Interspeech 2018
2-6 September 2018, Hyderabad

Speaker Adaptation and Adaptive Training for Jointly Optimised Tandem Systems

Y. Wang, C. Zhang, M. J. F. Gales, P. C. Woodland

Cambridge University Engineering Dept, Trumpington St., Cambridge CB2 1PZ, U.K
{yw396, cz277, mjfg, pcw}@eng.cam.ac.uk

Abstract

Speaker independent (SI) Tandem systems trained by joint optimisation of bottleneck (BN) deep neural networks (DNNs) and Gaussian mixture models (GMMs) have been found to produce similar word error rates (WERs) to Hybrid DNN systems. A key advantage of using GMMs is that existing speaker adaptation methods, such as maximum likelihood linear regression (MLLR), can be used which to account for diverse speaker variations and improve system robustness. This paper investigates speaker adaptation and adaptive training (SAT) schemes for jointly optimised Tandem systems. Adaptation techniques investigated include constrained MLLR (CMLLR) transforms based on BN features for SAT as well as MLLR and parameterised sigmoid functions for unsupervised test-time adaptation. Experiments using English multi-genre broadcast (MGB3) data show that CMLLR SAT yields a 4% relative WER reduction over jointly trained Tandem and Hybrid SI systems, and further reductions in WER are obtained by system combination.

Index Terms: Speech recognition, Tandem system, joint training, speaker adaptive training

1. Introduction

In recent years, deep neural networks (DNNs) have become key components of speech recognition systems. DNNs can be used to estimate the posterior probabilities for context-dependent phone states, which are then converted into scaled likelihoods for use as hidden Markov model (HMM) observation probabilities. This configuration is referred to as DNN-HMM Hybrid system [1]. Another system configuration, referred to as a Tandem system configuration [2], uses the activations from a bottleneck (BN) hidden layer of a DNN as the input features for a GMM-HMM model [3]. In contrast to Hybrid systems whose parameters are all simultaneously trained, Tandem systems often have the GMMs estimated using a pre-trained BN DNN. This issue can be addressed by jointly training BN DNN and GMMs based on either the cross entropy (CE) [4, 5] or the minimum phone error (MPE) criteria [6]. These jointly trained speaker independent (SI) Tandem systems yield similar word error rates (WERs) to SI Hybrid systems.

Tandem system outputs are often complementary to those from Hybrid systems and hence are useful in system combination with Hybrid systems [7, 8, 6]. Furthermore Tandem systems can use the many GMM-HMM based speaker adaptation [9] and adaptive training (SAT) [10] approaches, such as Maximum A Posteriori (MAP) [11], Maximum Likelihood Linear Regression (MLLR) [12], and Cluster Adaptive Training [13]. By using a single linear transform to adapt both mean and variances of all models, adaptation can be applied at the feature level in constrained MLLR (CMLLR) [14]. When applied in SAT, CMLLR transforms and GMMs are iteratively updated to obtain better speaker-independent (SI) models, or canonical models [13].

In addition to the GMM based adaptation approaches, techniques that have been developed for DNN adaptation can also be used with Tandem systems. These include the use of additional fixed-length input that encodes speaker-specific information [15, 16, 17]. Another approach is to adapt some network parameters, such as the DNN weights, to speaker-dependent (SD) characteristics [18, 19], or adapt additional parameters associated with the activation functions [20, 21, 22, 23]. The SD parameters can be trained at either the frame or sequence level [24, 19, 6] with some regularisation [19, 25, 26].

This paper studies both SAT and unsupervised test-time speaker adaptation for jointly trained Tandem systems. The Tandem system joint training optimises the trained BN DNN and GMMs using the MPE criterion [6] and updates parameters using stochastic gradient descent (SGD). SAT operates during the joint training stage by interleaving updates of the SI parameters with the CMLLR transforms. At test-time, unsupervised adaptation is used to generate CMLLR transforms. In addition, parameterised sigmoid activation function (p-sigmoid) as well as model level MLLR transforms were also investigated. The experimental evaluation is based on transcription of British English multi-genre broadcast (MGB3) data.

The rest of the paper is organised as follows. Section 2 briefly reviews previous work and Section 3 describes the proposed methods. Sections 4 and 5 give the setup and results of speech recognition experiments on the MGB3 task. This is followed by conclusions.

