COMBINING I-VECTOR REPRESENTATION AND STRUCTURED NEURAL NETWORKS FOR RAPID ADAPTATION

Chunyang Wu, Penny Karanasou, Mark J.F. Gales

Cambridge University Engineering Dept,
Trumpington St., Cambridge, CB2 1PZ U.K.
Email: {*cw564, pk407, mjfg*}@*eng.cam.ac.uk*

ABSTRACT

Rapid adaptation of deep neural networks (DNNs) with limited unsupervised data remains a significant challenge. This paper investigates the combination of two schemes that have been proposed to address this problem: i-vector representations and multi-basis adaptive neural networks (MBANNs). Two approaches for combining these schemes together are described. The first uses i-vectors as one of the input features to the MBANN. The purpose is to combine the speaker representation of the i-vector with the network interpolation of the MBANN scheme. The second approach aims to reduce the computational cost, and improve the robustness to hypothesis errors, of the MBANN scheme. Here i-vectors are used to predict the interpolation weights of the MBANN scheme. This removes the need for an initial decoding pass, and alignment, which was previously used. These approaches are evaluated using acoustic and language models trained on a U.S. English Broadcast News (BN) transcription task. Two distinct sets of test data are examined. The first from the BN task, yields test data acoustically matched to the training data. The second, acoustically mismatched, set is from Youtube videos. The performance gains from these schemes is found to be sensitive to the level of mismatch between training and test.

***Index Terms*—** Rapid adaptation, structured deep neural networks, i-vectors, acoustic modeling

1. INTRODUCTION

Deep neural network (DNN) acoustic models have leaped forward in recent years, outperforming the traditional Gaussian Mixture Hidden Markov Model (GMM-HMM) in large vocabulary continuous speech recognition tasks [1, 2, 3]. However, according to recent comparative studies [4], a DNN does not always manage to handle the impact from background noise or speaker variations. Thus, effective adaptation techniques on DNNs remain to be explored. Neural networks are difficult to be adapted though. This is because a large amount of parameters is likely to be over-fitted with limited adaptation data, whilst it is hard to split them into meaningful groups to apply similar transforms as that used in GMM-HMMs.

There have already been a number of attempts in literature on neural-network based compensation and adaptation. Feature-space transforms like CMLLR [5], initially developed for GMM-HMM models, can be directly deployed on a DNN-based system. An important field of DNN adaptation concentrates on auxiliary indicators at the network front-end to compensate the robustness under mismatched conditions, *e.g.*, using i-vector [6, 7, 8], automatic speaker codes [9, 10], or external heterogeneous knowledge [11] as additional input features. The speaker i-vector is a low-dimensional fixed-length representation of the speaker space, which can rapidly adapt the neural network in an unsupervised fashion. Extensions of the i-vector adaptation include acoustic factorization [12] and informative priors [13].

In addition to the augmented feature models, structure-based adaptation techniques have also been investigated. The concept of structural adaptive neural network is to impose interpretable modules to the structure, exposing meaningful parameters to adapt the system efficiently. These transformation-based schemes add additional linear hidden layers as speaker-dependent (SD) transforms prior to the input layer [14], to hidden layers [15] or prior to the output layer [16]. In [17], additional bottom normalization layers along with i-vectors are introduced to project raw acoustic features into a speaker-normalized space. The activation-based approach [18] uses the Hermitian polynomial as the adaptable activation function. Apart from modifying the components of a generic multi-layer perceptron, delicate adaptive network topologies have also been investigated. [19] introduces a scaling factor on hidden-layer activations and in [20], the differentiable pooling technique is used to obtain the speaker-dependent compensation from a hidden-activation candidate pool. In [21, 22], a set of conjugated hidden-layer transformations are interpolated with speaker-dependent factors and in [23], the multi-basis adaptive neural network imposes paralleled subnetworks as well as an adaptable combination module to handle the acoustic distortions. However, many of these structured models still involve a large number of parameters to adapt, hence they would be over-fitted in rapid adaptation scenarios with limited data.

