LEARNING-BASED AUDITORY ENCODING FOR ROBUST SPEECH RECOGNITION

Yu-Hsiang Bosco Chiu, Bhiksha Raj and Richard M. Stern

Department of Electrical and Computer Engineering and Language Technologies Institute
Carnegie Mellon University, Pittsburgh PA 15213 USA
{ychiu,bhiksha,rms}@cs.cmu.edu

ABSTRACT

This paper describes ways of speeding up the optimization process for learning physiologically-motivated components of a feature computation module directly from data. During training, word lattices generated by the speech decoder and conjugate gradient descent were included to train the parameters of logistic functions in a fashion that maximizes the *a posteriori* probability of the correct class in the training data. These functions represent the rate-level nonlinearities found in most mammalian auditory systems. Experiments conducted using the CMU SPHINX-III system on the DARPA Resource Management and Wall Street Journal tasks show that the use of discriminative training to estimate the shape of the rate-level nonlinearity provides better recognition accuracy in the presence of background noise than traditional procedures which do not employ learning. More importantly, the inclusion of conjugate gradient descent optimization and a word lattice to reduce the number of hypotheses considered greatly increases the training speed, which makes training with much more complicated models possible.

Index Terms— automatic speech recognition, discriminative training, auditory models, data analysis

1. INTRODUCTION

Our auditory system serves a wide range of tasks in our daily life. One particular function of the auditory system which is of the most importance is to encode and recognize the diversity of environmental sounds – human speech, birds singing and even market noises. To be able to accomplish this task, an essential property is the formation of loudness perception among different frequencies that form a particular instance of sound input. While the mechanisms by which the auditory system encodes the loudness of sound remain open to debate (*e.g.* [1, 2]), one can argue that it has been tuned, at some level, for better recognition of sounds, including human speech.

It is often hypothesized that the various features of human sound perception, such as the frequency resolution of the cochlea [3], nonlinear compressive effects of the middle ear [4], simultaneous and temporal masking effects [5] etc. aid or enhance human ability to recognize speech, particularly in the presence of noise. Researchers have therefore attempted to model many of these features in automatic speech recognition systems as well, with varying degrees of detail (*e.g.* [6, 7]).

In our previous work, we proposed a top-down process for robust speech recognition [8] that optimizes a physiologically-motivated *feature computation* procedure for recognition. But rather than using a continuous speech recognition system for the optimization, we built a simple phoneme/state classifier to avoid computational complexity. While even this simple procedure was observed to result in significant improvement in speech recognition accuracy obtained with the optimized feature computation, it also raises the question of the optimality of the representation. We were specifically concerned with the extent to which our optimization procedure would continue to produce improvements in recognition accuracy as the speech recognition system became more complex as more and more training data became available.

To address this problem, the tied states from an LVCSR system are used for optimizing the parameters of the rate-level nonlinearity using discriminative gradient descent procedures. The models in turn are optimized for the features obtained using maximum likelihood training. The two steps – model optimization for the features, and feature optimization for the models continue iteratively. The whole process, however, can become very computationally expensive as the model complexity scales up with more and more training data available. In order to overcome the computational complexity problem, we investigate conjugate gradient descent and the use of a reduced recognition lattice obtained from a decoder such that only the derivatives over possible candidate states are considered for the discriminative feature optimization.

The rest of this paper is organized as follows. In Sec. 2 we describe the feature computation scheme we employ. In Sec. 3 we describe the algorithm that learns the relevant parameters of the feature computation. In Sec. 4 we describe experiments conducted on the DARPA Resource Management (RM) and Wall Street Journal (WSJ) databases. Finally, we summarize our conclusions in Sec. 5.

