MIXTURE INPUT TRANSFORMATIONS FOR ADAPTATION OF HYBRID CONNECTIONIST SPEECH RECOGNIZERS

Victor Abrash

Speech Technology And Research Laboratory
SRI International
Menlo Park, California 94025
U.S.A.
victor@speech.sri.com

ABSTRACT

We extend the input transformation approach for adapting hybrid connectionist speech recognizers to allow multiple transformations to be trained. Previous work has shown the efficacy of the linear input transformation approach for speaker adaptation [1][2][3], but has focused only on training global transformations. This approach is clearly suboptimal since it assumes that a single transformation is appropriate for every region in the acoustic feature input space, that is, for every phonetic class, microphone, and noise level. In this paper, we propose a new algorithm to train mixtures of transformation networks (MTNs) in the hybrid connectionist recognition framework. This approach is based on the idea of partitioning the acoustic feature space into R regions and training an input transformation for each region. The transformations are combined probabilistically according to the degree to which the acoustic features belong to each region, where the combination weights are derived from a separate acoustic gating network (AGN). We apply the new algorithm to nonnative speaker adaptation, and present recognition results for the 1994 WSJ Spoke 3 development set. The MTN technique can also be used for noise or microphone robust recognition or for other nonspeech neural network pattern recognition problems.

1. INTRODUCTION

Hybrid connectionist hidden Markov model (HMM) speech recognizers model the state-dependent acoustic observation densities $b_j(\boldsymbol{y}_t)$ with neural networks rather than Gaussian mixtures. In particular, the connectionist version of DECIPHER [4][5] utilizes a multilayer perceptron (MLP) that is trained to classify input speech into one of N phonetic classes. The MLP input $\boldsymbol{Y_t} = \{\boldsymbol{y}_{t-4}, \ldots, \boldsymbol{y}_t, \ldots, \boldsymbol{y}_{t+4}\}$ consists of 9 frames of 26-dimensional cepstral feature vectors $\boldsymbol{y}_t$. The multiframe input window $\boldsymbol{Y_t}$ enables the MLP to utilize local acoustic context in computing the desired output probabilities. The MLP has a single sigmoidal hidden layer with 1000 units and a sigmoidal output layer with N units, one for each context-independent phone class in the recognizer. Each MLP output unit $o_j(t)$ computes an estimate of $P(phone_j|\boldsymbol{Y_t})$, the posterior probability of the j-th phone given the acoustic speech feature window $\boldsymbol{Y_t}$ at time t. For recognition, the HMM observation densities $b_j(\boldsymbol{Y_t})$ are estimated by the scaled likelihoods obtained by dividing the MLP outputs by the prior phone probabilities.

The MLP is a statistical model whose parameters (weights) are estimated using speech data drawn according to some unknown but fixed distribution that we want to model. This data is called the *training set*, and consists of speech utterances from many speakers with known orthographic transcriptions. We denote the speaker-independent (SI) MLP and its weight values after training by the label, "SI MLP". Testing utterances may be drawn from the same or a different distribution, which we denote as the *matched* or *mismatched* condition. Recognition performance may be seriously degraded when there is mismatch between training and testing, as often happens in real applications. Common sources of error include speaker (e.g., outlier or nonnative speakers), environmental (noise), channel, or microphone mismatches.

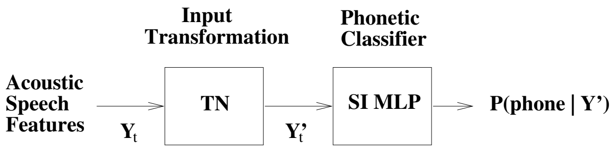

Figure 1. Adaptation using a single linear transformation network (TN).

Previous speaker adaptation work [1][2][3] has shown that recognition accuracy can be improved significantly for both native and nonnative speakers by linearly transforming the MLP input $\boldsymbol{Y_t}$ using an additional linear transformation network (TN) as shown in Figure 1. During adaptation, the SI MLP weights are fixed and only the TN weights are modified. This technique successfully reduced the recognizer word error rate by 15–35%, depending on the recognition task and type of mismatch.

