INVESTIGATING ONLINE LOW-FOOTPRINT SPEAKER ADAPTATION USING GENERALIZED LINEAR REGRESSION AND CLICK-THROUGH DATA

Yong Zhao, Jinyu Li, Jian Xue, and Yifan Gong

Microsoft Corporation, One Microsoft Way, Redmond, WA 98052, USA

{yonzhao; jinyli; jianxue; ygong}@microsoft.com

ABSTRACT

To develop speaker adaptation algorithms for deep neural network (DNN) that are suitable for large-scale online deployment, it is desirable that the adaptation model be represented in a compact form and learned in an unsupervised fashion. In this paper, we propose a novel low-footprint adaptation technique for DNN that adapts the DNN model through node activation functions. The approach introduces slope and bias parameters in the sigmoid activation functions for each speaker, allowing the adaptation model to be stored in a small-sized storage space. We show that this adaptation technique can be formulated in a linear regression fashion, analogous to other speak adaptation algorithms that apply additional linear transformations to the DNN layers. We further investigate semi-supervised online adaptation by making use of the user click-through data as a supervision signal. The proposed method is evaluated on short message dictation and voice search tasks in supervised, unsupervised, and semi-supervised setups. Compared with the singular value decomposition (SVD) bottleneck adaptation, the proposed adaptation method achieves comparable accuracy improvements with much smaller footprint.

Index Terms: automatic speech recognition, deep neural network, speaker adaptation, low footprint

1. INTRODUCTION

Recent progress in deep learning has attracted a lot of interest in automatic speech recognition (ASR) [1], [2], [3], [4]. The discovery of the strong modeling capabilities of deep neural networks (DNNs) and the availability of high-speed hardware has made it feasible to train large networks with tens of millions of parameters. In the framework of context-dependent DNN hidden-Markov-models (CD-DNN-HMMs) [1], the conventional Gaussian Mixture Model (GMM) is replaced by a DNN to evaluate the senone log-likelihood.

However, the outstanding performance of CD-DNN-HMMs requires huge number of parameters, which makes adaptation very challenging, especially with limited adaptation data. Several methods for DNN adaptation have previously been proposed. The most popular approach for adapting DNNs is applying a linear transformation to a certain DNN layer to account for the mismatch between the training and test conditions. In [5], [6], [7], [8], [9], an additional layer is defined between the input observations and the first hidden layer, similar to the conventional feature space maximum likelihood linear regression (fMLLR) [10] in CD-GMM-HMMs. The linear transformation has been further applied to the hidden layers [11], and to the top layer [8], [12]. One main issue in these adaptation techniques is that they typically need to update and store a large amount of adaptation parameters due to the high dimensionality of the DNN layers. Feature discriminative linear regression (fDLR) [9] introduces a small-sized adaptation model by sharing each of the input frames with the same transform. In [13], a factorized adaptation method is proposed to adapt a DNN with only limited number of parameters by taking into account the underlying factors that contribute to the distorted speech signal. Nevertheless, all these techniques define the transforms on one or a few DNN layers, and the potential of deeply adapting the DNN model across many layers has not yet been fully explored. In [14], we have proposed the SVD bottleneck adaptation by adapting all the linear bottleneck layers in the SVD-restructured model. This technique involves much less adaptation parameters, while providing significant accuracy improvement.

The aforementioned adaptation methods either adapt or add matrices to characterize the target speaker or environment. There have been few efforts in the literature to adjust the node activation functions. In [15], [16], a very complicated Hermitian polynomial function is used as the hidden node activation function of a shallow neural network. However, there is no conclusion whether the discovery in [16] with the shallow Hermitian polynomial neural network can be applied to DNN adaptation. In [17], the DNN model is adapted by augmenting the activations with the amplitude parameter.

In this study, we propose to adapt the DNN model by adjusting the node activation functions. The proposed approach introduces slope and bias parameters in the activation functions for each speaker. One advantage of adapting the node activation function is that the number of adaptation parameters is much smaller than those used for matrix adaptation. Therefore, it is very suitable for low-footprint adaptation or personalization. We show that this adaptation method can also be formulated in a linear regression fashion. The unified view facilitates the implementation of a general framework for adapting the DNN model.

