Audio Classification using Extended Baum-Welch Transformations

Tara N. Sainath[1], *Victor Zue*[1] *and Dimitri Kanevsky*[2]

[1]MIT Computer Science and Artificial Intelligence Laboratory
32 Vassar St. Cambridge, MA 02139, U.S.A
[2]IBM T. J. Watson Research Center, Yorktown, NY 10598, U.S.A
{tsainath, zue}@mit.edu[1], kanevsky@us.ibm.com[2]

Abstract

Audio classification has applications in a variety of contexts, such as automatic sound analysis, supervised audio segmentation and in audio information search and retrieval. Extended Baum-Welch (EBW) transformations are most commonly used as a discriminative technique for estimating parameters of Gaussian mixtures, though recently they have been applied in unsupervised audio segmentation. In this paper, we extend the use of these transformations to derive an audio classification algorithm. We find that our method outperforms both the Support Vector Machine (SVM) and Gaussian Mixture Model (GMM) likelihood classification methods.

Index Terms: audio classification, gradient methods

1. Introduction

Audio steams, such as broadcast news or meeting recordings, contain audio from a wide variety of sources, including speech, music, coughing, laughter, etc. Classification has become an important tool to describe an audio scene characterized by numerous acoustic events. In addition, it has also been used as a preprocessing step in speech recognition to segment an audio stream into homogeneous regions where each region can be handled in a different manner.

Audio classification research has focused in two main areas, namely in developing numerous audio features and classification techniques. For example, while Mel-frequency cepstral coefficients (MFCCs) have become the dominant feature representation in speech recognition, they do not capture pitch and timbre information which is also important for representing general audio sounds. In [1], time and frequency-based features are extracted to represent perceptual features such as loudness, pitch, brightness, bandwidth, and harmonicity, while in [2] a combination of perceptual and cepstral features are used.

Various classification techniques have also been explored in parallel. In [3], speech/music classification using a Gaussian maximum *a posteriori* (MAP) estimator, a GMM, a spatial partitioning scheme based on k-d trees, and a nearest neighbor classifier are compared. In recent years, SVMs have been shown to offer improved performance over these previous classification techniques [2].

EBW transformations have been used extensively in the speech recognition community [4], specifically as a discriminative training technique to estimate model parameters of Gaussian mixtures. Given an initial model and input data, [5] derives an explicit formula to measure the gradient steepness required to estimate a new model via the EBW transformations. This gradient measurement is an alternative to likelihood to describe how well the initial model explains the data.

In [6] we redefined the likelihood ratio test, typically used for unsupervised segmentation tasks, with this measure of gradient steepness. We showed that our EBW segmentation method offered improvements over two standard techniques. In this paper, we further demonstrate that the EBW transformations appear to be a general technique to explain the quality of a model used to represent the data. Specifically, we use these transformations to develop a novel audio classification algorithm, which is able to outperform both the GMM likelihood and SVM techniques.

The following section provides background on the EBW transformations, followed by the EBW classification algorithms 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. Motivation of using EBW Transformations

Given some input data, there are many different approaches used to calculate how well a model represents this data. One common approach is to calculate the likelihood, that is $p(data|model)$. Another method is to calculate the gradient, as shown in Figure 1. Given an initial model for our data and an objective function, we can estimate a new model for our data 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. A steep slope indicates the initial model does not fit the data well, while a flat slope indicates the initial model is a good fit for the data. The EBW transformations provide solutions to estimate this new model, and also provide a measure of the gradient steepness to explain the quality of the initial model to fit the data.

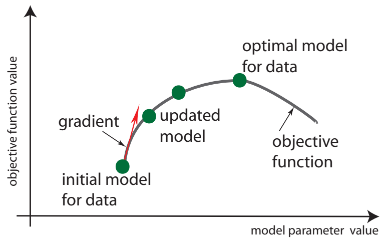

Figure 1: EBW Model Update Graph

2.2. Derivation of EBW Transformations

The EBW procedure involves continuous transformations that can be described as follows. Assume that data $X = (x_1, ...x_M)$, from frames 1 to M, is drawn from a Gaussian mixture model θ^k, with each component j parameterized by the following mean and variance parameters $\lambda_j^k = \{\mu_j^k, \sigma_j^k\}$. Let us define the probability of frame $x_i \in X$ given mixture component j as $p(x_i|\lambda_j^k) = z_{ij}^k = \mathcal{N}(\mu_j^k, (\sigma_j^k)^2)$. 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 formulas to re-estimate model parameters $\lambda_j^k(D) = \{\mu_j^k(D), \sigma_j^k(D)\}$ as:

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

$$(\hat{\sigma}_j^k)^2 = \hat{\sigma}_j^k(D)^2 = \frac{\sum_{i=1}^M c_{ij}^k x_i^2 + D \ \ (\mu_j^k)^2 + (\sigma_j^k)^2}{\sum_{i=1}^M c_{ij}^k + D} - (\hat{\mu}_j^k)^2 \tag{2}$$

Here D is a large constant chosen such that the objective function increases with each iteration, that is $F(\hat{z}_{ij}^k) \geq F(z_{ij}^k)$.

Using EBW transformations (1) and (2) such that $\lambda_j^k \rightarrow \hat{\lambda}_j^k(D)$ and $z_{ij}^k \rightarrow \hat{z}_{ij}^k$, [5] derives a linearization formula between $F(\hat{z}_{ij}^k)$ and $F(z_{ij}^k)$ for large D as:

$$F(\hat{z}_{ij}^k) - F(z_{ij}^k) = T_{ij}^k / D + o(1/D) \tag{3}$$

Here T measures the gradient required to adapt initial model λ_j to data x_i, or equivalently how well the data is explained by the initial model λ_j. A large value in T means the gradient to adapt the initial model to the data is steep and $F(\hat{z}_{ij})$ is much larger than $F(z_{ij})$. Thus the data is much better explained by the updated model $\hat{\lambda}_j(D)$ compared to the initial model λ_j. However a small value in T indicates that the gradient is relatively flat and $F(\hat{z}_{ij})$ is close to $F(z_{ij})$. Therefore, the initial model λ_j is a good fit for the data. In the next section, we derive our EBW classification technique using both sides of Equation 3.

3. Classification

Given a set of class models $\Theta = \{\theta_1, \theta_2, \ldots, \theta_K\}$, the goal of classification is to categorize an ensemble of input frames $X = \{x_1, \ldots, x_M\}$, where $x_i \in R^d$, as belonging to one of these models. Below we present the standard GMM likelihood classification method and then discuss three different classifiers derived from the EBW transformations.

3.1. GMM Likelihood

GMM likelihood classifiers are commonly used for various audio classification tasks [3]. Assume that each frame, x_i, is drawn from a mixture of N gaussians where z_{ij}^k is the likelihood of frame x_i given component j from GMM k and w_j^k the *a priori* weight of component j. Positing that each of the x_i vectors are independent and identically distributed, we define the log-likelihood of X given model θ_k by $F(z_{1:M}^k)$ as follows:

$$F(z_{1:M}^k) = p(X|\theta_k) = \sum_{i=1}^M \log \sum_{j=1}^N w_j^k z_{ij}^k \tag{4}$$

Given an input sample X, we compute how well the data is modeled by each class θ_k and choose the class θ^* which has the maximum likelihood. In other words:

$$\theta^* = \arg\max_{\theta^k} F(z_{1:M}^k) \tag{5}$$

In the next section we will redefine the likelihood criterion with our measure of EBW gradient steepness.

3.2. EBW-T

Instead of calculating the likelihood of data X belonging to model θ_k, we can measure this via the T value in Equation 3, similar to [6]. Let us define c_{ij}^k as:

$$c_{ij}^k = z_{ij}^k \frac{\delta}{\delta z_{ij}^k} F(z_{ij}^k) = \frac{z_{ij}^k w_j^k}{\sum_{l=1}^N w_l^k z_{il}^k}. \tag{6}$$

Using Equation 3 and the objective function for $F(z_{1:M}^k)$ given in Equation 4, [5] derives a closed-form expression for $T_{1:M}^k$ for large D as follows:

$$T_{1:M}^k = \sum_{j=1}^N \left\{ \sum_{r=1}^d \frac{\{\sum_{i=1}^M c_{ij}^k [(x_{ir} - \mu_{rj}^k)^2 - (\sigma_{rj}^k)^2]\}^2}{2(\sigma_{rj}^k)^4} \right\} + \sum_{j=1}^N \left\{ \sum_{r=1}^d [\frac{\sum_{i=1}^M c_{ij}^k (x_{ir} - \mu_{rj}^k)}{\sigma_{rj}^k}]^2 \right\} \tag{7}$$

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 class can be written as:

$$\theta^* = arg \min_{\theta^k} T_{1:M}^k \tag{8}$$

Note that Equation 3 holds only for large D. In the next section, we analyze a classification performance as we vary D.