2. FEATURE COMPUTATION WITH LOUDNESS EQUALIZATION AND RATE-LEVEL NONLINEARITY

We parameterize speech signals using the feature computation scheme proposed by Chiu and Stern [8, 9, 10]. The overall scheme is shown in Fig.1. Each analysis frame of the incoming speech signal is analyzed by a fast Fourier transform. Each frequency component is then weighted by the frequency-dependent gain function derived from an equal-loudness curve that characterizes a psychophysical estimate of the loudness response of the auditory system [11]. The resulting spectrum is reduced to a smaller number of Mel-spectral values using conventional Mel-frequency weighting [6]. Each Mel-spectral value is compressed logarithmically, and these compressed Mel-spectral values are passed through a subsequent sigmoidal nonlinearity that represents the physiologically-observed rate-level nonlinearity. This nonlinearity is given by

$$x_i[t] = \frac{\alpha[i]}{1 + exp(w_1[i] \cdot y_i[t] + w_0[i])} \quad (1)$$

This work was sponsored by NSF Grants IIS-0420866 and IIS-0916918 and by the Draper Laboratory.

where $y_i[t]$ is the i^{th} log Mel-spectral value and $x_i[t]$ is the corresponding sigmoid-compressed value of frame t. In [10] a set of parameters of the nonlinearity were obtained by fitting to physiological measurements followed by some subsequent manual refinement. These estimated values were $\alpha[i] = 0.05$; $w_0[i] = 0.613$; $w_1[i] = -0.521$ for $\forall i$. The compressed values are then projected down to a 13-dimensional cepstral vector by a conventional discrete cosine transform (DCT).

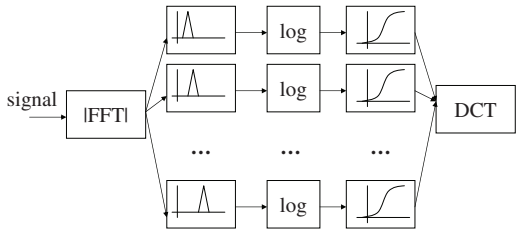

Fig. 1. *Summary of feature computation scheme.*

3. LEARNING THE NONLINEARITY

To combine the acoustic model training and feature extraction more effectively, we make use of the Gaussian models from the speech recognizer (SPHINX-III) to train the system. More specifically, the acoustic models obtained from a traditional Baum-Welch training procedure are used for estimating the nonlinearity parameters by using a gradient descent algorithm with a maximum-mutual information (MMI) criterion. This is illustrated in Fig. 2.

The procedure for optimizing the nonlinearity is as follows. Let $\boldsymbol{\mu}_{mC}$ be the mean vector and $\boldsymbol{\sigma}_{mC}$ be the covariance matrix for the m^{th} Gaussian with weight $\boldsymbol{w}_{mC}$ in the Gaussian mixture density of any sound class C. For our purposes, individual tied states in the recognizer are considered to be sound classes. The likelihood of any vector $\mathbf{s}$ as computed by the distribution for a particular sound class is assumed to be given by the density function $\sum_m w_{mC} N(\mathbf{s}|\boldsymbol{\mu}_{mC}, \boldsymbol{\sigma}_{mC})$.

The posterior probability of any sound class C given a specific observation $\mathbf{s}$ is given by

$$P(C|\mathbf{s}) = \frac{P(\mathbf{s}|C)P(C)}{\sum_{C'} P(\mathbf{s}|C')P(C')} = \frac{P(\mathbf{s}|C)}{\sum_{C'} P(\mathbf{s}|C')} = \frac{\sum_m w_{mC} N(\mathbf{s}|\boldsymbol{\mu}_{mC}, \boldsymbol{\sigma}_{mC})}{\sum_{C'} \sum_m w_{mC'} N(\mathbf{s}|\boldsymbol{\mu}_{mC'}, \boldsymbol{\sigma}_{mC'})} \quad (2)$$

with the priors for each sound class $P(C)$ assumed to be equal.

We assume that we have a collection of training data, and that for each analysis frame of this data we know the identity of the correct sound class. The parameters of the feature computation are initialized with the values from [10]. Each recording from the training data is parameterized using these initial values. Cepstral mean subtraction(CMS) is performed on every training recording in order to remain consistent with the processing that is performed in a complete speech recognition system.