2. Tandem Systems and SAT

2.1. Tandem system

A Tandem system uses a BN DNN to extract features for training GMM-HMM acoustic models. The BN DNN has a BN layer whose size is normally much smaller than other hidden layers, in order to generate compact output vectors that are suitable to be used as features in GMMs. In the l^{th} layer of the L-layer DNN, the activation $\mathbf{a}_l(t)$ at time t is given by

$$\mathbf{a}_l(t) = \mathbf{W}_l \mathbf{x}_l(t) + \mathbf{b}_l \tag{1}$$

where $\mathbf{x}_l(t)$ represents the input vector of the lth layer of the DNN and $\mathbf{W}_l$, $\mathbf{b}_l$ are the weight matrix and the bias vector respectively. An activation function, $f_l(\cdot)$, is then applied to transform the activation values to generate the output of the layer. The commonly used activation functions include sigmoid, $f_l(a_{li}(t)) = (1+\exp(-a_{li}(t)))^{-1}$, and ReLU, $f_l(a_{li}(t)) = \max(0, a_{li}(t))$, where $a_{li}(t)$ is the i^{th} element of $\mathbf{a}_l(t)$. Denoting the output values of the BN layer as $\mathbf{y}_{\text{bn}}(t)$,

This research was partly funded under the ALTA Institute, University of Cambridge. Thanks to Cambridge English, University of Cambridge, for supporting this research.

10.21437/Interspeech.2018-2432

the likelihood of a state j in the HMM model is given by

$$p\left(\mathbf{y}_{\mathsf{bn}}(t);\boldsymbol{\mu}^{(j)},\boldsymbol{\Sigma}^{(j)}\right)=\sum_{m}\phi^{(jm)}\mathcal{N}\left(\mathbf{y}_{\mathsf{bn}}(t);\boldsymbol{\mu}^{(jm)},\boldsymbol{\Sigma}^{(jm)}\right) \tag{2}$$

where $\phi^{(jm)}$, $\boldsymbol{\mu}^{(jm)}$ and $\boldsymbol{\Sigma}^{(jm)}$ represent the component weight, mean vector and covariance matrix for Gaussian m of state j. Thus, the likelihood of the GMM-HMM model in the Tandem system is calculated using $\mathbf{y}_{\mathsf{bn}}(t)$ instead of the standard acoustic features. Furthermore, $\mathbf{y}_{\mathsf{bn}}(t)$ is sometimes concatenated with acoustic features, $\mathbf{o}(t)$, to form an alternative type of input to the GMM-HMM model [2, 3]. Feature transforms, such as Heteroscedastic Linear Discriminant Analysis (HLDA) [28] and Semi-tied Covariance [29] transforms can also be used to refine the concatenated features.

2.2. Maximum Likelihood Linear Regression

MLLR uses a regression class tree to dynamically specify each group (or class) of HMM states, and creates a pair of linear transforms to adapt the means and variances of all states in that class by

$$\hat{\boldsymbol{\mu}}^{(sm)}=\mathbf{B}^{(sc)}\boldsymbol{\mu}^{(m)}+\mathbf{c}^{(s)};\;\;\hat{\boldsymbol{\Sigma}}^{(sm)}=\mathbf{H}^{(sc)}\boldsymbol{\Sigma}^{(m)}\mathbf{H}^{(sc)\mathsf{T}}, \tag{3}$$

where s is a speaker and c is the class relevant to Gaussian component m; $\mathbf{B}^{(sc)}$ and $\mathbf{c}^{(sc)}$ serves as the weight matrix and bias vector of the mean transform; $\mathbf{H}^{(sc)}$ is the covariance transform. CMLLR constrains both mean and variance to use the same transform. For CMLLR with only one class, the linear transform can be presented as

$$\hat{\boldsymbol{\mu}}^{(sm)}=\tilde{\mathbf{B}}^{(s)}\boldsymbol{\mu}^{(m)}+\tilde{\mathbf{c}}^{(s)};\;\;\hat{\boldsymbol{\Sigma}}^{(sm)}=\tilde{\mathbf{B}}^{(s)}\boldsymbol{\Sigma}^{(m)}\tilde{\mathbf{B}}^{(s)\mathsf{T}}, \tag{4}$$

