UNSUPERVISED SPEAKER ADAPTATION OF DEEP NEURAL NETWORK BASED ON THE COMBINATION OF SPEAKER CODES AND SINGULAR VALUE DECOMPOSITION FOR SPEECH RECOGNITION

Shaofei Xue[1], *Hui Jiang*[2], *Lirong Dai*[1]*, *Qingfeng Liu*[1]

[1]National Engineering Laboratory of Speech and Language Information Processing,
University of Science and Technology of China, Hefei
[2]Department of Electrical Engineering and Computer Science, York University, Toronto, Canada
Email: xuesf@mail.ustc.edu.cn, hj@cse.yorku.ca lrdai@ustc.edu.cn, qliu@ustc.edu.cn,

ABSTRACT

Recently, we have proposed a general adaptation scheme for deep neural network based on discriminant condition codes and applied it to supervised speaker adaptation in speech recognition based on either frame-level cross-entropy or sequence-level maximum mutual information training criterion [1, 2, 3, 4]. In this case, each condition code is associated with one speaker in data, which is thus called speaker code for convenience. Our previous work has shown that speaker code based methods are quite effective in adapting DNNs even when only a very small amount of adaptation data is available. However, we have to use a large speaker code size and complex processes to obtain the best ASR performance since good initializations of speaker codes and connection weights are very important. In this paper, we propose a method using singular value decomposition (SVD) as in [5] to initialize speaker codes and connection weights to obtain a comparable ASR performance as before but with a smaller speaker code size and much less computation complexity. Meanwhile, we have evaluated unsupervised speaker adaptation with the proposed method in large vocabulary speech recognition in the Switchboard task. Experimental results have shown that it is effective for providing well initializations and suitable in adapting large DNN models.

Index Terms— Deep Neural Network (DNN), Speaker Code, Speaker Adaptation, singular value decomposition (SVD)

1. INTRODUCTION

Speaker adaptation has been an important research topic in automatic speech recognition (ASR) for decades. Speaker adaptation techniques attempt to optimize ASR performance by transforming speaker-independent models towards one particular speaker or modifying the target speaker features to match pre-trained speaker-independent models based on a relatively small amount of adaptation data from the target speaker. In the past few decades, several successful speaker adaptation techniques have been proposed for the conventional HMM/GMM based speech recognition systems, such as MAP [6, 7], MLLR [8, 9], and CMLLR [10]. Recently, a number of speaker adaptation methods have been proposed for neural networks. Successful methods like LIN and L-HN in [11, 12] add additional layer into neural networks and alleviate the over-fitting problem to some extent. On the other hand, Hermitian-based MLP (HB-MLP) method in [13] achieves the adaptive capability through the use of new orthonormal Hermite polynomials as activation functions in NN. The feature discriminative linear regression technique in [14] and the output-feature discriminative linear regression in [15] have also been proposed to perform speaker adaptation for DNNs. Furthermore, it has proposed to use Kullback-Leibler (KL) divergence as regularization in the adaptation criterion in [16] since it forces the state distribution estimated from the adapted DNN to stay close enough to the original model to avoid over-fitting. [17] augments DNN inputs with speaker i-vector features to facilitate speaker adaptation and results in significant improvements. In our previous works [1, 2, 3, 4], several fast speaker adaptation methods for DNN and CNN based on the so-called speaker codes have been proposed, in these methods speaker codes are directly fed to various layers of a pre-trained DNN through a new set of connection weights. These methods are appealing because the connection weights can be reliably learned from the entire training data set while only a small speaker code is learned from adaptation data for each speaker. Moreover, the speaker code size can be freely adjusted according to the amount of available adaptation data. However, we have to use a large speaker code size and complex processes to obtain the best ASR performance since good initializations of speaker codes and connection weights are very important.

In this paper, we propose a method using singular value

*This work was partially supported by the National Nature Science Foundation of China (Grant No. 61273264) and the electronic information industry development fund of China (Grant No. 2013-472).

decomposition (SVD) to initialize speaker codes and connection weights to obtain a comparable ASR performance as before but using a smaller speaker code size and less computation complexity. As opposed to supervised adaptation in our previous work [1, 2, 3, 4], in this paper, we have evaluated the proposed adaptation scheme under an unsupervised speaker adaptation setting for large vocabulary speech recognition in the Switchboard task. Experimental results have shown that the proposed hybrid Speaker code and SVD method is robust to perform effective unsupervised adaptation and it may provide well initialization for adapting large DNN models.