Let $\mathbf{s}_{u,t}$ be the feature vector obtained for the t^{th} analysis frame of the utterance u, and let $C_{u,t}$ be the sound class that the corresponding segment of speech belongs to. The overall accumulated posterior probability of the entire training data is given by

$$P = \prod_{u,t} \frac{\sum_m w_{mC_{u,t}} N(\mathbf{s}_{u,t}|\boldsymbol{\mu}_{mC_{u,t}}, \boldsymbol{\sigma}_{mC_{u,t}})}{\sum_C \sum_m w_{mC} N(\mathbf{s}_{u,t}|\boldsymbol{\mu}_{mC}, \boldsymbol{\sigma}_{mC})} \quad (3)$$

The parameters of the of the sigmoidal nonlinearity in the feature computation are now iteratively optimized to maximize $\log(P)$.

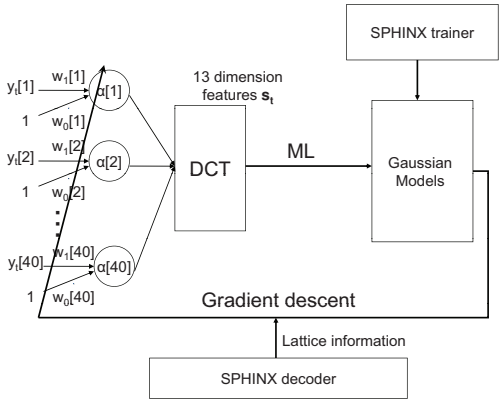

Fig. 2. *The integrated system that refines the extracted features.*

3.1. Estimating the sound-class distribution parameters

The model parameters $\{\boldsymbol{w_C}, \boldsymbol{\mu}_C, \boldsymbol{\sigma}_C\}$ for each sound class are obtained using the same maximum likelihood criterion employed by the speech recognizer after each gradient descent step.

3.2. Estimating the sigmoidal parameters

The parameters for the logistic function $\mathbf{F}= \{\boldsymbol{\alpha}, \boldsymbol{w_0}, \boldsymbol{w_1}\}$ are estimated to maximize $log(P)$ using the conjugate gradient descent approach [12]. Taking the derivative of the objective function with respect to $\mathbf{F}$, the nonlinear parameters are updated as follows:

$$\begin{aligned} \boldsymbol{\alpha}^{new} &= \boldsymbol{\alpha}^{old} + 0.001 step \frac{\partial \log P}{\partial \boldsymbol{\alpha}}, \\ \boldsymbol{w}_0^{new} &= \boldsymbol{w}_0^{old} + step \frac{\partial \log P}{\partial \boldsymbol{w}_0}, \\ \boldsymbol{w}_1^{new} &= \boldsymbol{w}_1^{old} + 0.2 step \frac{\partial \log P}{\partial \boldsymbol{w}_1} \end{aligned} \quad (4)$$

Note that the inverse of the Hessian matrix is approximated by the weighting shown above such that the convergence rate for each individual set of parameters is roughly the same. After each step, which is performed on both the clean training data and noisy test data, the model parameters are updated on a clean training set only. After the objective function has converged only the nonlinear parameters $\mathbf{F}=\{\boldsymbol{\alpha}, \boldsymbol{w_0}, \boldsymbol{w_1}\}$ are retained for the feature extraction process, and the model parameters are retrained using the Baum Welch algorithm, using the clean training data.

The entire learning algorithm is summarized in Algorithm 1. Here $\boldsymbol{y}_{u,t}$ represents the log Mel-spectral vector corresponding to the t^{th} analysis window of the u^{th} utterance, $\boldsymbol{s}_{u,t}$ is the feature vector computed from it, and $C_{u,t}$ is the corresponding sound class.