which can be achieved equivalently by transforming the input features as

$$\begin{aligned}&\mathcal{N}\left(\mathbf{y}_{\mathsf{bn}}(t);\hat{\boldsymbol{\mu}}^{(sm)},\hat{\boldsymbol{\Sigma}}^{(sm)}\right)=\\&|\tilde{\mathbf{B}}^{(s)-1}|\mathcal{N}\left(\tilde{\mathbf{B}}^{(s)-1}\mathbf{y}_{\mathsf{bn}}(t)-\tilde{\mathbf{B}}^{(s)-1}\tilde{\mathbf{c}}^{(s)};\boldsymbol{\mu}^{(m)},\boldsymbol{\Sigma}^{(m)}\right).\end{aligned} \tag{5}$$

Since $|\tilde{\mathbf{B}}^{(s)-1}|$ is a constant for each speaker across all states and hence does not affect state posterior or best path calculations given the speaker s, a single class CMLLR transform can be implemented as a speaker dependent affine transform applied to normalise the input features, whose weight matrix and bias vector are $\tilde{\mathbf{B}}^{(s)-1}$ and $-\tilde{\mathbf{B}}^{(s)-1}\tilde{\mathbf{c}}^{(s)}$. Because of the simplicity in implementation, CMLLR instead of MLLR is often used for SAT. During SAT, the adaptation transforms and the canonical model parameters are updated in an interleaved fashion until the estimate of the canonical model converges or a desired number of iterations is reached.

2.3. Paramterised sigmoid based speaker adaptation

In addition to MLLR and CMLLR, this work also used the p-sigmoid parameterised activation function adaptation approach [23]. The p-sigmoid function for speaker adaptation used a trainable output value scaling factor for each hidden unit, which can be written as

$$f_i^{(s)}\left(a_{li}(t)\right)=\alpha_{li}^{(s)}/\left(1+\exp(-a_{li}(t))\right) \tag{6}$$

where i is the hidden unit index, s is a speaker, and $\alpha_{li}^{(s)}$ is the SD activation function parameter for speaker s and hidden unit i at layer l. Only the first hidden layer activation function was adapted in this paper as we observed overfitting when adapting more hidden layers in this task. Note that to stabilise the adaptation, gradient clipping is required when training $\alpha_{li}^{(s)}$.

3. SAT and Speaker Adaptation of Jointly Trained Tandem Systems

3.1. Joint MPE training of Tandem system

For the conventional Tandem system introduced in Section 2.1, the BN DNN and the GMMs are trained separately, where the BN features are not optimised for the GMMs. For the joint MPE trained Tandem system, the DNN parameters $\{\mathbf{W}_{1\cdots L},\mathbf{b}_{1\cdots L}\}$ and the GMM parameters $\{\phi^{(jm)},\boldsymbol{\mu}^{(j)},\boldsymbol{\Sigma}^{(j)}\}$ are trained concurrently using SGD and the MPE criterion. During this training, not only are the GMMs are estimated using the BN features, but the BN features are also optimised for the GMMs. The joint MPE training procedure includes the following steps [6].

(i) A BN DNN is first trained using the CE criterion using the alignments generated by a pre-trained system.
(ii) Once an initial BN DNN has been obtained, the layers after the BN layer are removed. The BN layer activation function is changed to the linear function to generate BN features.
(iii) The BN layer linear activation function is converted to an almost equivalent ReLU function by increasing the bias values of the layer by six times the standard deviation of the linear BN features.
(iv) A set of monophone GMM-HMMs are constructed using a maximum likelihood (ML) criterion based on $\mathbf{y}_{\mathsf{bn}}(t)$, the ReLU output values of the BN layer. These systems are denoted BN-GMM-HMMs.
(v) The monophone BN-GMM-HMM system is extended to an initial ML tied-state triphone GMM-HMM system following the HTK recipe [30, 31], which is then reconstructed using a *two-model re-estimation* method to acquire more accurate state-level alignments to generate better decision trees.
(vi) Finally, the BN DNN and the GMMs are jointly optimised using SGD based on the MPE criterion.