The accuracy of the hypothesized transcripts in the adaptation set plays an important role in training the speaker-dependent (SD) model. It is desirable to adapt models in an unsupervised or semi-supervised manner given the difficulties of obtaining the correct transcripts for every speaker. In this work, we leverage the abundance of implicitly labeled voice search queries that are logged in search engines. We investigate semi-supervised adaptation by making use of the user click-through data as a supervision signal.

The rest of this paper is organized as follows. We will first briefly introduce the DNN adaptation from linear regression perspective in Section 2. In Section 3, we propose the DNN adaptation method by adjusting node activation functions. Section 4 describes the strategies to train the adaptation model. Then, we evaluate our proposed method and compare it with the existing adaptation methods in Section 5, and conclude the study in Section 6.

2. DNN ADAPTATION USING LINEAR REGRESSION

A deep neural network (DNN) [1] can be considered as a conventional multi-layer perceptron (MLP) with many hidden layers, where the input feature is concatenated from multiple consecutive frames and the output predicts the posterior probabilities of thousands of senones. Given a DNN with L hidden layers, the output at the l-th hidden layer, $\boldsymbol{h}^l$, is recursively defined as the nonlinear transformation of the $(l-1)$-th layer:

$$\boldsymbol{h}^l = \sigma(\boldsymbol{v}^l) = \sigma(\boldsymbol{W}^l \boldsymbol{h}^{l-1} + \boldsymbol{b}^l) \quad (1)$$

where $\boldsymbol{W}^l$ is the weight matrix, $\boldsymbol{b}^l$ is the bias vector, and $\sigma(\cdot)$ is the sigmoid activation function defined element-wise

$$\sigma(v) = 1/(1+e^{-v}) \tag{2}$$

Note $\boldsymbol{h}^l$ and $\boldsymbol{v}^l$ correspond to the activations and excitations of the l-th layer, respectively. $\boldsymbol{h}^0 = \boldsymbol{x}$ is the input observation vector. For CD-DNN-HMMs [1], the output layer is normalized by the softmax function to produce the posterior probability of senone id s, $p(s|\boldsymbol{x})$.

Many DNN adaptation techniques have been developed in the past. As discussed in the introduction section, the most popular approach for adapting DNNs is applying a linear transformation to the certain DNN layer to account for the mismatch between the training and test conditions [5], [6], [7], [8], [9], [11]. The basic idea of this model is illustrated in Fig. 1a. Note that the parameters corresponding to the red dashed links are trained using the adaptation set, keeping other weights of the original DNN fixed. One main issue in these adaptation techniques is that they typically need to update and store a large amount of adaptation parameters due to the high dimensionality of the DNN layers.

2.1. SVD bottleneck adaptation

We recently presented a SVD-based method in [18] to restructure the DNN model in a significantly small size while maintaining the recognition accuracy. Given an $m \times n$ weight matrix $\boldsymbol{W}$ in a DNN, we approximate it as the product of two low-rank matrices by applying SVD

$$\boldsymbol{W}_{m\times n} \approx \boldsymbol{U}_{m\times k}\boldsymbol{N}_{k\times n} \tag{3}$$

If W is a low-rank matrix, k will be much smaller than m and n, and the number of parameters is reduced from mn to $(m+n)k$. Applying this decomposition to the weight matrix, it acts as if inserting a linear bottleneck layer of fewer units between the original nonlinear layers. Thus, the original large full-rank DNN model is converted to a much smaller low-rank model without loss of accuracy.

Furthermore, we propose the SVD bottleneck adaptation in [14] to produce low-footprint SD models by making use of the SVD-restructured topology. The linear transformation is applied to each of the bottleneck layer by adding an additional layer of k units, as illustrated in Fig. 1b. We have

$$\boldsymbol{W}_{s,m\times n} = \boldsymbol{U}_{m\times k}S_{s,k\times k}\boldsymbol{N}_{k\times n} \tag{4}$$

where $S_{s,k\times k}$ is the transformation matrix for speaker s and is initialized to be an identity matrix $I_{k\times k}$. The advantage of this approach is that only a couple of small matrices need to be updated for each speaker. This dramatically reduces the deployment cost for speaker personalization.