2. ADAPTATION IN MODEL SPACE BASED ON SPEAKER CODE

In this work, we study the adaptation in model space based on speaker code method (mSA-SC) in [4] that conducts speaker adaptation in model space of DNNs . As show in Fig. 1, the total neural network consists of initial speaker-independent weights matrices and a set of new connection weights. we feed the speaker codes directly to the hidden layers and the output layer of the initial neural network through the new connection weights. In this way, speaker codes are used to adapt the speaker-independent DNNs towards new target speakers.

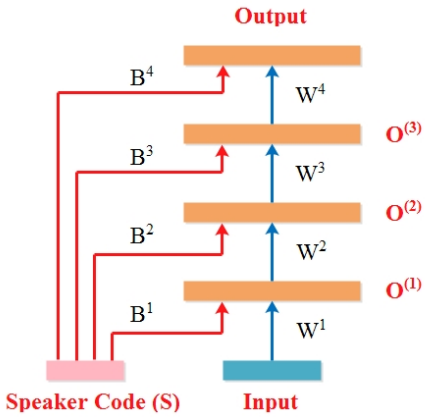

Fig. 1. Adaptation of DNNs based on speaker code.

Let us denote $\mathbf{W}^{(l)}$ as the l-th layer weights in the initial neural network that consists of n layers (including input and output layer), and $\mathbf{B}^{(l)}$ as weight matrix to connect speaker code to l-th layer in DNNs, and $\mathbf{S}^{(c)}$ stands for the speaker code specific to c-th speaker. For notational simplicity, we may expand the bias vector into the weight matrix. Each l-th layer of the neural network receives all activation output signals of the lower layer along with a speaker specific code, as follows:

$$\mathbf{O}^{(l)} = \sigma(\mathbf{W}^{(l)}\mathbf{O}^{(l-1)} + \mathbf{B}^{(l)}\mathbf{S}^{(c)}) \quad (\forall\, l) \tag{1}$$

Where $\mathbf{O}^{(l)}$ denotes outputs from l-th layers of neural networks and $\sigma(\cdot)$ stands for sigmoid based nonlinear activation function. Assume we need to adapt a well-trained DNN (represented by $\mathbf{W}^{(l)}$). In the following, we investigate how to estimate connection weights, $\mathbf{B}^{(l)}$, and speaker codes, $\mathbf{S}^{(c)}$, from training data for this adaptation scheme. Assume E denotes the objective function for DNN training or adaptation, such as frame-level cross-entropy (CE) or sequence level minimum mutual information (MMI) criterion [4]. For simplicity, we use the cross entropy criterion for adaptation. During the adaptation procedure, we only estimate $\mathbf{B}^{(l)}$ (for all l) and speaker codes $\mathbf{S}^{(c)}$ (for all speakers in the training set) using the stochastic gradient descent algorithm while keeping all $\mathbf{W}^{(l)}$ unchanged. Therefore, the derivative with respect to any element in $\mathbf{B}^{(l)}$, i.e., $B^{(l)}_{kj}$, that connects between the k-th node in the speaker code and the j-th node in l-th layer of initial neural network can be computed as:

$$\frac{\partial E}{\partial B^{(l)}_{kj}} = \frac{\partial E}{\partial O^{(l)}_{j}}(1 - O^{(l)}_{j})O^{(l)}_{j}S^{(c)}_{k} \tag{2}$$

where $S^{(c)}_{k}$ that stands for the k-th node in speaker code of c-th speaker.