Note that unlike the conventional Tandem systems whose decision trees are normally constructed based on the standard acoustic features, the decision trees for the BN-GMM-HMM system are built based on the CE BN features, which is a better approximation to the final MPE jointly trained BN features and can have better performance [6]. Furthermore, to get good performance with the SGD based MPE training, I-smoothing [32], the use of a dynamic maximum mutual information (MMI) prior, and percentile based variance flooring are all adapted from the extended Baum-Welch (EBW) framework to the SGD based framework [6]. Moreover, to make the model training stable and effective, a number of methods, such as amplifying the GMM learning rate and clipping the update values based on a relative threshold are adopted [6].

3.2. Speaker adaptive training

According to Eqn. (5), the SD CMLLR transforms can be implemented as affine transforms which are used to normalise the BN features. In this paper, such CMLLR transforms are estimated using the traditional forward-backward algorithm, and used by converting them to a special SD fully-connected DNN layer with a linear activation function, whose weight matrix

$\mathbf{W}^{(s)}_{\text{cmllr}}$ and bias vector $\mathbf{b}^{(s)}_{\text{cmllr}}$ are the same as those of the CMLLR affine transforms, i.e.

$$\mathbf{W}^{(s)}_{\text{cmllr}} = \tilde{\mathbf{B}}^{(s)-1}, \ \ \mathbf{b}^{(s)}_{\text{cmllr}} = \tilde{\mathbf{B}}^{(s)-1}\tilde{\mathbf{c}}^{(s)}.$$

This SD CMLLR linear layer is inserted between the BN layer and the GMMs, whose parameters are frozen during SGD based joint training. In an analogous manner to traditional CMLLR-based GMM-HMM SAT, the CMLLR transforms are updated in an interleaved fashion after each SGD based joint training epoch. The detailed steps taken to incorporate CMLLR-based SAT to MPE joint training are listed below.

(i) Train the BN-GMM-HMM system using the ML criterion and use this system to estimate an initial CMLLR transform for each speaker using the data from that speaker;

(ii) Jointly train the BN DNN parameters and the GMM parameters using MPE criterion for one epoch, with the weights and biases from the most recent SD CMLLR transform inserted as an SD layer after the BN layer. The parameters of the SD layer are switched to those of the CMLLR transform for the current speaker, and are not updated during the next epoch of joint training;

(iii) Re-estimate the CMLLR transforms for all speakers based on the most recent MPE jointly trained BN features and GMMs;

(iv) Repeat step (ii) and step (iii) until the training converges or the required number of iterations is reached.

It worth noting that the CMLLR transforms are estimated based on the optimised BN features instead of the standard acoustic features. At test-time, the CMLLR transforms are estimated iteratively in order using BN-GMM-HMMs generated at the end of each epoch of the joint training in step (iii).

3.3. Unsupervised speaker adaptation

As well as test-time CMLLR transforms that are used with the SAT models, MLLR and p-sigmoid activation function adaptation methods are also used at test-time in an unsupervised fashion. MLLR is applied since it can remove the constraint from CMLLR based SAT and allows the use of multiple sets of linear transforms for each speaker and distinct transforms for the GMM mean and variance values.

The use of p-sigmoid speaker adaptation is also investigated in this paper since it has been observed to be complementary to CMLLR and MLLR in previous studies [23]. In this paper, p-sigmoid was used for test-time adaptation rather than SAT. In contrast to previous work [21, 23] where the $\alpha^{(s)}_{li}$ were estimated based on a DNN acoustic model with a softmax output layer, here the p-sigmoid parameters were estimated using a jointly trained Tandem model with a GMM output layer. To be consistent with MLLR and CMLLR, ML based sequence training rather than the CE based frame-level training [21, 23] is used for p-sigmoid adaptation in this paper, where the partial derivatives of the ML criterion w.r.t. the GMM observation density function are the *ML state occupancies* calculated at the sequence level using the *forward-backward* algorithm [6]. The detailed steps for p-sigmoid adaptation consists of the following steps:

(i) Generate phone sequence labels from the hypotheses using the target system for adaptation;

(ii) Initialise the $\alpha^{(s)}_{li}$ for all speakers to 1.0, to make the p-sigmoid functions start as standard sigmoid functions;

(iii) Find $\alpha^{(s)}_{li}$ for all speakers using SGD based on the ML state occupancies. SGD is used to update the $\alpha^{(s)}_{li}$ once per utterance.

