonging to model θ_k, we can measure this via the T value in Equation 6, as initially demonstrated in [5]. In [9], Kanevsky derives a closed form

solution for T given any rational objective function $F(z_i^k)$ and $c_{ij}^k = z_{ij}^k \frac{\delta}{\delta z_{ij}^k} F(z_{ij}^k)$. In this work, we consider $F(z_i^k)$ as given by Equation 7 and thus:

$$c_{ij}^k = z_{ij}^k \frac{\delta}{\delta z_{ij}^k} F(z_{ij}^k) = \frac{z_{ij}^k w_j^k}{\sum_{l=1}^N w_l^k z_{il}^k}. \quad (8)$$

From [9], it then follows that for small ϵ, T_i^k is given as follows:

$$T_i^k = \sum_{j=1}^N \left\{ \sum_{r=1}^d \frac{\{\sum_{i=1}^M c_{ij}^k [(x_{ir} - \mu_{rj}^k)^2 - (\sigma_{rj}^k)^2]\}^2}{2(\sigma_{rj}^k)^4} \right\} + \sum_{j=1}^N \left\{ \sum_{r=1}^d [\frac{\sum_{i=1}^M c_{ij}^k (x_{ir} - \mu_{rj}^k)}{\sigma_{rj}^k}]^2 \right\} \quad (9)$$

Note that T has a closed form solution and does not require model re-estimation, making it computationally cheap [5]. The best model θ^* is the one where the gradient to adapt this model is smallest, and thus has the smallest T. Thus our decision rule for the best model can be written as: $\theta^* = arg\min_{\theta_k} T_i^k$. Note that Equation 6 holds only for small ϵ. In the next section, we introduce another EBW metric using the left side of Equation 6.

3.3. EBW-F

In this metric, given an input sample x_i, the best model θ^* is the one which has the smallest increase in likelihood given the updated model $F(\hat{z}_i^k)$ relative to the likelihood given the initial model $F(z_i^k)$. In other words the decision rule for the best model is:

$$\theta^* = arg\min_{\theta^k} \left(F(\hat{z}_i^k) - F(z_i^k) \right) / \epsilon \quad (10)$$

We look at using a global value of ϵ, as well as an adaptive value of ϵ, similar to [7]. The higher the likelihood of frame x_i given model θ_k, i.e. $p(x_i|\theta_k)$, the better the initial model θ_k. Therefore, we also explore setting $1/\epsilon = p(x_i|\theta_k) = \sum_{j=1}^N w_j^k z_{ij}^k$, which offers the property that the higher the likelihood the smaller ϵ and the slower the updated model is estimated. In [6] and [7], we have only explored setting $F(z_i^k) = p(x_i|\theta_k)$. Below we derive EBW-F gradient metrics for other objective functions.

3.4. Normalized EBW-T

In [7] we showed that normalizing the EBW-F scores at each frame allowed for improved performance in HMMs, as scores for a state sequence are computed by summing up scores assigned to individual frames. Here, we derive a similar normalization method for EBW-T. We can define the normalized EBW distance associated with model θ_k and frame x_i by normalizing T_i^k, the change in likelihood between an initial and updated model, with the likelihood given the initial model, $p(x_i|\theta_k)$, as:

$$T_i^k / \{p(x_i|\theta_k)\}^\alpha \quad (11)$$

where α is some positive number that controls the weight given to the likelihood $p(x_i|\theta_k)$ relative to T_i^k.