Similarly, we compute the derivative of E with respect to each element of all speaker codes based on the chain rule. Since the propagation errors from all layers in the neural network contribute to the derivative of $S^{(c)}_{k}$, we need to summarize all as follows:

$$\frac{\partial E}{\partial S^{(c)}_{k}} = \frac{1}{n-1}\sum_{l=1}^{n-1}\sum_{j=1}^{J}\frac{\partial E}{\partial O^{(l)}_{j}}(1 - O^{(l)}_{j})O^{(l)}_{j}B^{(l)}_{kj}. \tag{3}$$

In learning, we first randomly initialize all $\mathbf{B}^{(l)}$ and $\mathbf{S}^{(c)}$. Next, we run several epochs of stochastic gradient descents over the training data to update $\mathbf{B}^{(l)}$ and $\mathbf{S}^{(c)}$ based on the gradients computed in eqs.(2) and (3). For speaker codes, $\mathbf{S}^{(c)}$ is only updated by data from c-th speaker. At the end, we have learned all weight matrices $\mathbf{B}^{(l)}$, which are capable of adapting the speaker-independent DNN to any new speaker given a suitable speaker code.

The next step in adaptation is to learn a speaker code for each new speaker. During this phase, only the speaker code is estimated based on eq.(3) for the new speaker from a small number of adaptation utterances while all $\mathbf{B}^{(l)}$ and $\mathbf{W}^{(l)}$ remain unchanged. After the speaker code is learned for each test speaker, the speaker code is imported into the neural network through $\mathbf{B}^{(l)}$ as in eq.(1) to compute posterior probabilities of test utterances for final recognition.

Obviously, a joint training of all parameters ($\mathbf{B}^{(l)}$, $\mathbf{S}^{(c)}$ and $\mathbf{W}^{(l)}$) using training data set as well as speaker label can generates better and more compact models. We call this method speaker adaptive train based on speaker code (SAT-SC) [4]. After learning all parameters, the adaptation process of a new speaker is the same as mSA-SC.

3. SVD BASED INITIALIZATIONS

In this section, we briefly review the basic idea of SVD and present SVD based initializations of speaker codes and connection weights for mSA-SC and SAT-SC methods.

3.1. Review of SVD

The singular value decomposition (SVD) is a factorization of a matrix, with many useful applications in signal processing and statistics. The decomposition of a matrix $\mathbf{A}$ can be described as follows:

$$\begin{aligned}\mathbf{A}_{m\times n} &= \mathbf{U}_{m\times m}\mathbf{S}_{m\times n}\mathbf{V}^T_{n\times n} \\ &\approx \mathbf{U}_{m\times k}\Sigma_{k\times k}\mathbf{V}^T_{k\times n} \\ &= \mathbf{U}_{m\times k}\mathbf{N}_{k\times n}\end{aligned} \tag{4}$$

Where $\mathbf{U}$ is an $m\times m$ unitary matrix, the matrix $\mathbf{S}$ is an $m\times n$ (we assume $m < n$) diagonal matrix with nonnegative numbers on the diagonal, and the $n\times n$ unitary matrix $\mathbf{V}^T$ denotes the conjugate transpose of the $n \times n$ unitary matrix $\mathbf{V}$. The diagonal entries s_i of $\mathbf{S}$ are known as the singular values of $\mathbf{A}$. A common convention is to list the singular values in descending order. In this case, the diagonal matrix $\mathbf{S}$ is uniquely determined by $\mathbf{A}$. We can save k singular values and approximate $\mathbf{A}_{m\times n}$ with $\mathbf{U}_{m\times k}\mathbf{N}_{k\times n}$.

Previous researches in DNN models with matrix decomposition mainly focus on reducing the number of parameters of the neural networks, such as [18, 19, 20]. They decompose the weights of DNN models with Low-Rank factorization or SVD to conspicuously reduce the free number of parameters of the neural networks and then restructure the neural networks without a significant loss in final recognition accuracy.

3.2. Initialization Process

In this paper, we propose a method using SVD for the initializations of speaker codes and connection weights to obtain a compared ASR performance as before but using a smaller speaker code size and less computation complexity. Speaker codes scheme can be viewed as a constrained modification of layer biases for each different speaker, it reduces the size of free parameters for adapting to each new speaker and alleviates the over-fitting problem by using a small training data set for estimating a set of speaker codes. The performance is highly dependent on the initializations of speaker codes and connection weights since we have to tune them at the same time. Obtaining good initializations quickly and steadily is very important in this adaptation scheme. We assume that a good speaker-dependent DNN models can be described as:

$$\mathbf{O}^{(l)} = \sigma(\mathbf{W}^{(l)}\mathbf{O}^{(l-1)} + \alpha\mathbf{C}^{(l)(c)}) \quad (\forall\ l) \tag{5}$$