3. DNN ADAPTATION THROUGH ACTIVATION FUNCTION

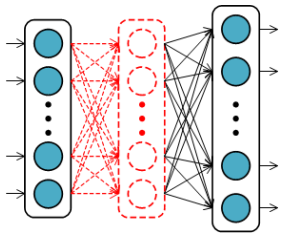
The aforementioned adaptation methods either adapt or add transformation matrices to characterize the target speaker. In this section, we propose to adapt the DNN model by adjusting the node activation functions. We modify the sigmoid function (2) in a general form

$$\tilde{\sigma}(v) = 1/(1+e^{-(\alpha v+\beta)}) \tag{5}$$

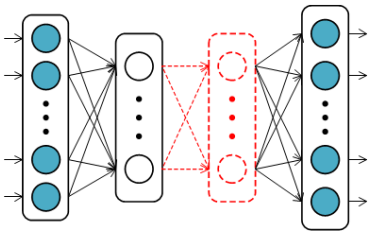
where α is slope and β is bias. The slopes and biases are initialized to 1 and 0, respectively, and updated for each speaker. The main advantage of adapting through activation functions is that the total number of adaptation parameters is much small, two times of the total number of hidden units.

Substituting (5) into (1), we have

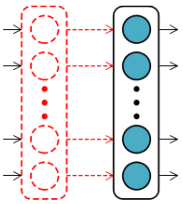
$$\boldsymbol{h}_s^l = \tilde{\sigma}_s\left(\boldsymbol{v}^l\right) = \sigma\left(\boldsymbol{A}_s^l\boldsymbol{v}^l + \boldsymbol{b}_s^l\right) \tag{6}$$



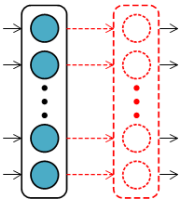
(a) Adaptation of one DNN layer



(b) SVD bottleneck adaptation



(c) Adapting slopes and biases in activation functions



(d) Adaptation of amplitudes in activation functions

Fig. 1: Illustration of network structures of different adaptation methods. Shaded nodes denote nonlinear units, unshaded nodes for linear units. Red dashed links indicate the transformations that are introduced during adaptation.

where $\boldsymbol{A}_s$ is the diagonal matrix with activation slopes α on the diagonal, and $\boldsymbol{b}_s$ is the activation bias vector. We can see that adapting the slopes and biases through the activation functions amounts to adding a linear layer right before the activation functions with the one-to-one correspondence, as shown in Fig. 1c.

3.1. Generalized linear regression

We have shown that many adaptation techniques introduced above belong to the family of the linear regression. Motivated by the widely used MLLR [19] and fMLLR [10] in the conventional CD-GMM-HMM, linear transformation matrices are inserted between the DNN layers to account for the mismatch between the training and test conditions. Various such adaptation schemes are illustrated in Fig. 1. Moreover, the DNN model can be adapted by augmenting the activations with the amplitude parameter, similar to the one described in [17]. This is equivalent to adding a linear layer after the activation functions are invoked, as shown in Fig. 1d. Note in [17], the amplitudes are transformed with the sigmoid functions to con-

strain the resulting activations to be nonnegative. The unified view from the generalized linear regression (GLR) perspective facilitates the implementation of a general framework for adapting the DNN model, and these adaptation techniques can be readily combined for potentially improved performance.

4. TRAINING ADAPTATION MODELS

In this section, we present several strategies for training the DNN adaptation model, so that the resulting model is enhanced in terms of stability against the limited amount of adaptation data and accuracy when manually labeled data are not available.