This local normalization at each frame x_i can be related to a "global" normalization for a sequence of observation frames $X = \{x_1, \dots x_i, \dots x_M\}$. For example, we can think of X as representing observations from the best HMM path. Assume the likelihood score function as $p(X|\theta_k) = \prod_1^m p(x_i|\theta_k)$. Then, the distance $T(p(X|\theta_k))$ is defined as $lim_{\epsilon \to 0} \frac{p(X|\theta_k(\epsilon) - p(X|\theta_k)}{\epsilon}$. This implies that the distance T has a usual multiplicativity property for derivatives of products, that is $T(p(X|\theta_k)) = \sum_j \prod_1^{j-1} p(x_j|\theta_k) * T(p(x_j|\theta_k) * \prod_{j+1}^m p(x_j|\theta_k)$. Therefore, by normalizing $T(p(X|\theta_k))/p(X|\theta_k)$ we can represent this as a sum of local normalizations at each frame: $\sum T_i^k / p(x_i|\theta_k)$, which is computed via Equation 11.

3.5. EBW-MMIE

Instead of using the objective function for F given by Equation 7, we can consider for each frame x_i and model θ_k the MMIE criteria:

$$F(z_{ij}^k) = \log I(x_i|\theta_k) = \log \frac{p(x_i|\theta_k)}{\sum_{m=1}^L p(x_i|\theta_m)} \quad (12)$$

where $p(x_i|\theta_k) = \sum_{j=1}^N w_j^k z_{ij}^k$ and L is the total number of models. Using this objective function, c_{ij}^k coefficients for MMIE are:

$$c_{ij}^k = z_{ij}^k \frac{\delta}{\delta z_{ij}^k} F(z_{ij}^k) = \frac{z_{ij}^k w_j^k}{\sum_{l=1}^L w_l^k z_{il}^k} - \frac{z_{ij}^k w_j^k}{\sum_{m,l=1} w_l^m z_{il}^m} \quad (13)$$

Equation 8 gives the formula for the ML based representation for c_{ij}^k. In this formula, we can see that the higher the likelihood given component j, that is $z_{ij}^k w_j^k$, the larger c_{ij}^k and the more weight is added to T. However the best model is one which has the smallest T. The MMIE based representation for c_{ij}^k will have a smoothing effect when some Gaussian component $w_l^m z_{tl}^m$ grows significantly. This can be seen easily from the following example: $\frac{1}{c} - \frac{1}{(c+d)} = \frac{d}{c(c+d)} \approx \frac{d}{c^2}$. In other words if c grows by some factor h then $1/c - 1/(c+d)$ decreases by the square factor $(c+h)^2$. Therefore, with the MMIE criterion, we do not increase c_{ij}^k by as much for a higher likelihood, which adds less weight to T.

3.6. EBW Forward Algorithm

In the previous methods, models are re-estimated and the EBW is scored on a per-frame basis. However, in this EBW metric we explore estimating models using history form previous frames. The EBW Forward algorithm is described as follows:

For each HMM state s_t, we associate the following HMM parameters: $\lambda^{s_t} = \{\mu_{s_t}^k, \Sigma_{s_t}^k, w_{s_t}^k\}$

Step 1 (t=1): For $t = 1$, find the "best" EBW first state $s_t = s_1$ using $T(x_i|s_t)$. We also associate with $s_t = s_1$ the parameters $C_{s_t} = c_{ij}^{s_t}$, $C_{s_t}^1 = c_{ij}^{s_t} * x_i$,$C_{s_t}^2 = c_{ij}^{s_t} * x_i^2$, and we set:

$$\tilde{\mu}_{ij}^{s_t} = \frac{c_{ij}^{s_t} * x_i * \epsilon + \mu_{ij}^{s_t}}{c_{ij}^{s_t} * \epsilon + 1} \quad (14)$$

$$\tilde{\sigma}_{ij}^{s_t} * \tilde{\sigma}_{ij}^{s_t} = \frac{c_{ij}^{s_t} * x_i^2 * \epsilon + ((\sigma_{ij}^{s_t})^2 + (\mu_{tj}^{s_t})^2)}{c_{ij}^{s_t} * \epsilon + 1} - (\tilde{\mu}_{ij}^{s_t})^2 \quad (15)$$