It consists of two parts, speaker-independent $\mathbf{W}^{(l)}\mathbf{O}^{(l-1)}$ and speaker-dependent bias $\mathbf{C}^{(l)(c)}$. And α is an adjustable weight parameter. We can train a robust speaker-independent part and a set of $\mathbf{C}^{(l)(c)}(\forall\ l)$ for speaker c in the training set through standard BP algorithm if we have a large number of labeled data. It means that we can get good $\mathbf{C}^{(l)(c)}$ for all training speakers during training process and then decompose it using eq.(4) as $\mathbf{B}^{(l)}\mathbf{S}^{(c)}$. In this case, $\mathbf{B}^{(l)}$ is an $m \times k$ matrix and $\mathbf{S}^{(c)}$ is an $k \times n$ matrix, where m is the l-th layer size, n is the number of total speakers and k can be adjusted as the speaker code size, k should be chose carefully since it is a tradeoff between the number of free parameters and maintaining model structure. Since those matrices are generated from well-trained $\mathbf{C}^{(l)(c)}(\forall\ l)$ that are estimated with a large number of labeled data, we believe they can model some speaker-independent structure through $\mathbf{B}^{(l)}$ and some speaker-dependent information with $\mathbf{S}^{(c)}$. We then use them like in eqs.(1), either directly as the connection weights and speaker codes or just as the initializations of them. After obtaining training speaker codes and connection weights, adaptation and test processes are the same as described in section 2. Our previous work [1, 2, 3, 4] mainly focus on supervised adaptation. As opposed to them, we have evaluated the proposed adaptation scheme under an unsupervised speaker adaptation setting for large vocabulary speech recognition in this paper.

4. EXPERIMENTS

In this section, we evaluate the proposed method for speaker adaptation in the large-scale 320-hr Switchboard task.

The SWB training data consists of 309 hour Switchboard-I training set and 20 hour Call Home English training set (1540 speakers in total). In this work, we use the NIST 2000 Hub5e set (containing 1831 utterances from 40 speakers) as the evaluation set. We use 39 dimensional PLP features to train a baseline realigned DNN (retrained by realigned state labels based on the cross-entropy criterion) as described in [21, 22, 23, 24] with RBM-based pretraining and BP-based fine-tuning. Two baseline DNNs with various sizes are built: i) 3 hidden layers with 1024 nodes in each hidden layer; ii) 6 hidden layers with 2048 nodes in each hidden layer. For training all parameters, we use the same strategies as described in [3, 4]. In the evaluation set (Hub5e00), each test speaker has different number of utterances. The test is conducted for each speaker with unsupervised adaptation method which means that all test utterances are used for estimating speaker codes then obtain the final results through re-decoding.

In this section, we first evaluate the performance of using only speaker-independent part in the trained models. As shown in Table 1, the adding of additional speaker-dependent bias can help to acquire better speaker-independent DNNs. For example, for 6-layer DNN, it can reduce word error rate from 15.9% down to 15.4% (about 3.1% relative error reduction) when using only speaker-independent part. In next experiments we use the best models as the baselines.

Table 1. *WER (in%) of using only speaker-independent part in DNNs.*

DNNs	baseline	value of α			
		0.25	0.5	0.75	1.0
3*1024	18.7	18.2	**17.7**	18.1	18.2
6*2048	15.9	15.5	15.5	**15.4**	15.4

Next, we consider to use the composed matrices directly as the connection weights without training of them, Different speaker code size is conducted for testing. We also investigate experiments using random initialisation for the connection weights to demonstrate the effect of SVD (see 6*2048(r)). The results in Table 2 show that the composed connection weights may bring obvious improvements. For example, for 6-layer CE DNNs, it reduces WER from 15.4% down to 14.8% (3.9% relative error reduction).

Table 2. *WER (in%) of using static composed matrices as speaker codes and connection weights.*

DNNs	baseline	speaker code size				
		200	500	800	1000	1500
3*1024	17.7	17.5	17.5	17.6	**17.4**	17.6
6*2048	15.4	15.0	**14.8**	14.9	14.9	15.0
6*2048(r)	15.4	15.3	15.3	**15.2**	15.3	15.4

Then we consider to use the composed matrices as the initializations of speaker codes and connection weights and tune them in mSA-SC scheme. As shown in Table 3, we find that the training process can only provide small gain compared with using them directly (for 6-layer CE DNNs and 500 code size, 1.4% relative error reduction).