The normal error backpropagation (BP) algorithm [20] can be directly used to train the SD models that employ the GLR. The model parameters are estimated by maximizing the negative cross entropy

$$D = \frac{1}{N}\sum_{t=1}^{N}\sum_{s_t=1}^{S}\tilde{p}(s_t|\boldsymbol{x}_t)\log p(s_t|\boldsymbol{x}_t) \tag{7}$$

where S is the total number of senones, and N is the number of samples in the training set. The BP algorithm updates the parameters by propagating the error signal backwards from the top layer to bottom as follows:

$$\boldsymbol{e}_t^l = \frac{\partial D}{\partial \boldsymbol{v}_t^l} = \left(\boldsymbol{W}^{l+1}\right)^{\mathrm{T}}\boldsymbol{e}_t^{l+1} \circ (\boldsymbol{h}_t^l)' \tag{8}$$

where the operator $\circ$ denotes an element-wise product. When the l-th layer is nonlinear with the sigmoid function, we have $(\boldsymbol{h}^l)' = \sigma'(\boldsymbol{v}^l) = \sigma(\boldsymbol{v}^l) \circ \sigma(1-\boldsymbol{v}^l)$. When the layer is linear, such as the one inserted for the DNN adaptation, $(\boldsymbol{h}^l)' = 1$. Given the adaptation data, we typically train the linear transforms starting from an identity matrix and zero bias, keeping the weights of the original DNN fixed.

4.1. KLD regularized adaptation

A straightforward approach to adapt a DNN is to estimate the SD parameter with the adaptation data using the regular cross entropy criterion in (7). However, doing so may over-fit the model to the adaptation data, especially when the adaptation set is small and the supervision hypotheses are erroneous. A regularized adaptation method was proposed to address this issue [21]. The idea is that the posterior senone distribution estimated from the adapted model should not deviate too far from the one estimated with the SI model. By adding the KullbackLeibler divergence (KLD) as a regularization term to (7), we get a regularized optimization criterion, which has the same form as (7) except that the target probability distribution $\tilde{p}(s_t|\boldsymbol{x}_t)$ is substituted by

$$\hat{p}(s_t|\boldsymbol{x}_t) = (1-\rho)\tilde{p}(s_t|\boldsymbol{x}_t) + \rho p^{\mathrm{SI}}(s_t|\boldsymbol{x}_t) \tag{9}$$

where ρ is the regularization weight, and $p^{\mathrm{SI}}(s_t|\boldsymbol{x}_t)$ is the posterior probability estimated from the SI model. It can be seen that $\hat{p}(s_t|\boldsymbol{x}_t)$ is a linear interpolation of the distribution estimated from the SI model and the ground truth alignment of the adaptation data. This interpolation constraints the adapted model not to deviate far away from the SI model, when the adaptation data are limited.

4.2. Supervision from click-through data

The accuracy of the hypothesized transcripts in the adaptation set plays an important role in training the SD model. Manually transcribing the adaptation data for each speaker is infeasible for the large-scale system deployment. It is desirable to adapt the models in unsupervised and semi-supervised manners. One popular approach is to generate better quality hypotheses using various offline decoding techniques, such as the use of more powerful acoustic models and language models, and multiple system combination [22], [23], [24], [25]. However, such an approach would be very time-consuming and expensive when the system serves a huge amount of users. An alternative approach is to reuse the online recognition results and select the utterances that are plausibly accurate [26]. Simple selection based on confidence measure may produce utterances with high accurate hypotheses, but is not optimal, as it just reinforces well-known and less informative patterns to the system, and limits the diversity of the data set. It has been observed that a good strategy is to discard utterances with either a very low or a very high confidence [25].

In this work, we leverage the abundance of implicitly labeled voice search queries that are logged in search engines. The large-scale search engines such as Bing can be accessed through voice interface. The user click for a voice query is a significant indicator for the satisfaction of the voice search service, in which the recognition accuracy plays an important part. As a preliminary study, we simply select the user click-through data for adapting the DNN models.

5. EXPERIMENTS AND RESULTS

The proposed adaptation method was evaluated on two tasks, short message dictation (SMD) and voice search (VS) and compared with the SVD bottleneck adaptation. Alternatively, there exist several fast adaptation schemes in the speech community such as i-vector based DNN adaptation. We do no include the i-vector based adaptation as a point of comparison, because many VS utterances are very short and the i-vector estimation in an utterance level is not reliable [27].