In this paper, we investigate the methods for rapid adaptation of DNNs. First, an extended multi-basis system with i-vectors as auxiliary inputs is proposed. The introduced i-vector is expected to reinforce the robustness of the basis hidden-layer representation. Meanwhile, the reinforced basis outputs are combined as before following the speaker-dependent interpolation scheme. However, structured neural networks usually require a second decoding pass in order to acquire sensible hypotheses for optimizing a speaker-dependent transform. This is not applicable to stringent real-time systems, like voice search. Meanwhile, the adaptation performance cannot be

The research leading to these results was supported by EPSRC grant EP/I031022/1 (Natural Speech Technology) and the Intelligence Advanced Research Projects Activity (IARPA) via Department of Defense U.S. Army Research Laboratory (DoD/ARL) contract number W911NF-12-C-0012. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. Disclaimer: The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of IARPA, DoD/ARL, or the U.S. Government. Supporting data for this paper is available at the https://www.repository.cam.ac.uk/handle/1810/253376 data repository.

978-1-4799-9988-0/16/$31.00 ©2016 IEEE

guaranteed with hypotheses of poor quality. Beyond the traditional second-pass framework, a fast and rapid predictive module for the structural multi-basis system is put forward to directly estimate the SD transforms from i-vectors.

The experiments are conducted on the utterance-level unsupervised adaptation of a large vocabulary continuous English broadcast news transcription task. It is shown that gains are obtained by using the multi-basis DNN with i-vector inputs for test sets with acoustic conditions that match the training one. The predictive adaptation module achieves similar performance in low-error-rate situations and, more importantly, outperforms the traditional tuning scheme in highly-mismatched scenarios.

The rest of this paper is organized as follows. The i-vector technique and the multi-basis adaptive one are briefly reviewed in Section 2. In Section 3, we present the two types of adaptation combination: the combined multi-basis system with i-vector input features and the fast multi-basis transform predictive module from i-vector representations. Experimental results are reported in Section 4. This paper is concluded in Section 5.

2. I-VECTORS AND STRUCTURED NEURAL NETWORKS

2.1. I-vector Estimation

Following [12], the i-vector extraction is performed by a type of model-based CAT estimation [24] where the HMM model is replaced by a GMM model, requiring no transcriptions of the data. The intrinsic variability of phonemes is represented by a canonical model $\mathcal{M}$ which is a GMM universal background model (UBM) with M mixture components [25]. It is defined by a mean super-vector of component means $\boldsymbol{\mu}_0^{(m)}$, diagonal component covariance matrices $\boldsymbol{\Sigma}^{(m)}$ and mixture coefficients $\omega^{(m)}$. The input acoustic feature vectors $\boldsymbol{x}_t \in \mathbb{R}^D$ are treated as samples generated by $\mathcal{M}$.

For training, we estimate one i-vector for each speaker using all the data belonging to it and this i-vector is constant across all utterances of the speaker. Each speaker is represented by a point in the “speaker eigenspace” spanned by the i-vectors. There is a linear dependence between the speaker-adapted means (i.e. speaker-dependent super-vectors) and the canonical means, which for a particular Gaussian component $m \in M$ is given by

$$\boldsymbol{\mu}^{(sm)} = \boldsymbol{\mu}_0^{(m)} + \boldsymbol{M}^{(m)}\boldsymbol{\lambda}_{iv}^{(s)} \tag{1}$$

where $\boldsymbol{\mu}^{(sm)}$ is the m-th component of speaker-dependent super-vector, $\boldsymbol{M}^{(m)}$ is the factor submatrix for component m of size $D \times P$, representing P bases spanning the subspaces with the highest variability in the mean super-vector space and $\boldsymbol{\lambda}_{iv}^{(s)}$ is the P-dimensional i-vector of speaker s.

To extract the initial speaker i-vectors, a speaker-dependent (SD) model using all the data belonging to the speaker is trained for extracting a mean super-vector. Principal component analysis (PCA) is then applied to these super-vectors to obtain the speaker i-vectors that span the P-space. Next, maximum-likelihood estimation of the model parameters and of the i-vectors is performed. The auxiliary function to be maximized is

$$\begin{aligned} &Q(\mathcal{M}, \boldsymbol{\lambda}_{iv}^{(s)}; \hat{\mathcal{M}}, \hat{\boldsymbol{\lambda}}_{iv}^{(s)}) = \\ &-\frac{1}{2}\sum_{s,t,m} \gamma_t^{(m)}(s)(\boldsymbol{x}_t - \boldsymbol{\mu}^{(sm)})^T \boldsymbol{\Sigma}^{(m)-1}(\boldsymbol{x}_t - \boldsymbol{\mu}^{(sm)}) \end{aligned} \tag{2}$$

