DNNs For Unsupervised Extraction of Pseudo FMLLR Features Without Explicit Adaptation Data

Neethu Mariam Joy, Murali Karthick Baskar, S. Umesh, Basil Abraham

Indian Institute of Technology-Madras, India

{ee11d009,ee13s010,umeshs,ee11d032}@ee.iitm.ac.in

Abstract

In this paper, we propose the use of deep neural networks (DNN) as a regression model to estimate feature-space maximum likelihood linear regression (FMLLR) features from unnormalized features. During training, the pair of unnormalized features as input and corresponding FMLLR features as target are provided and the network is optimized to reduce the mean-square error between output and target FMLLR features. During test, the unnormalized features are passed through this DNN feature extractor to obtain FMLLR-like features without any supervision or first pass decode. Further, the FMLLR-like features are generated frame-by-frame, requiring no explicit adaptation data to extract the features unlike in FMLLR or *i*-vector. Our proposed approach is therefore suitable for scenarios where there is little adaptation data. The proposed approach provides sizable improvements over basis-FMLLR and conventional FMLLR when normalization is done at utterance level on TIMIT and Switchboard-33hour data sets.

Index Terms: adaptation data, unsupervised speaker normalization, FMLLR, *i*-vector, basis-FMLLR, DNN

1. Introduction

Recently, deep neural networks (DNN) [1] have become the dominant paradigm in acoustic modeling. Although several studies have shown that DNNs are inherently invariant to speaker variations [2][3][4], speaker adaptation of DNN is being studied extensively [2][5][6][7] since they provide additional gains. Adapting the network weights [6] or just the biases [5] in the network using the speaker's adaptation data is one such approach. But this generally leads to overfitting since the number of parameters to be adapted is usually huge when compared to the amount of adaptation data. So these methods require some sort of regularization. Applying linear transformations to the input features (linear input networks [3][8][9]) or to activation of a hidden layer (linear hidden networks [10]) or input to the softmax layer (linear output networks [5][6][8]) are other approaches to network adaptation. Training the neural networks with speaker normalized features is an alternative to adapting the neural network. Features normalized via vocal tract length normalization (VTLN) and feature space maximum likelihood linear regression (FMLLR) [11][12] estimated with a Gaussian mixture model-hidden Markov model (GMM-HMM) can be used as input normalized features to DNN.

Allowing the neural network to learn speaker normalization by providing speaker-specific features along with unnormalized acoustic features as input during training, is yet another approach. Hence two sets of time-synchronous inputs are fed to the DNN, one for phonetic discrimination and another for speaker characterization. The use of *i*-vectors as the additional speaker-related feature for neural networks was proposed in [13] and further explored in [14][15][16][17]. George Saon *et al.* [13] concatenate *i*-vectors estimated per speaker to all the frames belonging to that speaker. Andrew Senior *et al.* [17] estimate a low dimensional utterance-level *i*-vector and append it to all the frames belonging to that utterance. Speaker codes proposed in [7][18] can also supplement speaker information.

In this paper, we focus on the special case of unsupervised speaker normalization with very little adaptation data. Our proposed method is useful for scenarios where only one (possibly short) test utterance is available for decoding and the test speaker's identity is also unknown. Hence each test utterance needs to be treated as though it is coming from different speakers and cannot be aggregated to do a speaker-level normalization. Estimating an utterance-wise FMLLR using such limited data will yield poor recognition results.

We propose using DNN to learn the feature normalizing transformations by providing the network with unnormalized train features as input and the corresponding FMLLR train features as the target. Once the DNN based pseudo-FMLLR extractor is trained, each unnormalized utterance can be fed into it to obtain corresponding pseudo-FMLLR feature. So the proposed method does unsupervised adaptation without imposing any constraint on the duration of the utterance. Also, the method neither requires any explicit adaptation data or first pass transcription during decoding. The pseudo-FMLLR features are extracted for both train and test from the DNN based FMLLR extractor. A conventional DNN model is then built using the train pseudo-FMLLR features and tested against the test pseudo-FMLLR features.

