Hermitian-Based Hidden Activation Functions for Adaptation of Hybrid HMM/ANN Models

Sabato Marco Siniscalchi[1,3] *Jinyu Li*[2] *and Chin-Hui Lee*[3]

[1] Faculty of Telematics Engineering, Kore University of Enna, Enna, Italy
[2] Microsoft Corporation, Redmond, WA, 98052 USA
[3]School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA, USA
marco.siniscalchi@unikore.it, jinyli@microsoft.com, chl@ece.gatech.edu

Abstract

This work is concerned with speaker adaptation techniques for artificial neural network (ANN) implemented as feed-forward multi-layer perceptrons (MLPs) in the context of large vocabulary continuous speech recognition (LVCSR). Most successful speaker adaptation techniques for MLPs consist of augmenting the neural architecture with a linear transformation network connected to either the input or the output layer. The weights of this additional linear layer are learned during the adaptation phase while all of the other weights are kept frozen in order to avoid over-fitting. In doing so, the structure of the speaker-dependent (SD) and speaker-independent (SI) architecture differs and the number of adaptation parameters depends upon the dimension of either the input or output layers. We propose an alternative neural architecture for speaker-adaptation to overcome the limits of current approaches. This neural architecture adopts hidden activation functions that can be learned directly from the adaptation data. This adaptive capability of the hidden activation function is achieved through the use of orthonormal Hermite polynomials. Experimental evidence gathered on the Wall Street Journal Nov92 task demonstrates the viability of the proposed technique.

Index Terms: Connectionist speech recognition systems, Neural networks, Adaptation algorithms, Speech recognition.

1. Introduction

In connectionist speech recognition systems [1], an ANN is used to estimate the state emission probabilities of a set of hidden Markov models (HMMs) using Bayes' rule. These state emission probabilities typically are related to phone (posterior) probabilities. In this framework, the temporal structure of the speech pattern is modeled by the HMMs, whereas the ANN is used to model the acoustic signal conditioned on the Markov process. Although several neural architectures have been proposed to tackle different speech recognition tasks, such as recurrent neural networks (e.g., [2, 3]) and time-delay neural network [4], the feed-forward multi-layer perceptron (MLP) is by far the most popular as it provides an attractive compromise between recognition rate, recognition speed, and memory resources. It has been demonstrated that high performance phone recognizers can be built by arranging sets of MLPs into multi-stage configurations, e.g., [5, 6, 7]. The leading idea of such an approach is to generate a hierarchical integration of phonetic and lexical knowledge during the estimation of the phone posterior probability values. Recently, there has been a resurgence of interest in HMM/ANN due to the success of deep neural networks (DNN) [8]. The current research mainly focuses on how to train a DNN, without fully investigating other supporting areas, such as speaker adaptation.

This work is concerned with speaker adaptation of hybrid HMM/MLP systems. In this context, most successful speaker adaptation techniques consist of augmenting the neural architecture with a linear transformation network connected to either the input [9] or the output layer [10]. The weights of this additional linear layer are learned during the adaptation phase while all of the other weights are kept frozen in order to avoid over-fitting. This approach has several disadvantages that will be described in Section 2. In this paper, we propose the adoption of a MLP architecture that has the capability of modifying the shape of its hidden activation functions [11] in order to address the speaker adaptation problem of hybrid HMM/ANN models while overcoming the limits of the standard technology. This adaption capability is obtained through the use of a weighted sum of R orthonormal Hermite functions, which has already proven successful in speech classification and recognition tasks [11]. The hidden activation functions of the proposed architecture are automatically learned from the training data but can change shape during the adaptation phase while all other neural parameters (i.e., weights) are kept frozen. We evaluate our approach on the Wall Street Journal Nov92 task. Experimental evidence demonstrates the effectiveness of the proposed speaker adaption technique for LVCSR.

The rest of the paper is organized as follows: A survey of related works is given in section 2. In Section 3 the Hermitian-based neural architecture is first presented, and details about the Hermite regression formula are provided. The experimental setup is described in Section 4 along with a discussion of the results. Finally, concluding remarks are given in Section 5.