This approach is clearly suboptimal since only a global (linear) transformation is estimated. It assumes that a single transformation is appropriate for every region in the acoustic feature input space, that is, every phonetic class, microphone, and noise level. Experience with Gaussian mixture HMM systems [6][7] clearly shows that the use of multiple transformations improves speaker adaptation performance, where each transformation is valid over a local region of the acoustic feature space and the regions are chosen according to phonetic similarity criteria.

In this paper, we propose a new algorithm to train MTN in the hybrid connectionist recognition framework. Each transformation in the mixture is optimized over a local region of the acoustic feature space. The transformations are linearly combined using soft decision boundaries derived from an AGN. We apply the new algorithm to nonnative speaker adaptation, and present recognition results for the 1994 WSJ Spoke 3 development set. The new technique may also be applicable for noise or microphone robust recognition.

2. MIXTURES OF TRANSFORMATION NETWORKS

In this work, we extend the input transformation approach for adapting hybrid connectionist speech recognizers from single transformations to mixtures of transformations. This new approach is based on the idea of partitioning the acoustic feature

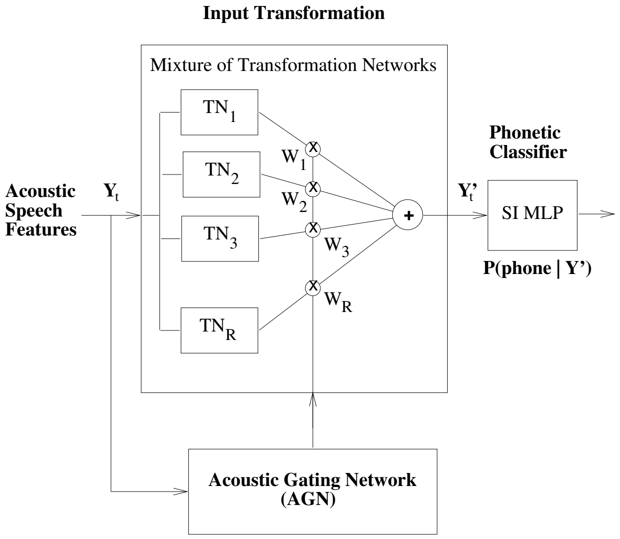

Figure 2. The mixture of transformation networks (MTN) architecture for adaptation. Each transformation network (TN) is trained on a local region of the acoustic feature space as specified by the acoustic gating network (AGN). The contribution of TN_r to the final transformation is weighted according to the probability that the central frame of the speech feature window Y_t belongs to the r-th acoustic region.

space into R regions and training an input transformation for each region. The transformations are combined probabilistically according to the degree to which the input feature window belongs to each acoustic region. No hard decisions are made regarding input pattern region membership. If an input pattern has nonzero membership in more than one region, more than one TN will contribute to the overall transformation.

As shown in Figure 2, the new architecture consists of an AGN with R outputs $w_r(t)$ and a set of R transformations that are combined linearly according to weights $w_r(t)$.

2.1. Acoustic Gating Network

The function of the acoustic gating network is to partition the input acoustic feature space into meaningful regions. These partitions can be related to phonetic class or be computed using automatic data-clustering techniques such as vector quantization or Gaussian mixtures. Soft region classification decisions are preferable in order to increase the robustness of the overall system. To this end, we design the AGN so that the r-th output $w_r(t)$ computes the probability that the center frame of the MLP input window belongs to the r-th acoustic region,

$$w_r(t) = P(region = r|\boldsymbol{Y_t}) \tag{1}$$

where $0 \leq w_r(t) \leq 1$ and $\sum_{r=1}^{R} w_r(t) = 1.0$ for all t.

The overall organization and function of the MTN is determined by the AGN design, for example, the number and type of regions and the form and training of the AGN.

Since this work involves speaker adaptation we divide the acoustic feature space according to regions of phonetic similarity [6][7]. In particular, we reuse our SI MLP to create $R = 48$ regions that have a one-to-one correspondence with the phones in our system, or $R = 6$ where we have grouped the MLP outputs into six broad phone classes.

Alternative AGN designs enable the MTN to be applied to other recognition robustness problems. For example, a microphone-independent system could be built using a microphone classification AGN and training the transformations with data from R microphones, or we could design a noise robust recognizer using one transformation for each of R SNR levels.