Our work is inspired from the speech enhanced DNNs proposed in [19][20][21] which used a DNN based signal preprocessing front-end to enhance speech by finding a mapping from noisy to clean speech signal. In a similar vein, the proposed pseudo-FMLLR feature generating DNN learns a mapping from unnormalized features to speaker normalized FMLLR features.

Another related work which estimates utterance-level FMLLR transforms is basis-FMLLR [22]. It is a basis representation of constrained MLLR transformation matrix, with variations among speakers concentrated in the leading coefficients. We compare the performance of our proposed method with test features obtained by using basis-FMLLR and utterance-wise FMLLR. In the case of the two existing FMLLR features, DNN was trained using speaker-wise FMLLR features while pseudo-FMLLR features were directly used for training with no additional speaker information. The proposed method gave significant performance improvements over both these techniques for TIMIT and 33-hour subset of Switchboard corpus, when utterance-level normalization is used.

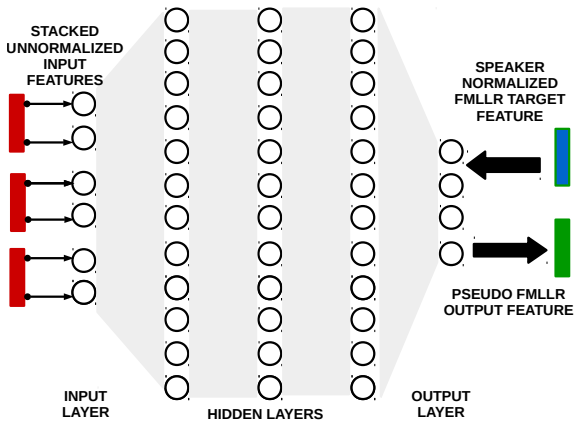

Figure 1: Proposed pseudo-FMLLR feature generation using DNN

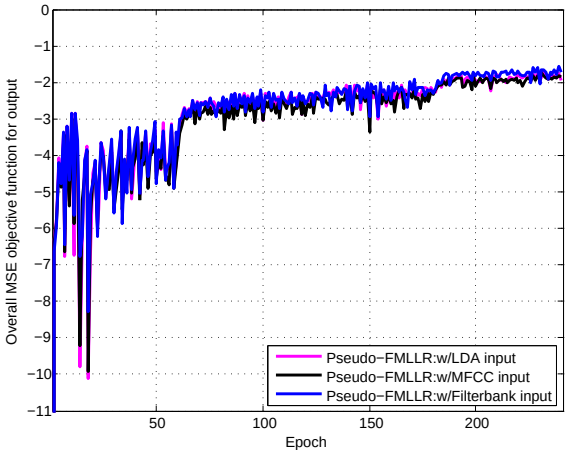

Figure 2: Change in MSE objective function of the DNN for extracting pseudo-FMLLR features for cross-validation data during each training epoch in Switchboard-33hour

The paper is organized as follows. Section 2 describes in detail the proposed method of generating pseudo-FMLLR features using DNN. The details of the experiments done on TIMIT and Switchboard-33hour corpus are given in section 3. The results of these experiments are presented and discussed in section 4. Finally, section 5 summarizes the findings and lists out the major contributions of the paper.

2. DNN-Based Extraction of Pseudo-FMLLR Features

Feature extraction using DNNs has been widely studied in the past. Bottleneck features were used as alternate inputs to conventional GMM-HMM based acoustic models. DNNs were also shown capable of estimating noise or speaker information from the utterance which can then be fed back to the network for the training algorithm to exploit the noise (noise-aware training or NaT) [23][24] or speaker (speaker-aware training or SaT) information.

In this paper, we propose a DNN based technique which generates speaker normalized features from unnormalized features. The proposed method relies on a DNN to learn a mapping for FMLLR feature normalization, by providing a time-synchronous pair of unnormalized feature as input and corresponding FMLLR feature as target from the train data. Using mean square error (MSE) between the target FMLLR feature and the pseudo-FMLLR feature generated by the network, DNN is optimized. Once this DNN is trained, pseudo-FMLLR features are generated for train and test utterances by feeding unnormalized features as input. These features are later used for DNN acoustic modeling.