where $\mathcal{M}$ and $\boldsymbol{\lambda}_{iv}^{(s)}$ are the canonical model and i-vectors to be estimated; $\hat{\mathcal{M}}$ and $\hat{\boldsymbol{\lambda}}_{iv}^{(s)}$ are the “old” ones. $\gamma_t^{(m)}(s)$ is the posterior probability of Gaussian component m at time t determined using $\hat{\mathcal{M}}$ and the speaker i-vectors $\hat{\boldsymbol{\lambda}}_{iv}^{(s)}$. The training procedure uses the Expectation-Maximization (EM) algorithm to estimate the parameters. The reader is referred to [12] for a more detailed presentation of the i-vector training procedure.

2.2. Multi-basis Adaptive Neural Network

The multi-basis adaptive neural network [23] is a structured neural-network topology requiring a very small set of adaptive parameters for rapid adaptation. Different from a conventional multi-layer perceptron, a set of distinct parallel sub-networks are introduced, referred to as the *bases*. These bases share the same input and their outputs are subsequently combined in the *combination* stage. Optionally, common layers can be introduced before the basis splitting or after their combination. One basis consists of several hidden layers. The neurons between two successive layers are restrictedly connected in the same basis while no inter-basis connections are allowed.

The combination part is specified as the adaptation transform to handle acoustic variations among speakers. In this work the interpolation scheme is followed, in which the outputs of the bases are interpolated with speaker-dependent weights before being propagated to the subsequent layers,

$$\bar{\boldsymbol{h}}(\boldsymbol{x}_t^{(s)}) = \sum_{k=1}^{K} \lambda_k^{(s)} \boldsymbol{h}^k(\boldsymbol{x}_t^{(s)}) \tag{3}$$

where $\boldsymbol{h}^k$ stands for the output of the k-th basis and K is the number of the bases. The interpolation weights are defined as the speaker-dependent transform, named as the *basis weight vector*

$$\boldsymbol{\lambda}_{mb}^{(s)} = \left[\lambda_1^{(s)}, \ldots, \lambda_K^{(s)}\right]^T. \tag{4}$$

The multi-basis transform $\boldsymbol{\lambda}_{mb}^{(s)}$ for both the training and the testing phases in this paper is optimized via stochastic gradient descent on the cross-entropy criterion

$$\mathcal{L}_{CE} = -\sum_{s}\sum_{t\in\mathbb{I}_s} \log p\left(y_t|\boldsymbol{x}_t; \boldsymbol{\lambda}_{mb}^{(s)}\right) \tag{5}$$

where $\mathbb{I}_s$ the index set of the speaker's frames; y_t is the state label and the gradient is calculated by

$$\frac{\partial \mathcal{L}_{CE}}{\boldsymbol{\lambda}_{mb}^{(s)}} = \frac{\partial \bar{\boldsymbol{h}}^T}{\partial \boldsymbol{\lambda}_{mb}^{(s)}} \frac{\partial \mathcal{L}_{CE}}{\partial \bar{\boldsymbol{h}}} = \left[\boldsymbol{h}^1, \ldots, \boldsymbol{h}^K\right]^T \frac{\partial \mathcal{L}_{CE}}{\partial \bar{\boldsymbol{h}}} \tag{6}$$

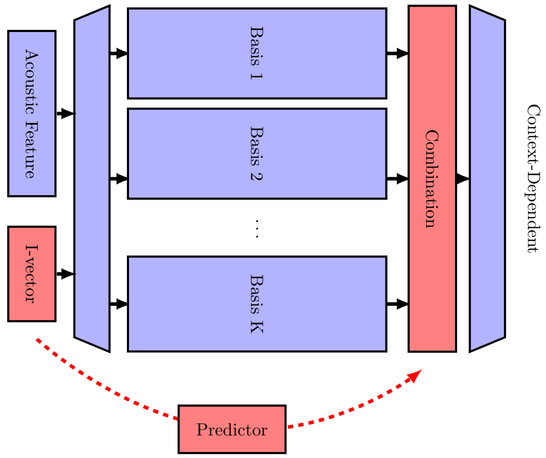
where $\frac{\partial \mathcal{L}_{CE}}{\partial \bar{\boldsymbol{h}}}$ is given by the back-propagation algorithm. The multi-basis DNN can be optimized in an adaptive training scheme as described in [23]. With the help of this multi-basis configuration, the bases would then jointly capture and model the diversified characteristics among the regions in the acoustic space.