Table 3. *WER (in%) of using composed matrices as the initializations of speaker codes and connection weights in mSA-SC scheme.*

DNNs	baseline	speaker code size				
		200	500	800	1000	1500
3*1024	17.7	17.1	**16.9**	17.0	17.0	17.0
6*2048	15.4	14.7	**14.6**	14.7	14.7	14.8

At last, we consider to use the composed matrices as the initializations of speaker codes and connection weights and tune them in SAT-SC scheme. The results in Table 4 show the best result we can obtain with SAT-SC scheme. We can also achieve compared performance through other two methods in [4]. First is SAT-SC with mSA-SC as initializations (the best performance 12.5%,speaker code size 1000), second is SAT-SC using I-Vector as speaker code (the best performance 12.4%,speaker code size 400). Compared with those methods, the SVD initialization provide a smaller speaker code size and less computation complexity. At the same time, it has no need to estimate the I-Vector for each different speaker.

Table 4. *WER (in%) of using composed matrices as the initializations of speaker codes and connection weights in SAT-SC scheme.*

DNNs	criterion	speaker code size				
		200	500	800	1000	1500
3*1024	CE	**15.9**	16.0	16.0	15.9	16.0
6*2048	CE	14.0	13.7	13.8	13.9	14.1
	MMI	12.7	**12.5**	12.6	12.8	12.9

In summary, the proposed method using SVD to initialize speaker codes and connection weights can obtain a compared best ASR performance as before while using a smaller speaker code size and less computation complexity. this make the speaker code based adaptation method more effective for large and deep neural networks.

5. CONCLUSION

In this paper, we propose a method using SVD to initialize speaker codes and connection weights to obtain a comparable ASR performance as before but with a smaller speaker code size and much less computation complexity. Meanwhile, we have evaluated unsupervised speaker adaptation with the proposed method in large vocabulary speech recognition in the Switchboard task. Experimental results have shown that it is effective for providing well initializations and suitable in adapting large DNN models.

6. REFERENCES

[1] Ossama Abdel-Hamid and Hui Jiang, "Fast speaker adaptation of hybrid NN/HMM model for speech recognition based on discriminative learning of speaker code," in *IEEE International Conference of Acoustics,Speech and Signal Processing (ICASSP)*. IEEE, 2013, pp. 7942–7946.

[2] Ossama Abdel-Hamid and Hui Jiang, "Rapid and effective speaker adaptation of convolutional neural network based models for speech recognition," in *INTERSPEECH*, 2013.

[3] Shaofei Xue, Ossama Abdel-Hamid, Hui Jiang, and Lirong Dai, "Direct adaptation of hybrid DNN/HMM model for fast speaker adaptation in LVCSR based on speaker code," in *IEEE International Conference of Acoustics,Speech and Signal Processing (ICASSP)*, 2014.

[4] Shaofei Xue, Ossama Abdel-Hamid, Hui Jiang, Lirong Dai, and Qingfeng Liu, "Fast adaptation of deep neural network based on discriminant codes for speech recognition," *IEEE/ACM Transactions on Acoustics, Speech and Signal Processing*, vol. 22, no. 11, November 2014.

[5] Shaofei Xue, Hui Jiang, and Lirong Dai, "Speaker adaptation of hybrid NN/HMM model for speech recognition based on singular value decomposition," in *9th International Symposium on Chinese Spoken Language Processing (ISCSLP)*, 2014.

[6] J. L. Gauvain and Chin-Hui Lee, "Maximum a posteriori estimation for multivariate Gaussian mixture observations of Markov chains," *IEEE Transactions on Speech and audio processing*, vol. 2, no. 2, pp. 291–298, 1994.

[7] S. M. Ahadi and P. C. Woodland, "Combined Bayesian and predictive techniques for rapid speaker adaptation of continuous density hidden Markov models," *Computer speech & language*, vol. 11, no. 3, pp. 187–206, 1997.