Figure 1 shows the block schematic of a pseudo-FMLLR generating DNN. Over a context window of N frames (we take N=9, i.e., ± 4 context), the unnormalized features are stacked and fed as input to the DNN. The corresponding FMLLR feature of the middle frame in the unnormalized feature input is given as the target. This DNN is fully connected with rectified linear units (ReLU) as the activation function in the hidden layers. As the transformation from unnormalized feature to normalized FMLLR feature is linear, using ReLU to characterize this in the DNN rather than other non-linear activation functions like tanh or sigmoid is a logical choice. We have also found experimentally, that ReLU gave performance improvements compared to sigmoid and tanh.

Figure 2 shows the MSE between target FMLLR features and pseudo-FMLLR features for cross-validation data during DNN training of Switchboard-33hour for the three unnormalized input features considered. As the objective function comes closer to zero, it is clear that the DNN learns about the FMLLR normalization from the input unnormalized features.

3. Experimental Details

3.1. Analysis Of Different Unnormalized Input Features

Using 25ms frame-length and a frame-shift of 10ms, 13-dimensional Mel frequency cepstral coefficient (MFCC) feature vectors were extracted for each frame. Delta and acceleration coefficients were then augmented to these features to get a 39-dimensional feature vector. These were then mean and variance normalized per speaker for the train utterances. The 9 consecutive frames of MFCC were spliced together and projected down to 40-dimensional feature vector using linear discriminant analysis (LDA) and further diagonalized by maximum likelihood linear transformation (MLLT). These are referred to as LDA features. To get speaker normalized features, FMLLR transform was computed for each speaker in the train data on top of LDA features. Conventional FMLLR and basis-FMLLR features were tested by normalizing both at speaker-level and utterance-level. Additionally, 40-dimensional log Mel filter bank features were also extracted for every 25ms frames, shifted at 10ms interval. An extra 3-dimensional pitch information is augmented to it, forming 43-dimensional feature vector. These are referred to as unnormalized filter bank features.

3.2. Training DNN for Pseudo-FMLLR Feature Extraction

Three different types of unnormalized input features were considered: (a) 39-dimensional MFCC, (b) 40-dimensional LDA, (c) 43-dimensional filter bank features with pitch information. The target features were FMLLR features in all the three cases. Considering a temporal context of 9 frames for the input features, the input layer of the DNN had $D \times 9$ neurons, where $D = \{39, 40, 43\}$ for MFCC, LDA and filter bank features respectively. All the three DNNs had 6 hidden layers with 1024

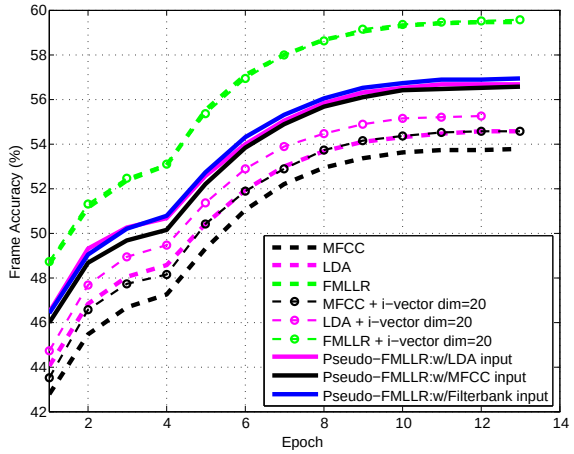

Figure 3: Frame Accuracy on cross-validation data for various DNNs in TIMIT

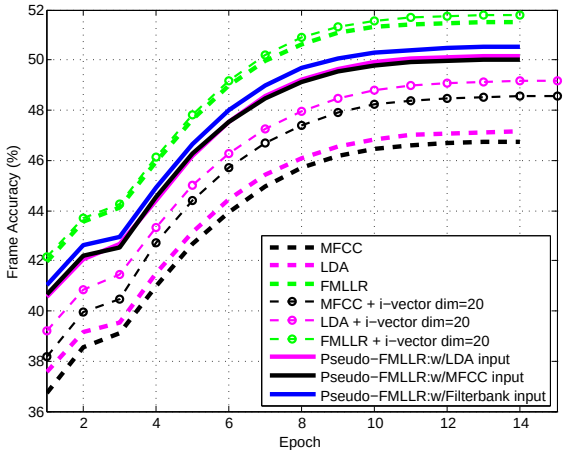

Figure 4: Frame Accuracy on cross-validation data for various DNNs in Switchboard-33hour

ReLU per layer. The output layer had 40 units representing the 40-dimensional FMLLR feature.