3. COMBINING I-VECTORS AND MULTI-BASIS DNN

3.1. Multi-basis DNN with I-vector Input Features

The i-vector adaptation approach explicitly informs the neural network of acoustic identifiers along with the acoustic features at a very primitive stage and expects that the interaction between these inhomogeneous features would result in robust and compact speaker-invariant abstraction in higher layers. However, the multi-basis adaptive neural network delays the adaptation phase to a latter stage where hidden activations are interpolated. The complementarity of these different strategies warrants some further investigation.

Fig. 1. Combining Multi-basis DNN with I-vectors. Adaptable modules are colored in red.



The proposed multi-basis DNN with i-vector input features is shown in Fig. 1. The waveform feature $\boldsymbol{x}_t^{(s)}$, *e.g.* PLP or filter bank, is concatenated with the i-vector $\boldsymbol{\lambda}_{iv}^{(s)}$ belonging to its associated speaker to form the input propagated to each of the multiple bases. The introduced i-vector is expected to reinforce the robustness of the basis hidden-layer representation. Meanwhile, the reinforced basis outputs are combined as before following the speaker-dependent interpolation scheme.

3.2. Predictive Multi-basis Transform Using I-vectors

In the multi-basis neural network, an estimation of the speaker-dependent transform $\boldsymbol{\lambda}_{mb}^{(s)}$ is required when evaluating an unknown speaker. Normally, it is optimized according the decoding hypotheses from a sensible speaker-independent (SI) system. A second-pass scheme is of necessity to firstly obtain the alignments of SI hypothesis, which would not be efficient enough in stringent real-time systems. Additionally, the performance cannot be guaranteed under highly-mismatched conditions with hypotheses of poor quality.

The fast predictive estimation module of the multi-basis transform is put forward to prevent these hazards. Apart from the hypothesis alignment, auxiliary indicators like the i-vectors, which contains rich information of the speaker acoustic properties, can be utilized to better estimate a multi-basis speaker transform. In this work, a predictor, as illustrated in the dashed-line part of Fig. 1, is trained to establish a mapping from the i-vector $\boldsymbol{\lambda}_{iv}^{(s)}$ to the basis interpolation weights $\boldsymbol{\lambda}_{mb}^{(s)}$,

$$\boldsymbol{\lambda}_{mb}^{(s)} = \boldsymbol{f}_{pred}(\boldsymbol{\lambda}_{iv}^{(s)}) \tag{7}$$

where $\boldsymbol{f}_{pred}(\cdot)$ represents the prediction model. The adaptation performance of the multi-basis neural network is then undertaken by the precision of the prediction mappings, which is irrelevant to the quality of decoding hypothesis. Besides, the predictive procedure can be used in the decoding phase in an efficient way.

The mismatch between the distribution of predicted basis weights and that of the original ones is likely to degrade the performance. In order to reduce the degradation caused by this sort of difference, an interleaving mode is utilized to update the multi-basis network and the predictor jointly. In each iteration, the predictor is trained on the estimated $\{\boldsymbol{\lambda}_{mb}^{(s)}\}_{esti}$ of the training set from the current multi-basis system; the re-estimated transforms $\{\boldsymbol{\lambda}_{mb}^{(s)}\}_{pred}$ given by this trained predictor is then used to update the neural network for the next iteration. The conjugate pair of neural network and predictor of each iteration is then used in evaluation.

4. EXPERIMENTS

The proposed input-feature combination scheme and the predictive transform approach were tested on a U.S. English broadcast news (BN) transcription task. Each approach is evaluated for rapid utterance-level unsupervised adaptation. The training set of this task included the 144-hour 1996 & 1997 Hub-4 English Broadcast News Speech dataset (`LDC97S44, LDC98S71`), consisting of 288 shows with approximately 8k speakers. In evaluation, both the BN testsets `dev03` & `eval03` as well as three Youtube (YTB) datasets `YTBElect`, `YTBGdev` and `YTBGeval` released by Google, were used. The utterances of all the testsets were processed by automatic segmentation and a brief summary of the resulted durations is illustrated in Table 1. Decoding was performed with the RT04 tri-gram language model [26].