3.3. EBW-F

We can also use the formula given in the right side of Equation 3 for classification. Here D is a constant chosen in the EBW model re-estimation formulas, given by Equations 1 and 2. If D is very large then training is very slow (but stable) but if D is too small model re-estimation may not increase the objective function on each iteration.

Thus, given an input sample X, the best class model θ_k is the one which has the smallest increase in likelihood given the updated model $F(\hat{z}_{1:M}^k)$ relative to the likelihood given the initial model $F(z_{1:M}^k)$. In other words the decision rule for the best model is:

$$\theta^* = arg \min_{\theta_k} \ \ F(\hat{z}_{1:M}^k) - F(z_{1:M}^k) \ \ \times D \tag{9}$$

3.4. EBW-SVM

In [4], Valtchev et. al shows that using a phone-specific D instead of a global value allows for better updated phoneme models. Similarly, we investigate the accuracy within each class using a class-specific value of D. Let us imagine taking 4 EBW-F classifiers with different D values, $D1$ to $D4$. If we apply $D1$ to input sample X we will get a score each of the K different classes, same for $D2$, $D3$, etc. We can construct a feature vector of these scores as:

$$fv = [D_{1,1}(X), D_{1,2}(X), \ldots D_{2,1}(X), \ldots D_{4,K}(X)] \tag{10}$$

Here $D_{1,1}(X)$ is the EBW-F score from Equation 9 that classifier $D1$ gives to assigning X to class 1, and $D_{1,2}(X)$ the score of assigning X to class 2. Given the output scores of different EBW-F D classifiers, we want to learn which optimal D classifier scores to weight more heavily in predicting a class.

Given a training set $\{(fv_1, c_1), \dots (fv_n, c_n)\}$ which consists of a set of feature vectors fv_i and corresponding class labels, an SVM learns the hyperplane $\mathbf{w} \cdot \mathbf{fv} - b = 0$ which best divides the data. For each class, we can think of the SVM as learning the appropriate weights for each score in order to maximize the separation margin. Or more intuitively the SVM learns which optimal D classifiers to weight more in classifying an input sound.

4. Experiments

4.1. Corpus

We perform classification experiments on the Computers in the Human Interaction Loop (CHIL) Isolated Acoustic Event data set. This database has been collected by the University Polytechnic of Catalonia (UPC) for their Acoustic Event Detection and Classification tasks [7]. The set is divided into 3 sessions, with 10 participants per session. Sounds are recorded in a closed room using 16 different microphone types. At each session, each participant takes a different place in the room and records isolated acoustic events from 14 different classes as indicated in Table 1. To match the classification experiments done in [8], we only use 12 classes, excluding the unknown and door opening classes.

Acoustic Event	Label	Acoustic Event	Label
Knock	kn	Door open	do
Door close	dc	Steps	st
Chair moving	cm	Spoon (cup jingle)	cl
Paper work	pw	Key jingle	kj
Keyboard typing	kt	Phone ringing/music	pr
Applause	ap	Cough	co
Laugh	la	Unknown	un

Table 1: CHIL Acoustic Events and Corresponding Labels

In our experiments, the data is sampled at 16kHz, and then windowed to 20ms frames with a 10ms overlap. We compare classifier performance using two different types of features. Our first feature set consists of 19 dimensional MFCCs. Our second feature set uses a combination of perceptual features similar to [8], namely short-time energy, zero crossing rate, subband energy spectral flux, in combination with the MFCCs.

We use Session 1 and Session 2 of the CHIL corpus for training. We train the GMM and EBW classifiers to find the optimal number of mixture components, as well as the optimal global and class-specific D values. For the SVM, 5-fold cross validation is performed to find the best kernel parameters, as well as the best C which represents a tradeoff between minimizing training error and maximizing classifier generalization. We found that a polynomial kernel was best for the EBW-SVM while an exponential RBF [2] was used for the baseline SVM.