Step t : For each state s_t perform the following computations:

$$\tilde{\mu}_{ij}^{s_t} = \frac{(\alpha_t * C_{s_{t-1}}^1 + c_{s_t}) * x_i * \epsilon_t + \mu_{ij}^{s_{t-1}}}{(C_{s_{t-1}} + c_{s_t}) * \epsilon_t + 1} \quad (16)$$

$$\tilde{\sigma}_{ij}^{s_t} * \tilde{\sigma}_{ij}^{s_t} = \frac{(\alpha_t * C_{s_{t-1}}^2 + c_{ij}^{s_t} * x_i^2) * \epsilon + ((\sigma_{ij}^{s_t})^2 + (\mu_{ij}^{s_t})^2)}{(C_{s_{t-1}} + c_{ij}^{s_t}) * \epsilon + 1} - (\tilde{\mu}_{ij}^{s_t})^2 \quad (17)$$

$C_{s_t} = \alpha_t * C_{s_{t-1}} + c_{s_t}$, $C_{s_t}^1 = \alpha_t * C_{s_{t-1}} + c_{s_t} * x_i$, $C_{s_t}^2 = \alpha_t * C_{s_{t-1}}^2 + c_{s_t} * x_i^2$. Where $\alpha_t = 1$ if $s_t = s_{t-1}$, or $\alpha_t = 0$ if $s_t \neq s_{t-1}$. Increase t by 1 and continue.

4. EXPERIMENTS

We perform speech/non-speech segmentation on the English Broadcast News component of the IBM Global Autonomous Language Exploitation (GALE) system [10]. In our system, speech and non-speech segments are both modeled by a five-state, left-to-right HMM [10]. The output distributions in each HMM are tied across all states and are modeled with mixture of diagonal-covariance Gaussians. In the HMM segmentation, we first score individual frames using the distance metrics discussed in Section 3. A Viterbi search then finds the most likely sequence of states based on these scores.

We use 16 Gaussian mixtures for non-speech and run three experiments, comparing the performance using 50, 100 and 240 Gaussian mixtures for speech. Models are trained using approximately 140 hours of hand-transcribed data from Hub4. For testing, we focus on the RT-04 test set, which contains 12 shows of roughly 25 minutes (totaling about 300 minutes of Broadcast News).

After speech/non-speech segmentation is performed, we then decode the resulting speech segments using a speaker-independent system similar to [10]. Since the overall goal of our GALE system is to try to find an appropriate segmentation to minimize word error rate (WER), we evaluate the performance of the different EBW distance metrics via this criterion.

5. RESULTS

Table 1 shows the final decoding WER and number of errors using the different EBW metrics and speech mixture components. Please note that results which are statistically insignificant from the best performing method in each column are noted by $\approx$. First, we see that the performance of each of the EBW metrics is relatively the same for 240 mixture components, and also similar to the likelihood and oracle where the true speech/non-speech segments are known *a priori*. This demonstrates that our new gradient steepness measurement is robust across different EBW metrics. In addition, note that the 4 new EBW metrics introduced in this work, namely the EBW-F Adaptive ϵ, EBW-Norm, EBW-MMIE, and EBW-Forward algorithms, offer slightly improved performance over the EBW-T and EBW-F metrics, previously explored in [5] and [6], [7] respectively.

Metric	240 Mixtures	100 Mixtures	50 Mixtures
Oracle Seg	16.3 (7578)	16.3 (7578)	16.3 (7578)
Likelihood	16.4 (7653) $\approx$	16.6 (7725)	16.6 (7743)
EBW-T	16.5 (7682)	18.0 (8393)	17.1 (7979)
EBW-F	16.5 (7665)	16.5 (7693)	16.5 (7705) $\approx$
EBW-Adapt.	16.4 (7656) $\approx$	16.5 (7664)	16.5 (7704) $\approx$
EBW-Norm	16.4 (7625) $\approx$	**16.4 (7621)**	**16.5 (7675)**
EBW-MMIE	16.4 (7661) $\approx$	16.5 (7691)	16.6 (7715)
EBW-Fwd	**16.4 (7617)**	16.7 (7788)	16.7 (7778)