Table 1. Summary of Evaluation Sets.

	BN		YTB		
	Dev03	Eval03	Elect	GDev	GEval
Total(hrs)	2.7	2.6	8.0	7.4	7.0
AvgUtt(secs)	10.7	10.9	7.9	6.2	6.9

4.1. Setup

The relevant GMMs, DNNs and our proposed models were trained or modified on an extended version of HTK Toolkit [27]. The 39-dimensional PLP+Δ+ΔΔ features processed by both global cepstral mean normalization (CMN) and cepstral variance normalization (CVN) were used to train a GMM-HMM model consisting of 6k tied triphone states on the maximum likelihood (ML) estimation. The features were then extended with the triples using HLDA to train a minimum-phone-error (MPE) model. This MPE model was used to obtain the state-level alignment of the training set.

For the SI DNN cross-entropy (CE) system, the input was the 468-dimensional PLP+Δ+ΔΔ+ΔΔΔ in a temporal context window of 9 frames. The neural network consisted of 5 hidden layers with 1000 nodes in each layer. The DNN parameters were initialized in a layer-wise discriminative pre-training fashion and subsequently optimized by back-propagation. 28 shows with about 600 speakers were used as the cross validation set. The well-tuned CE DNN was used to initialize the sequential DNN and this SI DNN-MPE system was then optimized on the MPE criterion.

Meanwhile, the parameters of the CE multi-basis system (denoted by *+mb*) were initialized by the well-trained SI DNN-CE model and then optimized in an interleaving mode as described in [23]. The multi-basis system was further tuned on the MPE criterion to obtain an MBANN-MPE model. In this paper, the number of basis were set to 2 and the interpolation weights of each training speaker were initialized as the *1-of-K* vector of i-vector clustering.

For the i-vectors, an SI UBM-GMM model with 2048 mixture components was initially trained on the training corpus. Speaker-dependent models were then estimated on the data belonging to each speaker to extract the speaker-level i-vectors. Focusing on rapid adaptation, each test utterance was treated as an independent entity and utterance-level i-vectors were extracted. The dimension was set to 30 and the i-vectors were globally normalized with zero mean and unit variance on the training set. The 30-dimensional i-vectors were then concatenated with the 468-dimensional SI DNN input features to train the i-vector DNN CE and MPE systems (*+iv*), following a similar procedure in the SI settings.

The CE multi-basis DNN with i-vector input features (*+iv+mb*) was initialized by the CE i-vector DNN and then updated in the MBANN training mode. It was further tuned with two sequential-trained epochs to obtain the MPE combined system. A support-vector regression model with linear kernels was trained using SVM-Light[1] over the training set to estimate the speaker-dependent interpolation weights for each of the MBANN systems. The predictor and the corresponding MBANN system were then used to evaluate the fast adaptation scheme (**-pred*). Moreover, the MBANN-CE fast predictive systems were respectively updated in the mode described in Section 3.2 for two iterations to obtain the refined predictive systems (**-pred-updt*).

4.2. Results

The CE decoding performance is reported in Table 2. On the matched BN testsets `dev03` and `eval03`, the extended MBANN with i-vector input features outperformed both of the primary i-vector and MBANN systems and gave up to 10% relative improvement contrasting to the SI baseline. Besides, the predictive schemes presented a performance similar to both the basic and i-vector-combined MBANNs. The comparison of the MBANN predictive schemes showed that the refined versions with interleaving updates can slightly enhance the initial predictive approach.

Table 2. CE Decoding Summary. (Word Error Rate [%])

System	BN		YTB		
	Dev03	Eval03	Elect	GDev	GEval
SI	12.5	10.9	33.8	58.5	62.1
+mb	11.9	10.3	33.8	56.9	61.2
+mb+pred	12.1	10.4	33.6	56.7	60.8
+mb+pred-updt	12.0	10.3	33.6	**56.1**	**60.5**
+iv	11.3	10.0	**32.8**	57.6	61.4
+iv+mb	**11.2**	**9.8**	33.3	57.1	61.3
+iv+mb+pred	**11.2**	**9.8**	33.5	58.0	61.7
+iv+mb+pred-updt	11.3	9.9	33.4	58.0	61.8

On the mismatched YTB data, the predictive system for the basic MBANN was able to acquire improvement compared to the second-pass decoding scheme on all the three testsets. Especially on `YTBGdev` and `YTBGeval`, the refined fast predictive module (*+mb+pred+updt*) reduced the word error rate by 0.8% and 0.7% (absolute values) respectively, compared with the standard hypothesis tuning method (*+mb*). This demonstrated the robustness of the proposed predictive scheme in high-error-rate scenarios.