4. Experimental Setup

Experiments were conducted using the data from the 2017 English Multi-Genre Broadcast (MGB3) challenge [33]. The data consists of audio from BBC television programmes. The data contains a wide range of genres such as comedy, drama and sports shows. A total of 375 hours of audio data with associated subtitles is available for acoustic model training. Lightly supervised decoding and selection was used to extract 275 hours for training [34, 35, 8]. The reference segmentation was used with automatic speaker clustering resulting in 192,209 utterances and 13,467 speaker clusters. A 5.5 hours development set, dev17b, was also supplied. For this data set, an automatic audio segmentation using a DNN based segmenter [36] trained on the MGB3 data was used and it resulted in 5201 utterances and 145 speaker clusters. The dev17b data set was used to test the performance of the systems and the results will be described in Section 5.

All experiments were conducted with HTK 3.5 [31, 27]. The GMM-HMM systems were trained on 52-dimensional PLP+Δ+Δ^2+Δ^3 features and around 9000 context dependent (SD) states were used. The GMMs have 16 Gaussian components per state, except for the 3 silence states, which have 32 Gaussian components per state. Both BN DNN and the Hybrid DNN were trained on the 40-dimensional log Mel-filter bank features which was expanded with Δ features. A concatenation of 9 consecutive feature vectors were used as the input to the DNNs. Utterance level mean normalisation and show-segment level variance normalisation were applied [8]. The BN DNN had a structure $720 \times 1000^4 \times 39 \times 1000 \times 9000$, where the BN feature size is 39. The Hybrid DNN had a structure $720 \times 1000^5 \times 9000$. Sigmoid activation functions were used in both DNN acoustic models and BN DNNs. The Hybrid SI system was first trained using the CE criterion and then sequence trained using the MPE criterion [37].

For the conventional Tandem system builds, which is referred to as Tandem system in this section, the 39-dimensional BN features were concatenated with the 52-dimensional PLP features. An HLDA transform was applied for PLP features and a global semi-tied transform was applied for BN features, thus the combined dimensionality of the Tandem features was reduced from 91 to 78. The Tandem SI system was trained using the MPE criterion and the Tandem SAT system was built CMLLR followed by MPE. For the joint MPE training of the Tandem system, which is referred to as joint-Tandem (J-Tandem) system, the GMM learning amplification factor was set to 20 and the other aspects of the configurations were as described in [6]. A trigram language model (LM) with a 64K word lexicon was trained on the audio subtitles and 650M word tokens of supplied BBC subtitles. All the systems outputs used the trigram LM and confusion network (CN) decoding [38].

5. Experimental Results

5.1. Joint training of Tandem system

The WERs of Tandem, Hybrid and joint Tandem (J-Tandem) SI systems are given in Table 1 which shows that both Hybrid and J-Tandem systems give about a 10% relative WER reduction (rWERR) over the Tandem system. The J-Tandem SI system outperforms the Hybrid SI system by 0.2% absolute WER. It is worth noting that while the J-Tandem WER is higher than the interleaved time-delay DNN (TDNN) long short-term memory

(LSTM) system tested on the same data set in [33], but the performance gap is expected to be reduced by using more powerful BN architectures (e.g. based on TDNNs or LSTMs).

System	%WER
Tandem SI	28.6
Hybrid SI	25.9
J-Tandem SI	25.7

Table 1: *%WER of speaker independent (SI) Tandem, Hybrid and joint-Tandem (J-Tandem) MPE systems.*

5.2. Speaker adaptive joint training of Tandem system

In this section, the performance of the Tandem SAT and J-Tandem SAT systems are compared. In addition the CMLLR test-set adaptation of the SAT models and the use of MLLR at test-time are also investigated.