Note that the AGN may be designed ahead of time, or its parameters can be trained during adaptation. The AGN may use acoustic features different from those used by the MLP; for example, microphone or SNR-dependent features may be more useful for region selection in channel or noise robustness applications [8].

2.2. Transformation Network Architecture

The transformation component of the MTN architecture is very flexible, and many types of TN can be trained. Each TN receives the same input. The R TN output vectors are weighted by $w_r(t)$, the probability that the input belongs to acoustic region r, and summed to create a new MLP input vector $\boldsymbol{Y_t'}$ with the same dimensionality as the original input $\boldsymbol{Y_t}$.

In this work, we train linear TNs as in [1][2][3] to create a piecewise linear overall transformation function. Each TN is initialized to the identity transform, guaranteeing that initial MTN performance remains the same as the SI recognition model. Although in principle non-linear TNs can be used to create a piecewise non-linear transformation, we continue to use linear TNs because of their favorable initialization properties.

In particular, we will present speaker adaptation recognition results using two TN architectures.

The first TN architecture (denoted "TN9") transforms the entire 9-frame MLP input window as a single unit, so the transformed MLP input $\boldsymbol{Y_t'}$ is computed as

$$\boldsymbol{Y_t'} = \sum_{r=1}^{R} w_r(t)\, \mathrm{TN9_r}(\boldsymbol{Y_t}), \tag{2}$$

where $\mathrm{TN9_r}(\cdot)$ is the transformation for the r-th acoustic region and $w_r(t)$ is computed by the AGN. The overall MTN using this type of TN is denoted by "MTN9 (R)".

The second TN architecture (denoted "TN1") transforms each frame in the 9-frame MLP input window independently but identically. In this case, the transformed MLP input is $\boldsymbol{Y_t'} = \{\boldsymbol{y'}_{t-4}, \ldots, \boldsymbol{y'}_t, \ldots, \boldsymbol{y'}_{t+4}\}$, where each column $\boldsymbol{y'}_{t+i}$ is computed as

$$\boldsymbol{y'}_{t+i} = \sum_{r=1}^{R} w_r(t)\, \mathrm{TN1_r}(\boldsymbol{y'}_{t+i}) \tag{3}$$

for $i \in [-4, 4]$. The overall MTN using this type of TN is denoted by "MTN1 (R)".

2.3. Training

Each TN is trained only for data falling into its acoustic region with nonzero probability. This is accomplished by weighting the error vector propagated back from the phonetic classifier MLP by the AGN output weight $w_r(t)$ to form an error vector specific to each TN.

The adaptation algorithm can be summarized as follows:

- The weights of the R TN networks TN_r are each initialized to the identity transform I.
- For supervised adaptation, enrollment data is segmented using forced Viterbi recognition and the data's known orthographic transcription to obtain phonetically labeled target values for the MLP. Recognition output can be used in place of the known transcriptions for unsupervised adaptation.

- The combined MTN and MLP system is trained to maximize relative entropy between the MLP outputs and their target values.
- Keeping the SI MLP weights fixed, the MTN weights are updated using standard error backpropagation techniques. The error vector for each TN_r in the MTN is weighted by $w_r(t)$, the input pattern membership in acoustic region r computed by the AGN. Currently, the AGN weights are not adapted.
- Training halts when phonetic frame classification performance on an independent cross-validation dataset ceases to improve.

2.4. Comparison with Other Algorithms

The MTN architecture is similar to both Huang's CDNN architecture [9] and to Neumeyer's POF algorithm [8]. Compared to CDNN, we implement a piecewise linear transformation rather than piecewise nonlinear, and use soft decision boundaries rather than hard VQ decisions for combining the sub-networks. Compared to POF, we use an MLP to create acoustic partitions based on the estimated phonetic identity of the input rather than Gaussian mixtures trained to estimate the probability of a speech frame falling into a VQ region. Both of these algorithms implement regression networks that map one set of speech features to another based on an arbitrary spectral distortion measure independently of the recognizer error criteria, and require stereo training data. The MTN architecture is part of the MLP phonetic classifier and is trained to optimize phonetic frame classification performance, which is related to recognition word error.