2. Related Work

In conventional state-of-the-art large vocabulary conversational speech recognition (LVCSR) systems, training material from a large number of speakers is collected to develop a single model that can deal with the variance across dialects, speaking styles, etc. In doing this, LVCSR systems should generalize well to any particular speaker. Unfortunately, a drop in performance is typically observed when moving from a speaker-dependent (SD) to a speaker-independent (SI) automatic speech recognition (ASR) system due to inter-speaker variability. Model adaptation techniques, e.g., [12, 13], have thus been devised to find algorithms that adapt an ASR system to a specific speaker using limited, but representative data. The adaptation can be carried out either on-line or off-line, in a supervised or unsupervised way, and it should ultimately result in a new, adapted system that improves

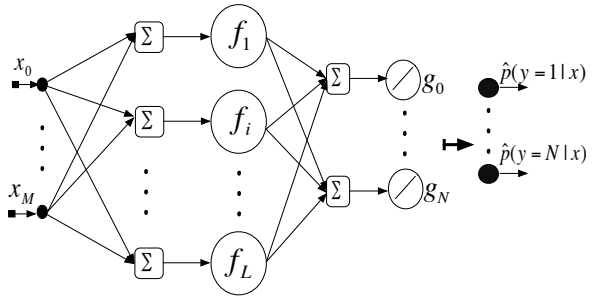

Figure 1: *Neural architecture with Hermitian hidden activation functions.*

ASR performance, i.e., giving us a lower word error rate, for a given speaker.

The simplest supervised speaker adaptation techniques performed in off-line fashion for hybrid HMM/MLP systems would modify the weights of the SI MLP using some specific speaker data; nevertheless, such an approach typically leads to over-fitting on the adaptation material when the amount of adaptation patterns for a given speaker is very little, for instance, a single spoken utterance for a given speaker. More successful solutions consist in augmenting the structure of the connectionist component of the SI ASR system by adding a linear transformation network either to input or to the output layer. For example, a linear input network (LIN) can be added to the input layer of the SI MLP to map SD input vectors to the SI ASR system [9]. This mapping (i.e., the LIN layer) is trained minimizing the error at the output of the neural architecture while keeping all other MLP weights frozen. Gemello et al. [10] proposed the use of a linear hidden layer (LHN), which consists in adding a linear transformation network before the output layer. The key idea for such an approach is that the outputs of an internal layer represent discriminative features of the input pattern suitable for the classification performed at the output of the MLP. Two main disadvantage of these techniques are a) the speaker-dependent (SD) neural architecture differs from and is more complex than the speaker-independent (SI) one, and b) the number of adaptation parameters cannot be set according to the amount of the available adaptation data, since it is bound to the number of inputs, or outputs of the SI MLP.

There exist other techniques for adapting connectionist ASR systems, and a review can be found in [9]. Nonetheless, it seems, to the best of the authors' knowledge, that these techniques have never been applied to LVCSR systems. Finally, it is worthy of mention that a preliminary investigation on DNN adaptation was carried out in [14]. The key idea was to fit the feature-space maximum-likelihood linear regression technique into the DDN framework. This method also adds one additional layer and freezes other parameters.

3. Hermitian-Based MLP

The neural architecture used in this work is a one-hidden-layer feed forward MLP, and is designed for estimating class posterior probabilities in a discriminative way (see Figure 1). In the following, $x = (x_1, ..., x_M)$ indicates an input feature vector with M components, and $y \in \mathcal{Y}$ is the phone label of x, where $\mathcal{Y}$ is the set of N classes. The MLP estimates the conditional probability of a phone label y given an input vector x using a nonlinear model of the form

$$\hat{p}_k = \hat{p}(y = k|x) = \frac{\exp g_k}{\sum_{i=1}^{N} \exp g_i}, \tag{1}$$

where g_k is the linear activation function of the kth output, and it is given by

$$g_k = \sum_{j=1}^{L} w_{kj}^{(2)} f_j \left(\sum_{i=1}^{M} w_{ji}^{(1)} x_i \right). \tag{2}$$

Here $w_{kj}^{(2)}$ and $w_{ji}^{(1)}$ denote weights in the second and first layer, respectively; f_j is the activation function of jth hidden neuron. There exist several types of activation functions that can be used in hidden neuron. If the Hermite polynomials are chosen, f_j is a linear combination of Hermite functions of the form

$$f_j(z) = \sum_{r=1}^{R} c_{jr} h_r(z), \tag{3}$$

where R is the degree of the Hermite polynomial, and $h_r(z)$ is the rth Hermite orthonormal function. The c_{jr} coefficients are learned during the training phase along with the $w_{kj}^{(2)}$ and $w_{ji}^{(1)}$ weights. The orthonormal Hermite polynomials will be described later along with their first-order derivatives.

In this work all of the hidden neurons employ a Hermite polynomial of the same degree. The softmax function is employed at the output layer, and the cross-entropy error function is chosen as function to be minimized during the training phase [15]. The reader is referred to [11] for details about the main differences between the proposed Hermitian MLP implementation and other similar architectures [16, 17].