The baseline SI models were trained with 300 hours VS and SMD data. The input feature to CD-DNN-HMM system is a 24-dimension mean-normalized log-filter bank feature with up to second-order derivatives and a context window of 11 frames, forming a vector of 792-dimension (72×11) input. On top of the input layer there are 5 hidden layers with 2,048 units for each. The output layer has a dimension of 5,976. We convert the full-rank DNN model to low-rank model by doing SVD on all the matrices except the one between the input and the first hidden layer, and keep 40% of total singular values. The numbers of units on the linear layers after SVD are 208, 184, 176, 200, and 344 accordingly, from bottom to top. We then retrained the low-rank model and obtained comparable accuracy to the full-rank model. More details of SVD based low-rank DNN model training can be found in [18].

Table 1: Comparison of the number of parameters for different adaptation methods.

Acoustic model	# of parameters
Full-rank SI model	30M
Low-rank SI model	7.4M
SVD adaptation	266K
Sigmoid adaptation	20K

Table 1 compares the number of parameters for different methods. The full-rank SI DNN has 30M parameters and the low-rank SI DNN has 7.4M parameters. The SVD adaptation produces 266K SD parameters for the introduced regression matrix on SVD linear layers. In contrast, The sigmoid adaptation methods adapts the slopes and biases of activation functions on the hidden layers, requiring 20K parameters, which is only 7.5% of parameters for the SVD adaptation.

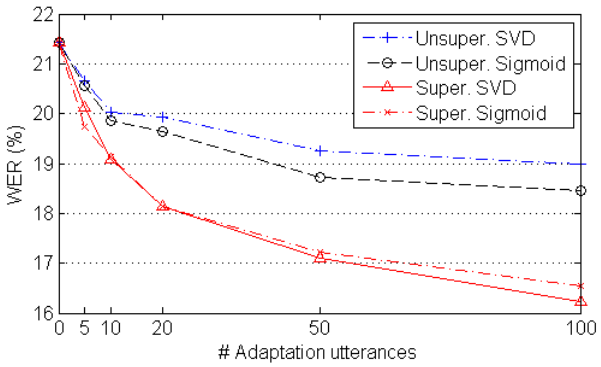


Fig. 2: WER (%) against the number of adaptation utterances for different adaptation models on the SMD task.

5.1. Short message dictation

The initial experiments were conducted on a SMD task which consists of 7 speakers. The total number of test set words is 20,203. There is no overlap among the development and test data. The baseline low-rank SI system achieves 21.43% WER averaged on 7 speakers. The DNN models are adapted in both supervised and unsupervised way, where the SI model is used to decode and align the development data. We varied the number of adaptation utterances from 5 (32 seconds) to 200 (22 minutes) for each speaker.

The regularization weight ρ has an observable impact on the adaptation performance [21], but its effect is not of interest in this paper. For the SVD adaptation, ρ is empirically set to 0.1 and 0.5 in supervised and unsupervised setups, respectively; for the sigmoid adaptation, ρ is set to 0 and 0.2, respectively.

Fig. 2 summarizes the WER on the SMD task for the two adaptation methods in supervised and unsupervised setups. From this figure we can make three observations. First, all of the adaptation models produce continuous improvement in WER along the increase of the available adaptation data. Second, in the unsupervised setup, the sigmoid adaptation produces 4.1-13.9% relative WER reduction (WERR) with 5-100 adaptation utterances, which outperforms the corresponding SVD adaptation (3.6-11.4% WERR). Third, in the supervised setup, the sigmoid adaptation yields the similar performance to the SVD adaptation with 10-50 adaptation utterances (10.7-19.7% WERR), and performs slightly better than the SVD adaptation with 5 adaptation utterances (7.9% vs. 6.1% WERR). These observations indicate that the sigmoid adaptation are more robust than the SVD adaptation when the adaptation set is very limited or the supervision hypotheses are erroneous.