The MBANN system with i-vector input features (*+iv+mb*) did not outperform the basic MBANN (*+mb*) on `YTBGdev` and `YTBGeval`. To explain this phenomenon, a comparison of the average euclidean distance between the test i-vectors and the mean of training ones on different evaluation sets is shown in Table 3. The BN testsets gave a similar average i-vector distance as the training set which pointed out their consistency spanning in the acoustic space. The BN test sets present a similar average i-vector distance as the training set which indicate a similar span of the BN test and training speaker spaces. The longer distances observed for the YTB i-vectors indicate the presence of i-vector estimations which are not or are not sufficiently represented by the training speaker space. This may explain why the i-vectors do not improve the baseline in the case of mismatched acoustic conditions. In addition, the mismatched i-vector inputs seem to incorrectly compensate the hidden representations among the bases of the multi-basis DNN system and degrade the performance of the combined system.

Table 3. Average I-vector Distance of Different Datasets.

Trn	BN		YTB		
	Dev03	Eval03	Elect	Gdev	Geval
5.4	5.9	5.8	9.0	10.0	9.4

The performance of the MPE models is summarized in Table 4. The multi-basis system combined with i-vector input still gives the lower WER under matched acoustic conditions (BN columns) and the MBANN system with i-vector predictor still achieved the best performance for `YTBGDev` and `YTBGEval`. The results on `YTBElect` are not very clear as before with SI system giving the lowest WER which is very close to the one achieved by the primary i-vector system.

Table 4. MPE Decoding Summary. (Word Error Rate [%])

System	BN		YTB		
	Dev03	Eval03	Elect	GDev	GEval
SI	11.2	10.2	**31.7**	55.5	59.2
+mb	10.7	9.5	32.5	55.4	60.3
+mb-pred	11.2	9.7	32.2	**54.0**	**58.6**
+iv	10.6	9.2	31.8	57.6	60.4
+iv+mb	**10.0**	**8.9**	32.1	55.4	60.3
+iv+mb-pred	10.3	9.0	32.1	56.1	59.6

Compared to the CE models, the MPE DNNs are more sensitive to the acoustic mismatches between training and test sets, because of the additional tuning epochs on the training set. This resulted in the degradation of the performance of the primary i-vector & multi-basis DNN systems on the YTB data. However, on the YTB sets, the multi-basis predictive approach was still able to correctly compensate the mismatches obtaining decoding improvement.

5. CONCLUSION

In this paper, we investigate the structured multi-basis adaptive neural network with i-vector representation for rapid adaptation in speech recognition. First, the i-vectors are appended to the input of a multi-basis DNN. Moreover, the i-vectors are used by a predictor to directly estimate the multi-basis transform, skipping the previously needed second decoding pass. The proposed approaches are evaluated on the utterance-level unsupervised adaptation of a large vocabulary continuous English broadcast news transcription task. The combination approach presents consistent gains in both the CE and MPE systems on the BN testsets which match the training acoustic conditions. Besides, under the highly-mismatched YTB conditions, the predictive approach of the multi-basis system outperforms the conventional second-pass decoding scheme.

[1] http://svmlight.joachims.org/

6. REFERENCES

[1] George E Dahl, Dong Yu, Li Deng, and Alex Acero, "Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 20, no. 1, pp. 30–42, 2012.

[2] Geoffrey Hinton et al., "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *Signal Processing Magazine, IEEE*, vol. 29, no. 6, pp. 82–97, 2012.

[3] Xie Chen, Adam Eversole, Gang Li, Dong Yu, and Frank Seide, "Pipelined back-propagation for context-dependent deep neural networks.," in *INTERSPEECH*, 2012.

[4] Yan Huang, Dong Yu, Chaojun Liu, and Yifan Gong, "A comparative analytic study on the gaussian mixture and context dependent deep neural network hidden markov models," *Interspeech*, 2014.

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

[6] George Saon, Hagen Soltau, David Nahamoo, and Michael Picheny, "Speaker adaptation of neural network acoustic models using i-vectors," in *Automatic Speech Recognition and Understanding (ASRU), 2013 IEEE Workshop on*. IEEE, 2013, pp. 55–59.