3.3. Reducing computational complexity by using a word lattice

As mentioned earlier, a complete MMI solution, that compares each "true" class label for *all* competing classes can become extremely computationally expensive. More specifically, the amount of computation for calculating derivatives for each set of parameters at each iteration will be on the order of $\Theta(KLMN)$, where K is the number

Input: $\mathbf{F}$,$\{(\boldsymbol{y}_{u,t}, C_{u,t}),\ u = 1..U, t = 1..T_U\}$
Output: $\mathbf{F}$

$r \leftarrow \frac{\partial \log P}{\partial \mathbf{F}}$
$s \leftarrow Mr$ where M is the weighting shown in Eq.4
$d \leftarrow s$
$\delta_{new} \leftarrow r^T d$
while *not converged* **do**
  $j \leftarrow 0$
  $\gamma \leftarrow \sigma_0$
  while $j < j_{max}$ **do**
    Compute feature vector $\{\mathbf{s}_{1,1}, ..., \mathbf{s}_{U,T_U}\}$ using Eq.(1) and DCT with CMS
    Estimate $\{\boldsymbol{w}_C, \boldsymbol{\mu}_C, \boldsymbol{\sigma}_C\}\ \forall C$ on clean training set
    Compute $\log(P)$ using Eq.(3) on both clean and noisy training set
    $\eta \leftarrow [\frac{\partial \log P}{\partial \mathbf{F}}]^T d$
    if $j \neq 0$ **then**
      $\gamma \leftarrow \gamma \frac{0.5\eta}{\eta' - \eta}$
    end
    $\mathbf{F}_{new} \leftarrow \mathbf{F}_{old} + \gamma d$
    $\eta' \leftarrow \eta$
    $j \leftarrow j + 1$
  end
  $r \leftarrow \frac{\partial \log P}{\partial \mathbf{F}}$
  $\delta_{old} \leftarrow \delta_{new}$
  $\delta_{mid} \leftarrow r^T s$
  $s \leftarrow Mr$ where M is the weighting shown in Eq.4
  $\delta_{new} \leftarrow r^T s$
  $\beta \leftarrow \frac{\delta_{new} - \delta_{mid}}{\delta_{old}}$
  if $\beta \leq 0$ **then**
    $d \leftarrow s$
  else
    $d \leftarrow s + \beta d$
  end
end

Algorithm 1: *Algorithm for learning the parameters of the sigmoidal nonlinearity where* $\sigma_0 = 0.05$ *and* $j_{max} = 5$.

of cepstral dimensions, L is the number of channels, M is the number of Gaussian mixtures, and N is the number of sound classes. As the number of sound classes and Gaussian mixtures increases with the complexity of the speech recognizer, the amount of computation becomes too large for machines to handle. To overcome this problem, rather than using all sound classes as the denominator for the MMI updates, we use word lattice as shown in Fig. 3 to include only the sound classes that are determined to be "competitors" by the decoder as the competing classes. The word lattices used in our experiments were generated by running the SPHINX decoder on the training data using the same initial acoustic model that had been used for generating the forced alignment. Once obtained, these lattices remained fixed throughout the optimization process.

4. EXPERIMENTAL RESULTS

Experiments were run on the RM and WSJ databases to evaluate the proposed method. The SPHINX-III continuous-density HMM-based system was used for all experiments. HMMs with 1000 tied states, each modeled by a mixture of 8 Gaussians were trained for baseline recognition experiments. The feature extraction employed a 40-filter

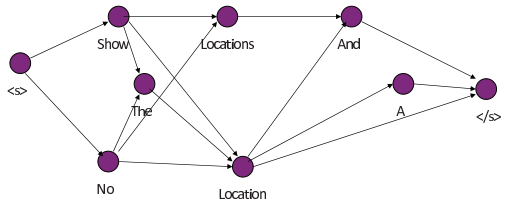

Fig. 3. *Example of a word lattice to reduce the computational complexity by including only decoder-identified candidates as the competing classes.*

Mel filter bank covering the frequency range 130 Hz – 6800 Hz.

Our rate-level sigmoidal nonlinearity was trained on both clean and noise-corrupted speech, with the noisy data obtained by adding pink noise from the NOISEX-92 database to clean training data at an SNR of 10 dB. To evaluate the dependence of accuracy on model complexity, 1000 and 2000 tied states for the RM database and 4000 tied states for the WSJ database were generated by forced-aligning clean training data using previously-trained models as class labels. The noisy testing sets were created by artificially adding babble noise from NOISEX-92 and noises recorded live in market, theater, and restaurant environments.