Waterhouse [10] recently introduced a different method to adapt connectionist recognition systems using multiple input transformations. The recurrent neural network (RNN) acoustic model was duplicated and a single linear input transformation was paired with each duplicate copy. The parallel models were then combined at the level of the RNN output layers by using a nonlinear gating function that was randomly initialized. The entire network was then trained using the hierarchical mixture of experts (HME) training algorithm. This approach can be characterized as a mixture of adapted acoustic models, where each model is adapted using a single transformation network. Because of the linear increase in computation with R, only two or four acoustic regions were used. Performance did not improve relative to a single transformation because of the poor quality of the acoustic regions and because the regions had no relation to the microphone robustness task being tackled.

3. EXPERIMENTAL RESULTS

SRI's DECIPHER speech recognizer was used for all experiments. Supervised adaptation experiments were performed on the male subset of the 1994 Spoke 3 (S3) development set of the Wall Street Journal (WSJ) speech corpus [11]. This data consists of read speech from nonnative speakers of American English, and was divided into 40 adaptation and 40 test sentences. The standard 5,000-word, closed-vocabulary bigram language model was used for recognition.

A speaker-independent MLP with 234 inputs, 1000 hidden units, and 48 outputs was trained with about 17,000 utterances from 140 male speakers. The MLP input was a 9-frame input window of 26-dimensional mean-normalized cepstral vectors consisting of the first 12 cepstral coefficients, cepstral energy, and their first derivatives. Each cepstral vector was further normalized to be zero mean and unit variance.

The 40 enrollment sentences were further divided into two subsets, three-quarters for adaptation and one-quarter for cross-validation. The MTNs were trained as described in Section 2.3.

The MTNs in these experiments used $R = 1$, $R = 6$, or $R = 48$ transformations, corresponding respectively to a global TN, one TN for each of six broad phone classes (silence, vowels, stops, nasals, fricatives, and approximates), or one TN for each context-independent phone class in the recognizer. For ease of implementation, the AGN was implemented as a duplicate of the SI MLP. For the broad class experiments, the MLP outputs were summed over all phones in a given class to yield the probability of that class. Two types of transformation network (TN1 and TN9) were investigated as described in Section 2.2. Thus, the experiments are labeled either MTN1-(R) or MTN9-(R), depending on the type of TN used and the number of acoustic regions.

Table 1 summarizes the recognition word error (Werr) rates obtained with these MTN configurations. After adaptation, we observe a significant decrease (15-33%) in recognition error in all cases.

The dramatic improvement using MTN1-(1) demonstrates the importance of adaptation, and that even a small number (700) of adaptation parameters can drastically reduce the mismatch between testing and training conditions. The next big jump in performance occurs when using $R > 1$, indicating that individual acoustic regions require different transformations and pointing out the inadequacy of a global adaptation model. Going from $R = 6$ to $R = 48$ gains relatively little additional improvement. This is probably due to AGN output noisiness (region misclassifications), which was partially smoothed out in the broad class case; the six broad classes were hand-selected so the AGN would make relatively few inter-(broad)class misclassification errors, encouraging individual TNs to be trained on disjoint regions of the acoustic feature space.

Comparing the MTN1-(R) and MTN9-(R) experiments, we observe that MTN9-(1) outperforms MTN1-(R) for any R, although the difference between MTN1-(48) and MTN9-(1) is small. This shows the importance of modeling between-frame cepstral correlation by the TN9 architecture, although we can approximate this effect through the use of multiple single-frame transformations when each TN is trained on the relatively more self-consistent set of cepstral vectors within each acoustic region. Finally, the MTN approach also improves performance for the TN9 transformation architecture, although at high cost; for $R = 48$ it used ten times more adaptation parameters than existed in the rest of the acoustic model. The MTN algorithm is

Experiment	Adaptation Parameters	Werr (%)	Improvement vs. SI (%)
SI MLP	–	33.6	–
MTN1 -(1)	702	28.4	15.5
MTN1-(6)	4,212	25.2	25.1
MTN1-(48)	33,696	24.4	27.5
MTN9-(1)	54,990	23.6	29.8
MTN9-(48)	2,639,520	22.5	33.2

Table 1. Recognition word error rates for the mixture of transformation networks (MTN) adaptation architecture for nonnative speakers from the 1994 WSJ S3 development set. MTN1 experiments used single-frame transformations; MTN9 experiments transformed the entire input window. The numbers in parentheses are the number of phonetic regions used.

robust against over-parameterization.