[7] Andrew Senior and Ignacio Lopez-Moreno, "Improving dnn speaker independence with i-vector inputs," in *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*. IEEE, 2014, pp. 225–229.

[8] Yulan Liu, Penny Karanasou, and Thomas Hain, "An investigation into speaker informed DNN front-end for LVCSR," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2015, South Brisbane, Queensland, Australia, April 19-24, 2015*, 2015, pp. 4300–4304.

[9] 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 *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*. IEEE, 2013, pp. 7942–7946.

[10] 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 *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*. IEEE, 2014, pp. 6339–6343.

[11] Xue Feng, Brigitte Richardson, Scott Amman, and James Glass, "On using heterogeneous data for vehicle-based speech recognition: A dnn-based approach," in *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*. IEEE, 2015.

[12] Penny Karanasou, Yongqiang Wang, Mark JF Gales, and Philip C Woodland, "Adaptation of deep neural network acoustic models using factorised i-vectors," in *Interspeech*, 2014.

[13] Penny Karanasou, Mark Gales, and Philip Woodland, "I-vector estimation using informative priors for adaptation of deep neural networks," ISCA, 2015.

[14] Frank Seide, Gang Li, Xie Chen, and Dong Yu, "Feature engineering in context-dependent deep neural networks for conversational speech transcription," in *Automatic Speech Recognition and Understanding (ASRU), 2011 IEEE Workshop on*. IEEE, 2011, pp. 24–29.

[15] 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.

[16] Bo Li and Khe Chai Sim, "Comparison of discriminative input and output transformations for speaker adaptation in the hybrid nn/hmm systems.," in *INTERSPEECH*, 2010, pp. 526–529.

[17] Yajie Miao, Hao Zhang, and Florian Metze, "Towards speaker adaptive training of deep neural network acoustic models," in *Proc. Interspeech*, 2014.

[18] Sabato Marco Siniscalchi, Jinyu Li, and Chin-Hui Lee, "Hermitian based hidden activation functions for adaptation of hybrid hmm/ann models.," in *INTERSPEECH*, 2012.

[19] Pawel Swietojanski and Steve Renals, "Learning hidden unit contributions for unsupervised speaker adaptation of neural network acoustic models," in *Spoken Language Technology Workshop (SLT), 2014 IEEE*. IEEE, 2014, pp. 171–176.

[20] Pawel Swietojanski and Steve Renals, "Differentiable pooling for unsupervised speaker adaptation," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2015, South Brisbane, Queensland, Australia, April 19-24, 2015*, 2015, pp. 4305–4309.

[21] Tian Tan, Yanmin Qian, Maofan Yin, Yimeng Zhuang, and Kai Yu, "Cluster adaptive training for deep neural network," in *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*. IEEE, 2015, pp. 4325–4329.

[22] Marc Delcroix, Keisuke Kinoshita, Takaaki Hori, and Tomohiro Nakatani, "Context adaptive deep neural networks for fast acoustic model adaptation," in *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*. IEEE, 2015, pp. 4535–4539.

[23] Chunyang Wu and Mark JF Gales, "Multi-basis adaptive neural network for rapid adaptation in speech recognition," in *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*. IEEE, 2015, pp. 4315–4319.

[24] Mark JF Gales, "Cluster adaptive training of hidden markov models," *Speech and Audio Processing, IEEE Transactions on*, vol. 8, no. 4, pp. 417–428, 2000.

[25] Douglas A Reynolds, Thomas F Quatieri, and Robert B Dunn, "Speaker verification using adapted gaussian mixture models," *Digital signal processing*, vol. 10, no. 1, pp. 19–41, 2000.

[26] SE Tranter, MJF Gales, R Sinha, S Umesh, and PC Woodland, "The development of the cambridge university rt-04 diarisation system," in *Proc. Fall 2004 Rich Transcription Workshop (RT-04)*, 2004.

[27] Steve Young, Gunnar Evermann, Mark Gales, Thomas Hain, Dan Kershaw, Xunying A Liu, Gareth Moore, Julian Odell, Dave Ollason, Dan Povey, Anton Ragni, Valtcho Valtchev, Phil Woodland, and Chao Zhang, "The HTK book (for HTK version 3.5)," 2015.