System	CMLLR	MLLR	%WER
Tandem SI	✗	✗	28.6
Tandem SAT	✓	✗	26.3
	✓	✓	25.8
J-Tandem SI	✗	✗	25.7
J-Tandem SAT	✓	✗	24.8
	✓	✓	24.8

Table 2: *%WER for Tandem SAT and Joint-Tandem systems with CMLLR SAT and test-time MLLR. The adaptation supervision was from the Hybrid SI system.*

The WERs for the Tandem SAT and J-Tandem SAT systems are shown in Table 2. To allow a straight-forward comparison, the adaptation supervision for both Tandem SAT and J-Tandem SAT systems was taken from the Hybrid SI system in Table 1. Table 2 shows that the use of CMLLR SAT training reduces the WER of the Tandem SI system by 2.3% absolute and the test-time MLLR speaker adaptive gives an additional 0.5% absolute WER reduction, resulting in a 25.8% WER. For the J-Tandem system, the WER reduction from SAT training is smaller than that from the Tandem system. The WER drops from 25.7% to 24.8% by using CMLLR SAT and in addition using MLLR test-time speaker adaptation results in no further performance gains. By comparing the Tandem SAT system and J-Tandem SAT system, it can be seen that the J-Tandem SAT system gives about a 4% relative WER reduction (rWERR) over the Tandem SAT system. Similarly, there is about a 4% rWERR comparing the J-Tandem SAT system to either the Hybrid SI or J-Tandem SI systems.

System	CMLLR	MLLR	Supervision	
			Hybrid	J-Tandem
J-Tandem	✓	✓	24.8	24.7

Table 3: *%WER for Joint-Tandem (J-Tandem) systems with CMLLR SAT and test-time MLLR using two different supervisions.*

As can be seen from Table 1, the performance of the J-Tandem SI system is slightly better than that of the Hybrid SI system. Thus, a J-Tandem SI system can also be used to provide the adaptation supervision. Table 3 shows the WERs of the J-Tandem SAT system when using either Hybrid SI or the J-Tandem SI system outputs as supervision. It shows that using J-Tandem SI system as the supervision, the J-Tandem SAT system is 0.1% better than that using Hybrid SI supervision.

One of the key reasons for using Tandem and J-Tandem systems is that they are complementary to Hybrid systems. The improvements from using confusion network combination (CNC) [39] to combine the Hybrid SI system with the Tandem and J-Tandem systems is shown in Table 4.

System		%WER	%rWERR
Hybrid SI $\oplus$ Tandem	SI	25.5	1.5
	SAT	24.5	5.4
Hybrid SI $\oplus$ J-Tandem	SI	24.6	5.0
	SAT	24.2	6.6

Table 4: *%WER of CNC of Hybrid SI system and Tandem and J-Tandem systems. The relative %WER reduction (%rWERR) is calculated over the Hybrid SI system. $\oplus$ denotes CNC.*

Table 4 shows that the CNC of the Hybrid SI system and J-Tandem SI system gives a 5% rWERR over the Hybrid SI system. When combining with Tandem SAT system, it gives an additional 1.6% rWERR. In contrast, the CNC of Hybrid SI and Tandem SI and SAT systems gives rWERRs of 1.5% and 5.4%, respectively.

5.3. Combination of SAT and parameterised sigmoid speaker adaptation

The combination of the SAT and p-sigmoid based test-time speaker adaptation was also investigated for the J-Tandem SAT systems and the results are shown in Table 5. The p-sigmoid SD parameters are applied on the first hidden layer of the BN DNN.

System	CMLLR	p-sigmoid	%WER
J-Tandem	✗	✗	25.7
	✗	✓	25.6
	✓	✗	24.8
	✓	✓	24.8

Table 5: *%WER for Tandem and Joint-Tandem (J-Tandem) SI and SAT systems with p-sigmoid speaker adaptation.*

In the top half of Table 5, it can be seen that applying p-sigmoid adaptation yields only 0.1% absolute WER reduction over the J-Tandem SI system. When applying p-sigmoid speaker adaptation in addition to CMLLR on the J-Tandem SAT system, as shown in the bottom half of the table, there is no performance gain.