To avoid local optima, the optimization of the nonlinearity parameters with larger numbers of tied states was initialized using the trained results from the 1000 tied-state case. Once the parameters of the feature computation module were learned, the feature computation module was employed to derive features from a *clean* version of the RM training set, from which the HMM model parameters were retrained.

Recognition experiments were run on speech corrupted to various SNRs by a variety of noises. Results for the WSJ database were obtained by training SPHINX-III with 4000 tied states, each modeled by a mixture of 16 Gaussians, with the nonlinearity parameters learned from training using the RM database. Figures 4 and 5 describe the dependence of recognition accuracy on analysis type and model complexity for the RM and WSJ corpora, respectively. Recognition accuracy is defined (as usual) as 100% minus the conventional word error rate including insertion, deletion, and substitution errors. These plots also include baseline accuracy using MFCC coefficients (triangles), and the accuracy obtained using the fixed RL nonlinearity derived as in [10] (squares). The latter processing also employed equal-loudness weighting. We note that none of the noises used in these experiments were used to train the rate-level nonlinearity.

It can be seen by comparing the square to triangular symbols of Figs. 4 and 5 that the use of the physiologically- and perceptually-motivated equal-loudness weighting and baseline rate-level nonlinearity (without learning) greatly improve noise robustness. The use of automatically-learned parameters (diamond and circular symbols), however, provides further improvements in performance, especially when the complexity of the model increases with the increase of the number of tied states. While the results obtained using multiple Gaussian mixtures are very close to the performance obtained with single Gaussians, training with much more data or with more complicated models is only possible with the lattice implementation. The use of the hypothesis lattice as in Fig. 3 speeds up the entire training process by a factor of about 2.5.

We also compared the convergence speed of the original gradient descent approach with the conjugate gradient descent that we deploy in the present paper. As shown in Fig. 6, conjugate gradient descent provides a speed of convergence that is is much faster than before.

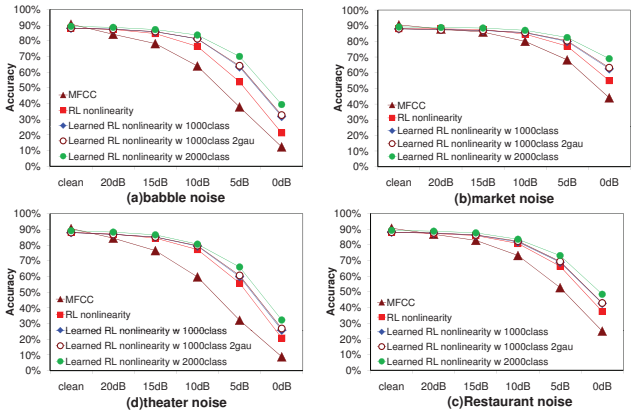

Fig. 4. *Comparison of recognition accuracy in the presence of four types of background noise using the RM corpus. WER under clean: MFCC: 9.45%, RL nonlinearity: 11.88%, RL nonlinearity from learning with 1000 tied states and 1 Gaussian: 12.03%, 2 Gaussian: 11.97%, with 2000 tied states: 10.53%*

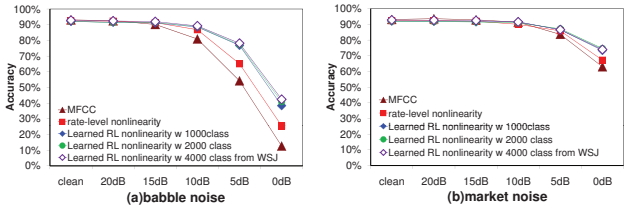

Fig. 5. *Comparison of recognition accuracy in the presence of two types of background noise on the WSJ corpus. WER under clean: MFCC: 6.91%, RL nonlinearity: 7.66%, RL nonlinearity learned from RM, 1000 tied states 7.94%, 2000 tied states: 7.96%, from WSJ 4000 tied states: 7.25%*

The major improvement in the speed of convergence seen in Fig. 6 comes from the use of conjugate gradient descent which reduces the number of total iterations. This is enabled by two factors: first, with conjugate gradient descent each search direction is linearly independent from its previous one [12]. In addition, with conjugate gradient descent the step size of the search (η) is automatically determined at each step to optimize the conversion speed while keeping the optimization process stable. In this way, we not only approach the maximum point (or at least a local maximum) more directly without wasting effort, but we can also achieve convergence using larger step sizes while maintaining a stable optimization process.