4. CONCLUSIONS

We have presented a new approach for adapting hybrid connectionist speech recognizers using mixtures of linear input transformation networks (MTNs). This approach is based on the idea of partitioning the acoustic feature space into R regions and training an input transformation for each region. The transformations are combined probabilistically according to the degree to which the acoustic features belong to each region. The combination weights are derived from a separate acoustic gating network (AGN), whose design determines the function of the overall transformation.

MTN adaptation overcomes the limitations of previous input transformation approaches that assumed that one transformation was optimal over all regions of the acoustic feature space and therefore estimated only a single global transformation. In this work, the AGN was designed to partition the acoustic space according to regions of phonetic similarity.

Using two different input transformations, we have applied the MTN algorithm to a nonnative speaker adaptation task (the 1994 WSJ Spoke 3 development set). For both types of input transformation, our results show improved recognition accuracy after adaptation relative to the single transformation approach. We have also demonstrated that if more input context is available to the transformation it will increase adaptation performance.

The new technique may also be applicable for noise or microphone robust recognition. The general approach is also applicable to nonspeech neural network pattern recognition problems.

5. ACKNOWLEDGEMENTS

We gratefully acknowledge partial support for this work from ARPA through Office of Naval Research Contract N00014-94-C-0181.

REFERENCES

[1] H. Franco, V. Abrash, M. Cohen, A. Sankar, and M. Weintraub, "Hybrid HMM/MLP speech recognition," in *ARPA Artificial Neural Network Technology 1994 Program Review*, (Key West, FL), December 6–8 1994.

[2] V. Abrash, H. Franco, A. Sankar, and M. Cohen, "Connectionist speaker normalization and adaptation," in *Proceedings of the 4th European Conference on Speech Communication and Technology*, (Madrid, Spain), pp. 2183–2186, Sept. 1995.

[3] J. Neto, L. Almeida, M. Hochberg, C. Martins, L. Nunes, S. Renals, and T. Robinson, "Speaker-adaptation for hybrid hmm-ann continuous speech recognition system," in *Proceedings of the 4th European Conference on Speech Communication and Technology*, (Madrid, Spain), pp. 2171–2174, Sept. 1995.

[4] M. Cohen, H. Franco, N. Morgan, D. Rumelhart, and V. Abrash, "Hybrid neural network/hidden markov model continuous-speech recognition," in *Proceedings of the International Conference on Spoken Language Processing*, (Banff, Alberta, Canada), pp. 915–918, Oct. 1992.

[5] H. Bourlard and N. Morgan, *Connectionist Speech Recognition: A Hybrid Approach*. Massachusetts: Kluwer Academic Publishers, 1994.

[6] V. Digalakis, R. Rtischev, and L. Neumeyer, "Fast speaker adaptation using constrained estimation of Gaussian mixtures," *IEEE Transactions on Speech and Audio Processing*, vol. 3, pp. 357–366, Sept. 1995.

[7] C. Leggetter and P. Woodland, "Maximum likelihood linear regression for speaker adaptation of continous density HMMs," *Computer Speech and Language Processing*, vol. 9, pp. 171–186, 1995.

[8] L. Neumeyer and M. Weintraub, "Probabilistic optimum filtering for robust speech recognition," in *Proceedings of the International Conference on Acoustics, Speech and Signal Processing*, (Adelaide, AU), pp. 417–420, apr 1994.

[9] X. Huang, "Minimizing speaker variation effects for speaker-independent speech recognition," in *DARPA Speech and Natural Language Workshop*, (Harriman, NY), pp. 191–196, Feb. 1992.

[10] S. Waterhouse, D. Kershaw, and T. Robinson, "Smoothed local adaptation of connectionist systems," in *Proceedings of the International Conference on Spoken Language Processing*, (Philadelphia, PA), 1996.

[11] F. Kubala *et al.*, "The hub and spoke paradigm for CSR evaluation," in *ARPA Spoken Language Technology Workshop*, (Plainsboro, NJ), pp. 9–14, Mar. 1994.