5. Results

Table 1 shows the classification results for the baseline and EBW classifiers under both feature sets, with the best classifier under each feature set highlighted in bold.

Classifier	MFCCs	MFCCs+Perc
GMM Baseline	88.91	91.88
SVM Baseline	92.43	93.04
EBW-T	89.82	91.19
EBW-F	90.27	93.04
EBW-SVM	**92.64**	**94.78**

Table 2: Accuracies for EBW and Baseline Classifiers

The EBW-T classifier outperforms the GMM for MFCC features but the opposite is true when perceptual features are also used. However, with an optimal global D value, the EBW-F classifier outperforms both EBW-T and GMM for both feature sets. To explain these results further, let us observe the tradeoff between between EBW-F accuracy and choice of D in Figure 2. Notice that for very large D the EBW-F classifier accuracy approaches that of the EBW-T. As we make D smaller and train the updated model quicker, we are able to still get an appropriate estimate for the updated model while still allowing the objective function to increase. It is particularly beneficial to quickly update the initial model if the slope of the objective function is relatively flat.

At the optimal D, the EBW-F outperforms the EBW-T and GMM. As shown by Equation 9, EBW-F captures the difference between the likelihood of a data given the initial model and the likelihood with a model estimated from the current data sequence being classified, while the GMM just calculates the former. Since the GMM does not take into account model error which can be present, model re-estimation via EBW using the current data is able to correct for this initial model error, and explains why EBW-F outperforms the GMM. We find that with the MFCCs, a higher value of D is preferred but with the MFCCs+Perc, a slightly lower value of D is preferred. Since EBW-T is defined for large D, this explains the performance difference between the EBW-T and GMM for the two features.

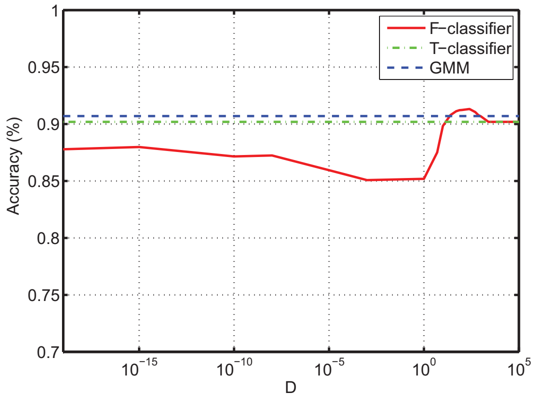

Figure 2: Accuracy vs. D for EBW-F Classifier

If we take D too small then we train our models too quickly and do not increase the value of the objective function on each iteration. Therefore we would expect that the EBW-F accuracy should continue to decrease for smaller D. However, Figure 2 shows that accuracy decreases for small D but then increases for very small D. To explain this factor, if we take D very small, then we can re-write Equations 1 and 2 as independent of D:

$$\hat{\mu}_j^k = \hat{\mu}_j^k(D) = \frac{\sum_{i=1}^M c_{ij}^k x_i}{\sum_{i=1}^M c_{ij}^k} \tag{11}$$

$$(\hat{\sigma}_j^k)^2 = \hat{\sigma}_j^k(D)^2 = \frac{\sum_{i=1}^M c_{ij}^k x_i^2}{\sum_{i=1}^M c_{ij}^k} - (\hat{\mu}_j^k)^2 \tag{12}$$

As D becomes smaller, the re-estimated model $\hat{\lambda}_j^k(D)$ is less influenced by original model λ_j^k. However, the updated means and variances are weighted by c_{ij}^k from Equation 6, and those c_{ij}^k which have higher likelihood z_{ij}^k are weighted more. Thus, for very small D the classifier accuracy increases as we put less weight on the poor model re-estimation and more emphasis on the initial likelihood z_{ij}^k. Thus the EBW-F score moves closer to the the GMM likelihood classifier, which in influenced entirely by z_{ij}^k. As we will see below, we can obtain better accuracy within some classes with a small D, where the initial likelihood is emphasized more.

Instead of using a global D, we also looked to combine EBW-F classifiers with different class-specific D values. Figure 3 shows that the highest accuracy for 5 different classes is achieved by a different D. This means that the slope of the objective function differs for various classes. Therefore each class prefers a different rate, captured by D, to estimate the updated model. For example, if the slope is relatively flat when re-estimating the initial model, a larger value of D is preferred to train models quicker and better estimate the updated model.