As the network needs to learn about speaker normalization pattern of FMLLR, speaker information is assumed to be available for the train utterances. The target speaker-specific FMLLR features were estimated from GMM-HMM. The DNN was optimized to reduce the MSE between these speaker normalized FMLLR features and the pseudo-FMLLR features generated by the network as shown in Figure 2. Once the network has completed training, pseudo-FMLLR features for train and test data were generated by passing the input features to the network.

3.3. Details Of Speech Corpus Used

The TIMIT continuous speech corpus [25] of read speech consists of broadband recordings of 630 speakers of eight major dialects of American English, each reading ten phonetically rich sentences [25]. Following the Kaldi recipe [26], 3696 speech files from 462 speakers were used for training and 400 utterances from 50 speakers was held out as the development set. The evaluation set has 192 utterances from 24 speakers. A bigram phoneme language model built from the train data was used in the decoding phase.

Switchboard–1 Release 2 telephone speech corpus [27] has 2400 two–sided telephone conversations from 543 speakers from all over the United States. The experiments were conducted on 33–hour (30K utterances) subset of the train data as mentioned in the Kaldi recipe. HUB5 English evaluation dataset [28] of conversational telephonic speech with 2.1 hours of audio and a development set with 5 hours of audio as mentioned in Kaldi recipe were used for testing. A 4-gram language model built from entire train data and Fisher English corpus [29] was used as the decoding language model.

3.4. Training DNN for Acoustic Modeling

Speaker independent and speaker normalized DNN acoustic models were trained, which differs only in the type of input provided. Speaker independent models used MFCC or LDA features as input. Conventional FMLLR or basis-FMLLR features estimated on per speaker basis were the inputs for speaker normalized DNN models. We used speaker-level normalization during train for these two features. In the case of pseudo-FMLLR, train features were extracted from the DNN feature extractor without using any explicit speaker information.

All DNN models shared the following same configuration: a context window of 11 frames applied on the input features, 6 hidden layers with tanh activation function and 2048 units per layer. The number of units in the output softmax layer corresponds to the number of context-dependent states in a phonetic decision tree with triphone context. The alignment information was taken from GMM-HMM model trained on conventional FMLLR features. The entire train data was first randomized at frame level prior to training. On minibatches of 250 frames, DNNs were trained with stochastic gradient descent approach and cross-entropy criterion. Layerwise discriminative pretraining was done prior to cross-entropy training.

Optionally, *i*-vectors were augmented to the input features for both speaker independent and speaker normalized DNN models. Following the recipe in [17], *i*-vectors were extracted on a per utterance basis and all the frames of a given utterance were augmented with the same M-dimensional *i*-vector. Both 20 and 40 dimensional *i*-vectors were extracted for all train and test utterances from a full-covariance GMM (128 and 512 mixture components in TIMIT and Switchboard-33hour respectively) built from 40-dimensional LDA features.

4. Results and Discussion

Figure 3 and 4 show the frame accuracy on cross-validation data for the afore mentioned various DNN acoustic models built for TIMIT and Switchboard-33hour respectively. It shows that the DNN trained on pseudo-FMLLR features is closer to the DNN trained on speaker-specific FMLLR than to those trained from unnormalized features like MFCC or LDA. This validates our claim that pseudo-FMLLR features, although learned from unnormalized features, are similar to the target FMLLR features. Even if additional speaker information is augmented to the unnormalized features via *i*-vector, the cross-validation frame accuracy of the resulting DNNs are not above that of the ones trained on pseudo-FMLLR features.