Table 1. Word Error Rates and (Number of Errors) for Seg. Metrics

In addition, each EBW metric offers its own individual benefits, which can be useful depending on the task at hand. For example, as shown by Equation 9, EBW-T does not require an updated model to be computed like the EBW-F methods, and therefore offers computation benefits. In addition, the EBW-F Adaptive method does not require us to tune ϵ for the specific task. In tasks where the best model at each frame is dependent on neighboring frames and does not change frequently, the EBW-Forward offer advantages, while the EBW-Norm offers advantages when summing scores across frames.

Finally, when we decrease the number of mixture components from 240 to 100 to 50, we see that the performance of the likelihood metric degrades much more relative to the EBW metrics. Please note that the performance of EBW-FB gets slightly worse because the model estimates at each point in time are computed based on previous model estimates, which will be poorer for smaller mixture components. As shown by Equation 10, the EBW metrics capture the difference between the likelihood of the data given the initial model and the likelihood with a model estimated from the current data frame being classified, while the likelihood just calculates the former. If the model estimate is poor, the likelihood is not able to take into account this model error [1] which can be present. Therefore model re-estimation via EBW using the current data is able to correct for this initial model error, and explains why the EBW metrics outperforms the likelihood for 50 and 100 components.

6. CONCLUSIONS

In this paper, we introduced a novel gradient steepness measurement that can be used for general pattern recognition tasks to explain how well the data fits the model. We derived a variety of EBW metrics from this gradient measurement an applied these metrics for HMM speech/non-speech segmentation. We found that our gradient measure was robust across different EBW metrics and model complexities. In the future, we would like to explore using the EBW gradient metrics in HMMs for other large scale vocabulary tasks.

7. REFERENCES

[1] R. Duda, P. Hart, and D. Stork, *Pattern Classification*, Wiley Interscience, 2nd edition, 2001.

[2] Y. Rabner, C. Tomasi, and L. J. Guibas, "The Earth Mover's Distance as a Metric for Image Retrieval," *International Journal of Computer Vision*, vol. 40, pp. 99–121, 2000.

[3] M. Padmanabhan and S. Dharanipragada, "Maximum-likelihood Nonlinear Transformation for Acoustic Adaptation," *IEEE Transactions on Speech and Audio Processing*, vol. 12, pp. 572 – 578, November 2004.

[4] V. Valtchev, P. C. Woodland, and S. J. Young, "Lattice-based Discriminative Training for Large Vocabulary Speech Recognition Systems," *Speech Communication*, vol. 22, 1996.

[5] T. N. Sainath, D. Kanevsky, and G. Iyengar, "Unsupervised Audio Segmentation using EBW Transformations," in *Proc. ICASSP*, April 2007.

[6] T. N. Sainath, V. Zue, and D. Kanevsky, "Audio Classification using EBW Transformations," in *Proc. Interspeech*, 2007.

[7] T. N. Sainath, D. Kanevsky, and B. Ramabhadran, "Unsupervised Audio Segmentation using EBW Transformations," in *To Appear in Proc. ASRU*, December 2007.

[8] B. Ramabhadran, J. Huang, U. Chaudhari, G. Iyengar, and H. J. Nock, "Impact of Audio Segmentation and Segment Clustering on Automated Transcription Accuracy of Large Spoken Archives," in *Proc. EuroSpeech*, 2003.

[9] D. Kanevsky, "Extended Baum Transformations For General Functions, II," Tech. Rep. RC23645(W0506-120), Human Language Technologies, IBM, 2005.

[10] D. Povey and B. Kingsbury, "Evaluation of Proposed Modifications to MPE for Large Scale Discriminative Training," in *Proc. ICASSP*, 2007.