Another important component of the speedup is the use of the lattice structure as described in Sec. 3.3 which reduces the processing time per iteration by reducing the number of competing candidate hypotheses that need to be considered. In empirical comparisons of the processing time with and without the lattice representation we observed that the use of the word lattice reduces the processing time for the gradient descent step of each iteration of the optimization by a factor of approximately 2.5.

5. CONCLUSIONS

We have presented an algorithm for learning physiologically-motivated components of feature extraction for optimal speech recognition. In general we observe (as before) that the use of learning in feature extraction results in consistently improved speech recognition over conventional feature computation without learning. In the present paper we describe further improvements in processing speed obtained through the use of conjugate gradient descent (which reduces the number of iterations needed to achieve convergence) and the use of a word lattice (which reduces the processing time per iteration by reducing the number of candidate hypotheses). These improvements enable training and evaluation on larger corpora with more detailed acoustic models.

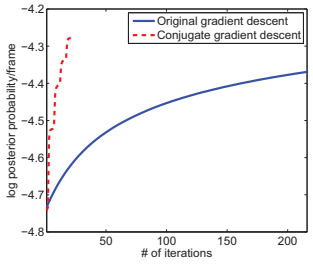

Fig. 6. *Comparison of convergence speed for original gradient descent and conjugate gradient descent.*

6. REFERENCES

[1] E.D. Young and M.B. Sachs, "Representation of steady state vowels in the temporal aspects of the discharge patterns of populations of auditory-nerve fibers," *J. Acoust. Soc. Am.*, 1979, vol. 66, pp. 1381–1403.

[2] N.F. Viemeister, "Auditory intensity discrimination at high frequencies in the presence of noise," *Science*, 1983, vol. 221, pp. 1206–1208.

[3] J. Volkmann, S.S. Stevens and E.B. Newman, "A Scale for the Measurement of the Psychological Magnitude Pitch," *J. Acoust. Soc. Am.*, 1937 vol. 8(3), pp. 208–208.

[4] M.B. Sachs and P.J. Abbas, "Rate versus level functions for auditory- nerve fibers in cats: Tone-burst stimuli," *J. Acoust. Soc. Am.*, 1974 vol. 56, pp. 1835–1847.

[5] E. Zwicker, "Temporal effects in simultaneous masking and loudness," *J. Acoust. Soc. Am.*, 1965 vol. 38, 132-141.

[6] S. Davis and P. Mermelstein, "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences," *IEEE Trans. on Acoust., Speech and Signal Processing*, 1980, vol. 28, pp. 357–366.

[7] S. Seneff, "A Joint synchrony/mean-rate model of auditory speech processing," *J. Phonetics*, 1988, vol. 15, pp. 55–76.

[8] Y.-H. Chiu, B. Raj and R. Stern, "Towards Fusion of Feature Extraction and Acoustic Model Training: A Top Down Process for Robust Speech Recognition," *Proc. ICSLP*, Sep. 2009.

[9] Y.-H. Chiu and R. Stern, "Analysis of Physiologically-Motivated Signal Processing for Robust Speech Recognition," *Proc. ICSLP*, Sep. 2008.

[10] Y.-H. Chiu and R. Stern, "Minimum variance modulation filter for robust speech recognition," *Proc. ICASSP*, Apr. 2009.

[11] E. Terhardt, "Calculating virtual pitch," *Hearing Research*, 1979, 1:155-182.

[12] J.R. Shewchuk, "An Introduction to the Conjugate Gradient Method Without the Agonizing Pain," *Technical Report: CS-94-125*, 1994.