Tables 1 and 2 provides the phone error rate (PER) and word error rate (WER) of the various DNN acoustic models mentioned in section 3.4 for TIMIT and Switchboard-33hour respectively. We make the following observations:

- Of the three types of pseudo-FMLLR features, the one

Table 1: Phone Error Rate (%) of various DNN acoustic models for TIMIT

Feature Type	*Without i-Vectors*				*With i-Vectors of dim 20*				*With i-Vectors of dim 40*			
	Speaker-wise Normalization		*Utterance-wise Normalization*		*Speaker-wise Normalization*		*Utterance-wise Normalization*		*Speaker-wise Normalization*		*Utterance-wise Normalization*	
	Dev	*Eval*	*Dev*	*Eval*	*Dev*	*Eval*	*Dev*	*Eval*	*Dev*	*Eval*	*Dev*	*Eval*
MFCC	19.0	20.3	20.0	21.2	18.2	20.3	19.3	20.6	18.3	20.1	19.2	21.0
LDA	18.8	19.9	19.9	21.1	17.8	19.3	18.9	20.6	18.2	19.3	19.0	20.7
FMLLR	**17.4**	**18.8**	24.6	25.0	**17.1**	**18.6**	23.8	25.1	**16.9**	**19.0**	23.9	24.8
BASIS FMLLR	17.6	19.1	19.3	20.6	17.4	18.9	19.0	20.9	17.2	19.0	19.1	20.9
Pseudo FMLLR Features Generated By DNN												
LDA -> FMLLR	18.5	19.7	19.2	20.2	17.8	19.9	18.8	20.3	18.0	19.7	18.9	20.6
MFCC -> FMLLR	18.4	19.7	19.2	20.4	17.8	20.0	18.9	20.7	18.1	20.1	18.8	20.4
FBANK -> FMLLR	18.1	19.1	**18.8**	**19.8**	17.9	19.5	**18.5**	**20.2**	17.7	19.1	**18.5**	**20.0**

Table 2: Word Error Rate (%) of various DNN acoustic models for Switchboard 33-hour

Feature Type	*Without i-Vectors*				*With i-Vectors of dim 20*				*With i-Vectors of dim 40*			
	Speaker-wise Normalization		*Utterance-wise Normalization*		*Speaker-wise Normalization*		*Utterance-wise Normalization*		*Speaker-wise Normalization*		*Utterance-wise Normalization*	
	Dev	*Hub5*	*Dev*	*Hub5*	*Dev*	*Hub5*	*Dev*	*Hub5*	*Dev*	*Hub5*	*Dev*	*Hub5*
MFCC	28.18	21.90	29.87	22.70	26.33	21.40	27.87	22.60	26.56	21.90	27.81	22.90
LDA	27.96	21.50	29.75	**22.60**	26.31	21.50	**27.87**	**22.40**	26.53	21.90	**27.80**	**22.50**
FMLLR	**25.61**	**19.50**	35.50	29.20	**25.17**	**19.70**	33.82	28.00	**25.04**	**19.40**	33.18	28.10
BASIS FMLLR	25.79	19.50	30.33	23.80	25.35	19.70	29.37	23.60	25.16	19.50	29.05	23.70
Pseudo FMLLR Features Generated By DNN												
LDA -> FMLLR	27.30	21.80	29.13	23.10	26.44	21.60	28.31	22.80	26.34	21.50	28.16	23.00
MFCC -> FMLLR	27.29	21.90	29.10	22.70	26.47	21.40	28.28	22.60	26.32	21.30	28.12	22.70
FBANK -> FMLLR	26.77	22.10	**28.88**	23.10	26.01	21.90	28.48	23.10	25.96	21.60	28.18	22.90

*Hub5 results are for the Switchboard component in the data. We chose this to show the effect of adding i-vectors to the input features.

learned from filter bank gave the better acoustic model gains. This can also be validated from Figures 3 and 4, where the cross-validation frame accuracy of the DNN acoustic model trained with pseudo-FMLLR features generated from filter bank features is closer to conventional speaker normalized FMLLR DNN.