6. Conclusions

This paper has investigated the use of CMLLR-based speaker adaptive training for a jointly MPE trained Tandem system. In this system the bottleneck features and the Gaussian parameters are jointly trained by SGD and in addition CMLLR transforms are applied in both training and test. Furthermore the use of p-sigmoid based unsupervised speaker adaptation was also investigated. Speech recognition experiments on the multi-genre broadcast MGB3 data showed that the jointly trained Tandem SAT systems could yield reductions in WER compared to the conventional Tandem SAT system, and also a Hybrid SI system. In both cases the jointly trained Tandem SAT system gave about a 4% lower WER. Furthermore, jointly trained Tandem systems are more complementary to Hybrid systems than conventional Tandem systems, and reduce the error rates further when using system combination. However, the combination of different unsupervised speaker adaptation approaches did not yield further improvements for the jointly trained Tandem SAT system.

7. References

[1] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A. Mohamed, N. Jaitly, A. Senior, et al. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal Processing Magazine*, 29(6):82–97, 2012.

[2] H. Hermansky, D.P.W. Ellis, and S. Sharma. Tandem connectionist feature extraction for conventional HMM systems. *Proc. ICASSP*, pp. 1635–1638, 2000.

[3] F. Grézl, M. Karafiát, S. Kontár, and J. Černocký. Probabilistic and bottle-neck features for LVCSR of meetings. *Proc. ICASSP*, pp. 757–760, 2007.

[4] E. Variani, E. McDermott, and G. Heigold. A Gaussian mixture model layer jointly optimized with discriminative features within a deep neural network architecture. *Proc. ICASSP*, pp. 4270–4274, 2015.

[5] Z. Tüske, P. Golik, R. Schlüter, and H. Ney. Speaker adaptive joint training of Gaussian mixture models and bottleneck features. *Proc. ASRU Workshop*, pp. 596–603, 2015.

[6] C. Zhang and P.C. Woodland. Joint optimisation of tandem systems using Gaussian mixture density neural network discriminative sequence training. *Proc. ICASSP*, pp. 5015–5019, 2017.

[7] P. Swietojanski, A. Ghoshal, and S. Renals. Revisiting hybrid and GMM-HMM system combination techniques. *Proc. ICASSP*, pp. 6744–6748, 2013.

[8] P.C. Woodland, X. Liu, Y. Qian, C. Zhang, M.J.F. Gales, P. Karanasou, P. Lanchantin, and L. Wang. Cambridge University transcription systems for the multi-genre broadcast challenge. *Proc. ASRU Workshop*, pp. 639–646, 2015.

[9] P.C. Woodland. Speaker adaptation for continuous density HMMs: a review. *Proc. ISCA Tutorial and Research Workshop (ITRW) on Adaptation Methods for Speech Recognition*, pp. 11–19, 2001.

[10] T. Anastasakos, J. McDonough, R. Schwartz, and J. Makhoul. A compact model for speaker-adaptive training. *Proc. ICSLP*, pp. 1137–1140, 1996.

[11] J.L. Gauvain and C.H. Lee. Maximum a posteriori estimation for multivariate Gaussian mixture observations of Markov chains. *IEEE transactions on speech and audio processing*, 2(2):291–298, 1994.

[12] C.J. Leggetter and P.C. Woodland. Maximum likelihood linear regression for speaker adaptation of continuous density hidden Markov models. *Computer Speech and Language*, 9(2):171–185, 1995.

[13] M.J.F. Gales. Cluster adaptive training of hidden Markov models. *IEEE Transactions on Speech and Audio Processing*, 8(4):417–428, 2000.

[14] M.J.F. Gales. Maximum likelihood linear transformations for HMM-based speech recognition. *Computer Speech and Language*, 12(2):75–98, 1998.

[15] M. Karafiát, L. Burget, P. Matějka, O. Glembek, and J. Černockỳ. iVector-based discriminative adaptation for automatic speech recognition. *Proc. ASRU Workshop*, pp. 152–157, 2011.

[16] G. Saon, H. Soltau, D. Nahamoo, and M. Picheny. Speaker adaptation of neural network acoustic models using i-vectors. *Proc. ASRU Workshop*, pp. 55–59, 2013.

[17] Y. Miao, H. Zhang, and F. Metze. Speaker adaptive training of deep neural network acoustic models using i-vectors. *IEEE/ACM Transactions on Audio, Speech and Language Processing*, 23(11):1938–1949, 2015.

[18] 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. *Proc. Eurospeech*, pp. 2171–2174, 1995.