3.1. Hermite Regression Formula

The orthonormal Hermite polynomials, $H_r(z)$, are defined over the interval $(-\infty, \infty)$, and there exist several ways to formally introduce them; in this study, we follow [17], and the orthonormal Hermite function of order r is then given by

$$h_r(z) = \alpha_r H_r(z) \phi(z), \tag{4}$$

where

$$\alpha_r = (r!)^{-\frac{1}{2}} \pi^{\frac{1}{4}} 2^{-\frac{r-1}{2}}, \tag{5}$$

$$\phi(z) = \frac{1}{\sqrt{2\pi}} e^{-\frac{z^2}{2}}, \tag{6}$$

$$\begin{aligned} H_r(z) &= (-1)^r e^{z^2} \frac{\partial^r}{\partial z^r} \left(e^{-z^2} \right) \\ &= 2z H_{r-1}(z) - 2(r-1) H_{r-2}(z), \\ &\quad r > 1, \quad H_0(z) = 1, \quad H_1(z) = 2z. \end{aligned} \tag{7}$$

The recursive nature of the Hermite polynomials makes the computation of the first-order derivative of Eq. 3 very simple and therefore its use as activation function is quite appealing. The first-order derivative is given by

$$\begin{aligned} \frac{\partial}{\partial z} f(z) &= \sum_{r=1}^{R} c_{jr} \frac{\partial}{\partial z} \left(h_r(z) \right) \\ &= \sum_{r=1}^{R} c_{jr} \left[(2r)^{\frac{1}{2}} h_{(r-1)}(z) - z h_r(z) \right]. \end{aligned} \tag{8}$$

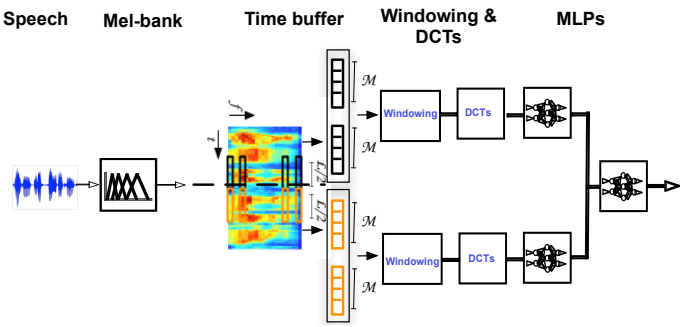

Figure 2: *The connectionist architecture used in this work. It is based on the split temporal context idea and three MLPs, as proposed in [5].*

Classical back-propagation algorithm with cross-entropy error function is used to train the neural networks. Several iterations of training of the whole system followed by realignment of labels are performed. The training algorithm begins from uniformly segmented phonemes.

3.2. Adaptation of neural networks

The goal is to change the shape of the Hermitian-based hidden activation function to better fit the speaker-specific features. Only the coefficients in Eq. 3 are adapted while all other parameters of the MLPs are kept frozen. Classical stochastic back-propagation algorithm with cross-entropy error function (J) is used to adapt the coefficients of Eq. 3. The change of the overall error with respect to the generic c_{jr} coefficient is computed as follows

$$\frac{\partial J}{\partial c_{jr}} = \frac{\partial J}{\partial f_j} \frac{\partial f_j}{\partial c_{jr}} = \left[\sum_{k=1}^{N} w_{kj}^{(2)} \delta_k \right] h_r(z). \qquad (9)$$

where δ_k is the *sensitivity* of the kth output unit (see [15]), which describes how the overall error changes with the unit's net activation function. In contrast to previous adaptation techniques proposed for connectionist ASR systems [10], the structure of our connectionist system is not modified.

4. Experiments

In the following sections we present the experimental setup and discuss the results.

4.1. Experimental Setup

Corpus: All experiments were conducted on the 5,000-word WSJ0 (5k-WSJ0) task. The training material from the SI84 set (7077 utterances, or 15.3 hours of speech from 84 speakers) was separated into a 6877-utterance training set and a 200-sentence cross-validation (CV) set. Evaluation was carried out on the Nov92 evaluation data with 330 utterances from 8 speakers. Adaptation data were obtained from the standard *si_et_ad* set of WSJ0, which consists of 8 speakers with 40 utterances per speaker. The phoneme set consisted of 41 phonemes (40 phoneme classes, and 1 garbage class).