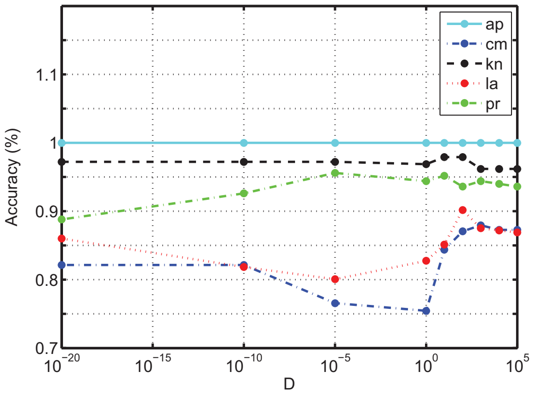

Figure 3: Class Accuracy vs. D

As shown by Table 2, when we take the EBW-F scores for different D values as features to an SVM, we find that the EBM-SVM outperforms not only the baseline SVM but also outperforms classifiers used in [8] on the same data set.

Finally, we analyze in more detail the classification performance of the EBW-SVM classifier. Figure 4 shows a bubble plot confusion matrix for this classifier. The classes are grouped according to general acoustic properties. The radii are linearly proportional to the error rate, with the largest circle representing 12.97% error. The figure shows that the classes are most easily confused with other classes that have similar acoustic properties. Generally, sounds which have loud period bursts knocking are very distinct and have high accuracies. However, harmonic sounds and soft period bursts have more varied acoustic properties and have much lower accuracies.

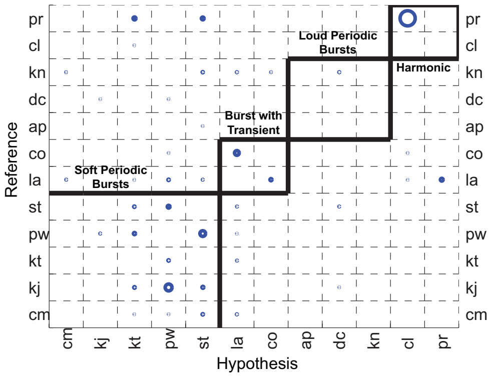

Figure 4: Classification bubble plot confusion matrix

6. Conclusions

In this paper, we expanded on our work from [6], showing that the EBW Transformations are a general technique to explain the quality of a model used to represent the data. Specifically, we found that our EBW-F classifier outperformed the GMM while the EBW-SVM technique offered improvements over both the SVM and GMM methods. In the future, we would like to apply the EBW methods to speaker verification, where GMM is currently the dominant approach. In addition, we would like to explore other approaches to measuring gradient steepness.

7. Acknowledgements

This work is sponsored by the T-Party Project, a joint research program between MIT and Quanta Computer Inc., Taiwan.

8. References

[1] E. Wold, T. Blum, D. Keislar, and J. Wheaton, "Content-Based Classification, Search and Retrieval of Audio," *IEEE Multimedia Magazine*, vol. 3, no. 1, July 1996.

[2] G. Guo and S.Z. Li, "Content-Based Audio Classification and Retrieval by Support Vector Machines ," *IEEE Transactions on Neural Networks*, vol. 14, no. 1, February 2003.

[3] E. Scheirer and M. Slaney, "Construction and Evaluation of a Robust Multifeature Speech/Music Discriminator," in *Proc. ICASSP*, 1997.

[4] V. Valtchev, J. J. Odell, P. C. Woodland, and S. J. Young, "MMIE Training of Large Vocabulary Speech Recognition Systems," *Speech Communication*, vol. 22, 1997.

[5] D. Kanevsky, "Extended Baum Transformations for General Functions," in *Proc. ICASSP*, 2004.

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

[7] A. Waibel et al., "CHIL: Computers in the Human Interaction Loop," in *WIAMIS*, 2004.

[8] A. Temko, R. Malkin, C. Zieger, D. Macho, C. Nadeu, and M. Omologo, "Acoustic Event Detection and Classification in Smart-Room Environment: Evaluation of CHIL Project Systems," in *The IV Biennial Workshop on Speech Technology*, November 2006.