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

[20] S.M. Siniscalchi, J. Li, and C. H. Lee. Hermitian polynomial for speaker adaptation of connectionist speech recognition systems. *IEEE Transactions on Audio, Speech, and Language Processing*, 21(10):2152–2161, 2013.

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

[22] Y. Zhao, J. Li, J. Xue, and Y. Gong. Investigating online low-footprint speaker adaptation using generalized linear regression and click-through data. *Proc. ICASSP*, pp. 4310–4314, 2015.

[23] C. Zhang and P.C. Woodland. DNN speaker adaptation using parameterised sigmoid and ReLU hidden activation functions. *Proc. ICASSP*, pp. 5300–5304, 2016.

[24] F. Seide, G. Li, and D. Yu. Conversational speech transcription using context-dependent deep neural networks. *Proc. Interspeech*, pp. 437–440, 2011.

[25] C. Wu, P. Karanasou, M.J.F. Gales, and K.C. Sim. Stimulated deep neural network for speech recognition. *Proc. Interspeech*, 2016.

[26] Z. Huang, S.M. Siniscalchi, I.-F. Chen, J. Li, J. Wu, and C.-H. Lee. Maximum a posteriori adaptation of network parameters in deep models. *Proc. ICASSP*, 2015.

[27] C. Zhang and P.C. Woodland. A general artificial neural network extension for HTK. *Proc. Interspeech*, 2015.

[28] X. Liu, M.J.F. Gales, and P.C. Woodland. Automatic complexity control for HLDA systems. *Proc. ICASSP*, 2003.

[29] M.J.F. Gales. Semi-tied covariance matrices for hidden Markov models. *IEEE Transactions on Speech and Audio Processing*, 7(3):272–281, 1999.

[30] S. J. Young, J.J. Odell, and P. C. Woodland. Tree-based State Tying for High Accuracy Acoustic Modelling. *Proc. ARPA Human Language Age Technology Workshop*, pp. 307–312, 1994.

[31] S.J. Young, G. Evermann, M.J.F. Gales, T. Hain, D. Kershaw, X. Liu, G. Moore, J.J. Odell, D. Ollason, D. Povey, V. Valtchev, and P.C. Woodland. *The HTK book (for HTK version 3.5)*. University of Cambridge, 2015.

[32] D. Povey and P.C. Woodland. Minimum phone error and I-smoothing for improved discriminative training. *Proc. ICASSP*, pp. 105–108, 2002.

[33] Y. Wang, X. Chen, M.J.F. Gales, A. Ragni, and J.H.M. Wong. Phonetic and graphemic systems for multi-genre broadcast transcription. *Proc. ICASSP*, pp. 5899–5903, 2018.

[34] P. Lanchantin, M.J.F. Gales, P. Karanasou, X. Liu, Y. Qian, L. Wang, P.C. Woodland, and C. Zhang, Selection of Multi-Genre Broadcast data for the training of automatic speech recognition systems, *Proc. Interspeech*, pp. 3057–3061, 2016.

[35] P. Bell, M.J.F. Gales, T. Hain, J. Kilgour, P. Lanchantin, X. Liu, A. McParland, S. Renals, O. Saz, M. Wester, and P.C. Woodland. The MGB challenge: Evaluating multi-genre broadcast media recognition. *Proc. ASRU Workshop*, pp. 687–693, 2015.

[36] L. Wang, C. Zhang, P.C. Woodland, M.J.F. Gales, P. Karanasou, P. Lanchantin, X. Liu, and Y. Qian. Improved DNN-based segmentation for multi-genre broadcast audio. *Proc. ICASSP*, pp. 5700–5704, 2016.

[37] K. Veselỳ, A. Ghoshal, L. Burget, and D. Povey. Sequence discriminative training of deep neural networks. *Proc. Interspeech*, pp. 2345–2349, 2013.

[38] L. Mangu, E. Brill, and A. Stolcke. Finding consensus in speech recognition: word error minimization and other applications of confusion networks. *Computer Speech and Language*, 14(4):373–400, 2000.

[39] G. Evermann and P.C. Woodland. Posterior probability decoding, confidence estimation and system combination. *Proc. Speech Transcription Workshop*, 2000.