Input Features and Connectionist Architecture: The feature extraction made use of long temporal context [5]. First, Mel filter bank energies were obtained in conventional way. Temporal evolutions of critical band spectral densities were taken around each frame. The context of 31 frames (310 ms) around the current frame was selected. This context was split into 2 halves: Left and Right Contexts. Both parts were processed by discrete cosine transform to de-correlate and reduce dimensionality (i.e., only certain dimensions were kept). Two single-layer MLP with Hermitian activation hidden function were trained to produce phone-state posterior probabilities for both context parts. A third Hermitian MLP functioned as a merger and produced the final set of phoneme-state posterior probabilities. Figure 2 shows the hierarchical neural architecture of the proposed connectionist model. It should be noted that this architecture was first introduced in [5] where Sigmoidal hidden activation functions have employed in all MLPs. In this work, Sigmoidal activation functions were replaced with Hermitian ones.

Evaluation criteria: The increment in classification error on the CV part during training was used as stopping criterion to avoid over-training. The number of neurons in hidden layer of all MLPs used was 800. The number of Hermitian coefficients was set to 10. All experiments reported in this paper use this number of hidden layer neurons unless stated otherwise.

4.2. Experimental Results

Table 1 summarizes the frame accuracy rates (FARs) at a state-phone level of the Hermitian-based MLP (HMLP) on the training, cross-validation (CV), and test set. The HMLP correctly classifies 77.8% of all training frames. Furthermore, it achieves a FAR of 76.1% and 76.0% on the validation and evaluation sets, respectively. The latter shows that over-fitting has been somehow avoided. LVCSR is carried out using the state-phone posterior estimates in a hybrid HMM/ANN system, that is, as local scores for decoding. A trigram language model (LM) is used during recognition. We refer to this LVCSR system as HMM/HMLP. Moreover, to assess the quality of the HMM/HMLP system, the performance of two connectionist HMM systems trained on the same SI-84 and evaluated on the same Nov92 test set reported in [10] are also included here for comparison. These systems are referred to as LOQ-1 and LOQ-2, respectively.

Table 2 shows word error rates (WERs) for all LVCSR systems studied in this work. Specifically, the second column of Table 2 shows the SI performance, the third column shows the performance of the adapted LVCSR systems, and the fourth column summarizes the relative improvement. The SD Loquendo results regard the WERs obtained after having applied the LIN algorithm and are given as reported in [10]. In [10], the authors also provide results for the LHN technique and for the combination of LIN and LHN. Unfortunately, these results are always given in combination with the conservative training technique, and therefore are not directly comparable with ours. Conservative training seems to mitigate the issues that arise when a neural network is adapted with new data that do not adequately represent the knowledge included in the original training data. Conservative training addresses this issue by not setting to zero the value of the targets of the missing units in the adaptation data. Applying this technique to our HMLP is out of the scope of this work.

The SI performance of the proposed HMM/HMLP system is given in the second row of Table 2, and it is equal to a WER of 6.4%. LOQ-1 and LOQ-2 achieve a WER of 8.4% and 6.5%, respectively, before adaptation. These results are reported in the third and fourth row of Table 2, respectively. This first set of results demonstrates that a reasonable SI LVCSR system can be designed using HMLPs. LIN adaptation performed over the

Table 1: *Frame accuracy rates (FARs) at state-phone level of the hierarchical structure of Hermetian MLPs on WSJ.*

System/Dataset	Train	Validation	Evaluation
HMLP	77.81	76.1%	76.0%

Table 2: *WER on the Nov92 task for several connectionist LVCSR systems. A trigram language model is used. LIN is used to perform adaptation for LOQ-1 and LOQ-2.*

System	SI	SD	Rel. Imp.
HMM/HMLP	6.4 %	5.5%	14.1%
LOQ-1	8.4 %	7.9%	5.9%
LOQ-2	6.5 %	5.6%	13.8%

LOQ-1allows a WER of 7.9%, which correspond to a relative improvement of 5.9%. The WER is reduced to 5.6% from the initial 6.5% by using LIN over LOQ-2, and this reduction corresponds to a relative improvement of 13.8%. The proposed hybrid HMM/HMLP LVCSR system attains a WER of 5.5% using the adaptation technique described in Section 3.1. This result corresponds to a relative improvement of 14.1% and demonstrates the viability of the proposed technique. As shown in Table 2, the proposed hybrid HMM/HMLP LVCSR system is significantly better than the LOQ-1 system, and slightly superior to the LOQ-2 system. Nevertheless, the major gain in adopting our technique is that the structure of the MLP needs not to be modified to perform adaptation. Finally, the number of parameters involved in the proposed adaptation procedure is 24,000 (i.e., 800 x 10 parameters for each of the three HLMP) much lower than that involved in the LIN procedure, which is 85,995 for LOQ-1 and 307,125 for LOQ-2.