- Pseudo-FMLLR features outperform utterance-normalized conventional FMLLR and basis-FMLLR features by a significant margin in both TIMIT and Switchboard-33hour experiments. The performance of all the three pseudo-FMLLR features are highly superior to conventional FMLLR and basis-FMLLR features, with or without *i*-vectors.
- For TIMIT, the pseudo-FMLLR features give improved phone recognition accuracy over unnormalized features like MFCC and LDA for both development and evaluation data sets. This observation holds true for both speaker-level and utterance-level normalization. In the case of Switchboard-33hour, pseudo-FMLLR features gives performance improvement over the unnormalized features only in the development set for speaker-level and utterance-level normalization scenarios. But for the HUB5 evaluation data, pseudo-FMLLR gives only comparable performance to that of unnormalized features for both normalization cases.
- When *i*-vectors were augmented with unnormalized features, considerable performance gain was observed for both utterance-level and speaker-level normalization cases. For TIMIT, this improved performance of unnormalized features with i-vectors was still inferior to recognition accuracy of pseudo-FMLLR without i-vectors. This is in accordance with the observations made from Figure 3. In the case of Switchboard-33hours, both methods did not give improvement for HUB5 evaluation data.
- Appending *i*-vectors to pseudo-FMLLR, basis-FMLLR or conventional FMLLR provides only marginal gains in TIMIT for both utterance-level and speaker-level normalization cases. A similar pattern was also observed in speaker-level normalized case of Switchboard-33hour. But, the improvements were more pronounced for utterance-level normalization cases. Thus, we can infer that pseudo-FMLLR features already has speaker information embedded in it and supplying an additional source of speaker information via *i*-vectors was redundant. The marginal improvement obtained can be attributed to the ability of *i*-vectors to encode channel and background noise variations in addition to speaker variations[14][17].

5. Conclusion

In this paper, we have proposed an unsupervised speaker normalization technique based on DNN which requires no adaptation data. A time-synchronous input-target pair of unnormalized feature and speaker-normalized FMLLR feature is used to train a DNN based regression model. This DNN feature extractor generates pseudo-FMLLR features for both train and test utterances from unnormalized feature input. These features can then be used for acoustic modeling and was shown to give performance improvement over utterance-normalized conventional FMLLR and basis-FMLLR features.

6. References

[1] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-R. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath, and B. Kingsbury, "Deep Neural Networks for Acoustic Modeling in Speech Recognition," *IEEE Signal Processing Magazine*, pp. 82–97, November 2012.

[2] 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.

[3] V. Abrash, H. Franco, A. Sankar, and M. Cohen, "Connectionist Speaker Normalization and Adaptation," in *Proc. EUROSPEECH*, 1995.

[4] H. Liao, "Speaker Adaptation of Context Dependent Deep Neural Networks," in *Proc. ICASSP*, 2013, pp. 7947–7951.

[5] 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.

[6] D. Yu, K. Yao, H. Su, G. Li, and F. Seide, "KL-Divergence Regularized Deep Neural Network Adaptation for Improved Llarge Vocabulary Speech Recognition," in *Proc. ICASSP*, 2013, pp. 7893–7897.

[7] O. Abdel-Hamid and H. Jiang, "Fast Speaker Adaptation of Hybrid NN/HMM Model for Speech Recognition Based on Discriminative Learning of Speaker Code," in *Proc. ICASSP*, 2013, pp. 7942–7946.

[8] 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.

[9] J. Trmal, J. Zelinka, and L. Müller, "Adaptation of a Feedforward Artificial Neural Network Using a Linear Transform," in *Proc. Text, Speech and Dialogue*, 2010, pp. 423–430.

[10] R. Gemello, F. Mana, S. Scanzio, P. Laface, and R. de Mori, "Linear Hidden Transformations for Adaptation of Hybrid ANN/HMM Models," *Speech Communication*, vol. 49, no. 10-11, pp. 827–835, 2007.

[11] S. P. Rath, D. Povey, K. Vesley, and J. H. Cernocky, "Improved Feature Processing for Deep Neural Networks," in *Proc. INTERSPEECH*, 2013, pp. 109–113.