5.2. Voice search using click-through data

The second experiment was conducted on a VS task to investigate the performance of the adaptation techniques using the click-through data as supervision signal. In this section, we report WERR as a reference of system performance, as the click-through data are collected from the deployed speech recognition service. To apprehend the acoustic characteristics of the click-through data, we first profiled a set of VS queries collected during a certain period of service, as shown in Table 2. It is observed that around 1/3 of VS queries are followed by the click actions. The user's clicking to a voice query acts as a significant indicator for the recognition correctness, as the click-through data remarkably decrease the WER by 60.5% relative. Moreover, the click-though data feature a higher confidence score and a higher number of words per utterance than the ordinary data in average.

Table 2: Profile of the user click-through data.

	Utts. (%)	WERR (%)	Conf. score	# words per utt.	Speech length (s)
All	—	—	0.724	3.99	1.49
Clicked	34.16	60.5	0.789	4.23	1.54

The evaluation was conducted on data from 30 speakers. Each speaker uses 100 utterances as adaptation data. In the semi-supervised setup, the utterances associated with the clicked queries are selected for adaptation, and the online recognition results are used as supervision hypotheses. In the unsupervised setup, the develop data of 100 utterances are randomly chosen. There is no overlap among the development and test data.

Table 3 compares the WERR for different adaptation methods in unsupervised and semi-supervised setups, respectively. We can see that the use of click-through data contributes significant gains to the recognition performance. In particular, adaptation using the click-through data provides 11.1% and 10.2% WERR for the SVD bottleneck adaptation and the sigmoid adaptation, respectively, compared with 3.6% and 4.1% WERR for the standard unsupervised adaptation. Moreover, the sigmoid adaptation obtains similar WER to the SVD bottleneck adaptation in both unsupervised and semi-supervised setups.

Table 3: Comparison of WERR (%) for different adaptation methods on the VS task.

Adaptation model	Unsuper.	Semi-super.
SVD adaptation	3.6	11.1
Sigmoid adaptation	4.1	10.2

6. CONCLUSION

In this paper we presented a low-footprint DNN adaptation technique that adapts the DNN model through node activation functions. This technique requires only a small amount of parameters for each speaker, two times of the total number of hidden units. We demonstrated that this adaptation technique falls in the category of generalized linear regression. Experiments demonstrated that the sigmoid adaptation achieves the remarkable accuracy improvements in various adaptation setups, which is comparable with the SVD adaptation. Moreover, the sigmoid adaptation is more robust than the SVD adaptation when the adaptation set is very limited or the supervision hypotheses are erroneous, which is attributed to its compact representation of speaker characteristics. Meanwhile, the sigmoid adaptation requires only 7.5% of parameters for the SVD adaptation. The small size of the SD model makes it appealing in deploying large-scale speech recognition service for possible millions of users.

Our preliminary investigation showed that the semi-supervised adaptation using click-through data outperformed the conventional unsupervised adaptation. In the future, we plan to explore more complicated methods to process the click-through data for the purpose of the DNN training and adaptation. The click-through data are still noisy. Often, a recognized query that partially matches the speech input triggers user clicks, because it retrieves the search results relevant to the user's intent. Sometimes, they are just random clicks. It is desirable to incorporate a confidence classifier to refine the click-through data. Moreover, selecting the adaptation data at a segment or frame level would be beneficial.

7. REFERENCES

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

[2] N. Jaitly, P. Nguyen, A. Senior, and V. Vanhoucke, "Application of pretrained deep neural networks to large vocabulary speech recognition," in *Proc. Interspeech*, 2012, pp. 2578–2581.

[3] L. Deng, J. Li, J.-T. Huang, et al., "Recent advances in deep learning for speech research at Microsoft," in *Proc. ICASSP*, 2013, pp. 8604–8608.

[4] T. N. Sainath, B. Kingsbury, B. Ramabhadran, P. Fousek, P. Novak, and A. Mohamed, "Making deep belief networks effective for large vocabulary continuous speech recognition," in *Proc. ASRU*, 2011, pp. 30–35.

[5] V. Abrash, H. Franco, A. Sankar, and M. Cohen, "Connectionist speaker normalization and adaptation," in *Proc. Eurospeech*, 1995, pp. 2183–2186.