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

[9] Mark J. F. Gales, "Maximum likelihood linear transformations for HMM-based speech recognition," *Computer speech & language*, vol. 12, no. 2, pp. 75–98, 1998.

[10] Vassilios V Digalakis, Dimitry Rtischev, and Leonardo G Neumeyer, "Speaker adaptation using constrained estimation of Gaussian mixtures," *IEEE Transactions on Speech and Audio Processing*, vol. 3, no. 5, pp. 357–366, 1995.

[11] Joao Neto, Lus Almeida, Mike Hochberg, Ciro Martins, Lus Nunes, Steve Renals, and Tony Robinson, "Speaker-adaptation for hybrid HMM-ANN continuous speech recognition system," in *EUROSPEECH*, 1995.

[12] Roberto Gemello, Franco Mana, Stefano Scanzio, Pietro Laface, and Renato De Mori, "Linear hidden transformations for adaptation of hybrid ANN/HMM models," *Speech Communication*, vol. 49, no. 10, pp. 827–835, 2007.

[13] Sabato Marco Siniscalchi, Jinyu Li, and C-H Lee, "Hermitian polynomial for speaker adaptation of connectionist speech recognition systems," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 10, pp. 2152–2161, 2013.

[14] Frank Seide, Gang Li, Xie Chen, and Dong Yu, "Feature engineering in context-dependent deep neural networks for conversational speech transcription," in *2011 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, 2011.

[15] Kaisheng Yao, Dong Yu, Frank Seide, Hang Su, Li Deng, and Yifan Gong, "Adaptation of context-dependent deep neural networks for automatic speech recognition," in *Spoken Language Technology Workshop (SLT)*. IEEE, 2012, pp. 366–369.

[16] Dong Yu, Kaisheng Yao, Hang Su, Gang Li, and Frank Seide, "KL-divergence regularized deep neural network adaptation for improved large vocabulary speech recognition," in *IEEE International Conference of Acoustics,Speech and Signal Processing (ICASSP)*. IEEE, 2013, pp. 7893–7897.

[17] Saon G, Soltau H, Nahamoo D, and Picheny M, "Speaker adaptation of neural network acoustic models using I-vectors," in *ASRU*, 2013.

[18] Misha Denil, Babak Shakibi, Laurent Dinh, Nando de Freitas, et al., "Predicting parameters in deep learning," in *Advances in Neural Information Processing Systems*, 2013, pp. 2148–2156.

[19] Tara N Sainath, Brian Kingsbury, Vikas Sindhwani, Ebru Arisoy, and Bhuvana Ramabhadran, "Low-rank matrix factorization for deep neural network training with high-dimensional output targets," in *IEEE International Conference of Acoustics,Speech and Signal Processing (ICASSP)*. IEEE, 2013, pp. 6655–6659.

[20] Jian Xue, Jinyu Li, and Yifan Gong, "Restructuring of deep neural network acoustic models with singular value decomposition," in *INTERSPEECH*, 2013.

[21] Jia Pan, Cong Liu, Zhiguo Wang, Yu Hu, and Hui Jiang, "Investigation of deep neural networks (DNN) for large vocabulary continuous speech recognition: Why DNN surpasses GMMs in acoustic modeling," in *8th International Symposium on Chinese Spoken Language Processing (ISCSLP)*, 2012, pp. 301–305.

[22] Yebo Bao, Hui Jiang, Cong Liu, Yu Hu, and Lirong Dai, "Investigation on dimensionality reduction of concatenated features with deep neural network for LVCSR systems," in *IEEE 11th International Conference on Signal Processing (ICSP)*, 2012, vol. 1, pp. 562–566.

[23] Yebo Bao, Hui Jiang, Lirong Dai, and Cong Liu, "Incoherent training of deep neural networks to de-correlate bottleneck features for speech recognition," in *IEEE International Conference of Acoustics,Speech and Signal Processing (ICASSP)*, 2013.

[24] Shiliang Zhang, Yebo Bao, Pan Zhou, Hui Jiang, and Lirong Dai, "Improving deep neural networks for LVCSR using dropout and shrinking structure," in *IEEE International Conference of Acoustics,Speech and Signal Processing (ICASSP)*, 2014.