[12] S. H. K. Parthasarathi, B. Hoffmeister, S. Matsoukas, A. Mandal, N. Strom, and S. Garimella, "fMLLR Based Feature-Space Speaker Adaptation of DNN Acoustic Models," in *Proc. INTERSPEECH*, 2015.

[13] G. Saon, H. Soltau, D. Nahamoo, and M. A. Picheny, "Speaker Adaptation of Neural Network Acoustic Models using I-Vectors," in *Proc. ASRU*, 2013, pp. 55–59.

[14] M. Rouvier and B. Favre, "Speaker Adaptation of DNN-based ASR with I-Vectors: Does It Actually Adapt Models to Speakers?" in *Proc. INTERSPEECH*, 2014, pp. 3007–3011.

[15] V. Gupta, P. Kenny, P. Ouellet, and T. Stafylakis, "I-Vector Based Speaker Adaptation of Deep Neural Networks for French Broadcast Audio Transcription," in *Proc. ICASSP*, 2014, pp. 6334–6338.

[16] S. Garimella, A. Mandal, N. Strom, B. Hoffmeister, S. Matsoukas, and S. H. K. Parthasarathi, "Robust I-Vector based Adaptation of DNN Acoustic Model for Speech Recognition," in *Proc. INTERSPEECH*, 2015.

[17] A. Senior and I. Lopez-Moreno, "Improving DNN Speaker Independence with I-Vector Inputs," in *Proc. ICASSP*, 2014, pp. 225–229.

[18] S. Xue, O. Abdel-Hamid, H. Jiang, and L. Dai, "Direct Adaptation of Hybrid DNN/HMM Model for Fast Speaker Adaptation in LVCSR based on Speaker Code," in *Proc. ICASSP*, 2014, pp. 6339–6343.

[19] J. Du, Q. Wang, T. Gao, Y. Xu, L. Dai, and C. Lee, "Robust Speech Recognition with Speech Enhanced Deep Neural Networks," in *Proc. INTERSPEECH*, 2014, pp. 616–620.

[20] Y. Xu, J. Du, L. Dai, and C. Lee, "An Experimental Study on Speech Enhancement Based on Deep Neural Networks," *IEEE Signal Processing Letters*, vol. 21, no. 1, pp. 65–68, 2014.

[21] ——, "A Regression Approach to Speech Enhancement Based on Deep Neural Networks," *IEEE/ACM Trans. Audio, Speech & Language Processing*, vol. 23, no. 1, pp. 7–19, 2015.

[22] D. Povey and K. Yao, "A Basis Representation of Constrained MLLR Transforms for Robust Adaptation," *Computer Speech and Language*, vol. 26, no. 1, pp. 35–51, Jan 2012.

[23] M. L. Seltzer, D. Yu, and Y. Wang, "An Investigation of Deep Neural Networks for Noise Robust Speech Recognition," in *Proc. ICASSP*, 2013, pp. 7398–7402.

[24] D. Yu, M. L. Seltzer, J. Li, J. Huang, and F. Seide, "Feature Learning in Deep Neural Networks - A Study on Speech Recognition Tasks," in *Proc. ICLR*, 2013.

[25] J. Garofolo, L. Lamel, W. Fisher, J. Fiscus, D. Pallett, N. Dahlgren, and V. Zue, "TIMIT Acoustic-Phonetic Continuous Speech Corpus LDC93S1," *Linguistic Data Consortium*, 1993.

[26] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. K. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz, J. Silovsky, G. Stemmer, and K. Vesely, "The Kaldi Speech Recognition Toolkit," in *Proc. ASRU*, December 2011.

[27] J. Godfrey and E. Holliman, "Switchboard-1 Release 2 LDC97S62," *Linguistic Data Consortium*, 1993.

[28] L. D. Consortium, "2000 HUB5 English Evaluation Speech LDC2002S09," *Web Download. Philadelphia: Linguistic Data Consortium*, 2002.

[29] C. Cieri, D. Graff, O. Kimball, D. Miller, and K. Walker, "Fisher English Training Speech Part 1 Speech LDC2004S13," *Linguistic Data Consortium*, 2004.