[6] J. Neto, L. Almeida, M. Hochberg, C. Martins, Lu. Nunes, S. Renals, and T. Robinson, "Speaker-adaptation for hybrid HMM-ANN continuous speech recognition system," in *Proc. Eurospeech*, 1995, pp. 2171–2174.

[7] B. Li and K. C. Sim, "Comparison of discriminative input and output transformations for speaker adaptation in the hybrid NN/HMM systems," in *Proc. Interspeech*, 2010, pp. 526–529.

[8] X. Liu, M. J. F. Gales, and P. C. Woodland, "Improving LVCSR system combination using neural network language model cross adaptation," in *Proc. Interspeech*, 2011, pp. 2857–2860.

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

[10] M. J. F. Gales, "Maximum likelihood linear transformations for HMM-based speech recognition," *Comput. Speech Lang.*, vol. 12, no. 2, pp. 75–98, Jan. 1998.

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

[12] K. Yao, D. Yu, F. Seide, H. Su, L. Deng, and Y. Gong, "Adaptation of context-dependent deep neural networks for automatic speech recognition.," in *Proc. SLT*, 2012, pp. 366–369.

[13] J. Li, J.-T. Huang, and Y. Gong, "Factorized adaptation for deep neural network," in *Proc. ICASSP*, 2014, pp. 5574–5578.

[14] J. Xue, J. Li, D. Yu, M. Seltzer, and Y. Gong, "Singular value decomposition based low-footprint speaker adaptation and personalization for deep neural network," in *Proc. ICASSP*, 2014, pp. 6409–6413.

[15] S. M. Siniscalchi, J. Li, and C.-H. Lee, "Hermitian based hidden activation functions for adaptation of hybrid HMM/ANN models," in *Proc. INTERSPEECH*, 2012, pp. 2590–2593.

[16] S. M. Siniscalchi, J. Li, and C.-H. Lee, "Hermitian polynomial for speaker adaptation of connectionist speech recognition systems," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 21, no. 10, pp. 2152–2161, 2013.

[17] P. Swietojanski and S. Renals, "Learning hidden unit contributions for unsupervised speaker adaptation of neural network acoustic models," in *Proc. SLT*, 2014, pp. 171–176.

[18] J. Xue, J. Li, and Y. Gong, "Restructuring of deep neural network acoustic models with singular value decomposition," in *Proc. Interspeech*, 2013, pp. 2365–2369.

[19] C. J. Leggetter and P. C. Woodland, "Maximum likelihood linear regression for speaker adaptation of continuous density hidden Markov models," *Comput. Speech Lang.*, vol. 9, pp. 171–186, 1995.

[20] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," in *Cognitive modeling*. MIT Press, Cambridge, MA, 1988.

[21] D. Yu, K. Yao, H. Su, G. Li, and F. Seide, "KL-divergence regularized deep neural network adaptation for improved large vocabulary speech recognition," in *Proc. ICASSP*, 2013, pp. 7893–7897.

[22] H. Y. Chan and P. C. Woodland, "Improving broadcast news transcription by lightly supervised discriminative training," in *Proc. ICASSP*, 2004, pp. 737–740.

[23] D. Yu, B. Varadarajan, L. Deng, and A. Acero, "Active learning and semi-supervised learning for speech recognition: A unified framework using the global entropy reduction maximization criterion," *Comput. Speech Lang.*, vol. 24, no. 3, pp. 433–444, 2010.

[24] K. Vesely, M. Hannemann, and L. Burget, "Semi-supervised training of deep neural networks," in *Proc. ASRU*, 2013, pp. 267–272.

[25] Y. Huang, D. Yu, Y. Gong, and C. Liu, "Semi-supervised GMM and DNN acoustic model training with multi-system combination and confidence re-calibration," in *Proc. Interspeech*, 2013, pp. 2360–2364.

[26] O. Siohan, "Training data selection based on context-dependent state matching," in *Proc. ICASSP*, 2014, pp. 3316–3319.

[27] A. Senior and I. Lopez-Moreno, "Improving DNN speaker independence with i-vector inputs," in *Proc. ICASSP*, 2014, pp. 225–229.