5. Conclusion

ANNs have been used in the speech community for long time, and several architectures based on ANNs have been proposed to accomplish different speech tasks. Nonetheless, very little attention has been paid on the rule of the hidden activation function so far. In [11], we have shown that the choice of the non-linearity in the hidden neurons can be crucial to obtain good generalization capability and better performance. We presented some initial, yet promising studies, on adapting the ASR system by adopting hidden activation functions that can be automatically learned from the data and change shape during training. This adaptive capability is achieved through the use of a linear combination of orthonormal Hermite polynomials as hidden activation function, as shown in Eq. 3. In the proposed work, we have evaluated our Hermitian MLPs in the context of speaker adaptation for LVCSR. We have shown that connectionist architecture of the hybrid HMM/ANN speech recognition system can be adapted to a specific speaker while keeping the complexity and the structure of the SD and SI LVCSR systems equivalent with beneficial effects on the performance. Furthermore, our approach compares favorably with standard LIN technique on the same task. This line of research is being furthered by incorporating a conservative training feature into the parameter adaptation process.

6. References

[1] H. Bourlard and N. Morgan, *Connectionist speech recognition - a hybrid approach*. Norwell, MA, USA: Kluwer Academic Publisher, 1994.

[2] J. S. Bridle, "Alpha-nets: a recurrent "neural" network architecture with a hidden Markov model interpretation," *Speech Communication*, vol. 9, no. 1, pp. 83–92, 1990.

[3] A. Graves and J. Schmidhuber, "Framewise phoneme classification with bidirectional LSTM and other neural network architectures," *Neural Networks*, vol. 18, no. 5-6, pp. 602–610, 2005.

[4] K. J. Lang, A. H. Waibel, and G. E. Hinton, "A time-delay neural network architecture for isolated word recognition," *Neural Networks*, vol. 3, pp. 23–43, 1990.

[5] P. Schwarz, P. Matějka, and J. Černocký, "Hierarchical structures of neural networks for phoneme recognition," in *Proc. ICASSP*, (Toulouse, France), May 2006.

[6] S. M. Siniscalchi, T. Svendsen, and C.-H. Lee, "Towards bottom-up continuous phone recognition," in *Proc. ASRU*, (Kyoto, Japan), pp. 556–569, Dec. 2007.

[7] J. Pinto, B. Yegnanarayana, H. Hermansky, and M. Magimai-Doss, "Exploiting contextual information for improved phone recognition," in *Proc. ICASSP*, (Las Vegas, NV, USA), pp. 4449–4452, Mar./Apr. 2008.

[8] G. Dahl, D. Yu, L. Deng, and A. Acero, "Context-dependent pre-trained deep neural networks for large vocabulary speech recognition," *IEEE Trans. Audio, Speech and Language Processing*, vol. 20, no. 1, pp. 30–42, 2012.

[9] 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 *Proc. Eurospeech*, (Madrid, Spain), pp. 2171–2174, Sept. 1995.

[10] R. Gemello, F. Mana, S. Scanzio, P. Laface, and R. D. Mori, "Linear hidden transformations for adaptation of hybrid ANN/HMM models," *Speech Communication*, vol. 49, no. 10-11, pp. 827–835, 2007.

[11] S. M. Siniscalchi, T. Svendsen, S. Sorbello, and C.-H. Lee, "Experimental studies on continuous speech recognition using neural architectures with "adaptive" hidden activation functions," in *Proc. ICASSP*, (Dallas, TX, USA), pp. 4882–4885, Mar. 2010.

[12] J. L. Gauvain and C.-H. Lee, "Maximum a posteriori estimation for multivariate Gaussian mixture observation of Markov chains," *IEEE Trans. on Speech and Audio Processing*, vol. 2, no. 2, pp. 291–298, 1994.

[13] C. J. Leggetter and P. C. Woodland, "Maximum likelihood linear regression for speaker adaptation for continuous density hidden Markov models," *Computer Speech & Language*, vol. 9, no. 2, pp. 171–185, 1995.

[14] F. Seide, G. Li, X. Chen, and D. Yu, "Feature engineering in context-dependent deep neural networks for conversational speech transcription," in *Proc. ASRU*, (Hawaii, USA), pp. 24–29, Dec. 2011.

[15] C. M. Bishop, *Neural networks for pattern recognition*. New York, USA: Oxford Univ. Press, 1995.

[16] L. Ma and K. Khorasani, "Constructive feedforward neural networks using Hermite polynomial activation functions," *IEEE Trans. Neural Networks*, vol. 16, no. 4, pp. 821–833, 2005.

[17] S. Gaglio, G. Pilato, F. Sorbello, and G. Vassallo, "Using the Hermite regression formula to design a neural architecture with automatic learning of the "hidden" activation functions," in *Proc. of AI*IA*, 1999.