INTERSPEECH 2017
August 20–24, 2017, Stockholm, Sweden

RNN-LDA Clustering for Feature Based DNN Adaptation

Xurong Xie[1,2], *Xunying Liu*[1,2], *Tan Lee*[2], *Lan Wang*[1,2]

[1]Key Laboratory of Human-Machine Intelligence-Synergy Systems,
Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen, China
[2]Chinese University of Hong Kong, Hong Kong, China

xrxie@ee.cuhk.edu.hk, xyliu@se.cuhk.edu.hk, tanlee@ee.cuhk.edu.hk, lan.wang@siat.ac.cn

Abstract

Model based deep neural network (DNN) adaptation approaches often require multi-pass decoding in test time. Input feature based DNN adaptation, for example, based on latent Dirichlet allocation (LDA) clustering, provide a more efficient alternative. In conventional LDA clustering, the transition and correlation between neighboring clusters is ignored. In order to address this issue, a recurrent neural network (RNN) based clustering scheme is proposed to learn both the standard LDA cluster labels and their natural correlation over time in this paper. In addition to directly using the resulting RNN-LDA as input features during DNN adaptation, a range of techniques were investigated to condition the DNN hidden layer parameters or activation outputs on the RNN-LDA features. On a DARPA Gale Mandarin Chinese broadcast speech transcription task, the proposed RNN-LDA cluster features adapted DNN system outperformed both the baseline un-adapted DNN system and conventional LDA features adapted DNN system by 8% relative on the most difficult Phoenix TV subset. Consistent improvements were also obtained after further combination with model based adaptation approaches.

Index Terms: deep neural network, adaptive training, recurrent neural network, latent Dirichlet allocation

1. Introduction

Recently, deep neural network (DNN) based acoustic modeling, such as DNN-HMM or tandem HMM [1, 2] models, have been widely used for large vocabulary continuous speech recognition (LVCSR) due to the strong representation ability of DNNs. However, these systems, especially systems for broadcast speech transcription, have to face with speech data from multiple sources with complex acoustic conditions, such as speakers, noises and bandwidths. Variation and mismatch from the acoustic conditions may lead to significant performance degradation on the LVCSR systems. To deal with this problem, acoustic condition adaptation approaches have been deeply investigated to improve the DNN based acoustic modeling performance.

Acoustic condition adaptation approaches can be divided into model based adaptation and feature based adaptation, according to the use of acoustic condition information. Model based adaptation approaches adapt models containing acoustic condition dependent parameters. These parameters can be linear transformation on the Gaussian mixture model (GMM) parameters like maximum likelihood linear regression (MLLR) transformation [3, 4, 5], weight vector for GMM mean or DNN transformation interpolation [6, 7, 8, 9], or scaling values on the DNN hidden outputs [10, 11]. The adaptation can be applied on evaluation data directly, or after the adaptive training [12] which jointly trains both speaker dependent and independent parameters on training data. Multi-pass decoding is often required to generate supervised labels for evaluation data adaptation.

Feature based adaptation approaches adaptively train acoustic models by employing acoustic condition information on the input features by such as fMLLR [13], i-vectors [14, 15, 16] speaker codes [17, 18, 19] and other appending features [20, 21], or on the output features by multi-task learning [22]. Most of the input feature based adaptation approaches do not require multi-pass decoding, thus the evaluation is more efficient than model based adaptation.

Latent Dirichlet allocation (LDA) clustering based DNN adaptation [23, 21] is one of this kind of input feature based adaptation approaches. During training, per-utterance clusters are discovered by a LDA model, and utilized as latent acoustic condition indicator features for DNN adaptive training. Hence, acoustic conditions are not necessary to provide explicitly, and the variation can be represented by distribution of these LDA clusters [21, 24] implicitly. However, LDA clustering normally treats the data as bags of discrete features, thus ignores the transition and correlation between neighboring clusters. This assumption initially made for document modeling may be not suitable for speech data.

In this paper, recurrent neural network (RNN) LDA clustering for feature based DNN acoustic model adaptive training is proposed. A RNN with a long short term memory (LSTM) and frame averaging layer was used to re-estimate the LDA clustering and their natural correlation over time. Subsequently, a range of feature based DNN adaptive training approaches, including feature concatenation, feature based hidden layer output scaling, and feature driven parameter interpolation condition on the RNN-LDA cluster posterior probabilities were investigated. On a DARPA Gale Mandarin Chinese broadcast speech transcription task, the proposed RNN-LDA cluster features adapted DNN systems outperformed both the baseline un-adapted DNN system and conventional LDA features adapted DNN system by character error rate (CER) reductions up to 8% relative on the most difficult Phoenix TV subset. After further combined with model based episode adaptation approaches, the best proposed DNN system outperformed the corresponding baseline DNN systems with and without episode adaptation by relative 5% and 12% respectively.

The rest of this paper is organized as follows. Conventional LDA clustering for feature based DNN acoustic modeling will be reviewed in section 2. Then, section 3 introduces the proposed RNN-LDA clustering. A range of feature based DNN adaptive training approaches will be investigated in section 4. In section 5 various DNN-HMM systems using RNN-LDA clus-

This work is supported by National Natural Science Foundation of China (NSFC 61135003, 91420301), ShenZhen Fundamental Research Program JCYJ20160429184226930, MSRA grant no. 6904412, CUHK grant no. 4055065 and no. 14227216.

http://dx.doi.org/10.21437/Interspeech.2017-368

tering information are evaluated on a DARPA Gale Mandarin Chinese broadcast speech transcription task. Section 6 draws the conclusion and future works.

2. LDA clustering for DNN adaptation

2.1. LDA based per-utterance clustering

Latent Dirichlet allocation (LDA) [25] is an unsupervised probabilistic generative model originally developed for modeling latent topics of documents. It assumes that each document is generated by a mixture of latent topics, and all mixtures follows a Dirichlet prior distribution. Therefore, documents can be clustered by their mixtures or distributions of latent topics. For acoustic speech data, each utterance can be regarded as a document, and each frame in the utterance can be regarded as a term in the document. The log likelihood of a LDA model can be described as

$$\log P(\mathcal{U}|\alpha) = \sum_{u=1}^{N} \log \int \prod_{t=1}^{T} \left(\sum_{k=1}^{K} P(\mathbf{o}_t^{(u)}|z_k) P(z_k|\theta) \right) P(\theta|\alpha) d\theta \quad (1)$$

where $\mathcal{U} = \{u : u = 1, 2, ..., U\}$ denotes the utterance set, $\mathbf{o}_t^{(u)}$ denotes the acoustic features for tth instant, z_k denotes the kth cluster or latent topic, and θ denotes the mixture probability of topics for each utterance.

Normally, $P(\theta|\alpha)$ is a Dirichlet distribution depending on "prior" α, $P(z_k|\theta)$ is a multinomial distribution, and $\mathbf{o}_t^{(u)}$ should be discrete and following a multinomial distribution depending on z_k. For continuous acoustic data, an efficient and convenient choice is to quantifies the acoustic features into a discrete space by, for example, GMM based LBG vector quantification [26, 21]. For inference, cluster posterior probabilities $P_{\text{LDA}}(z_k|\mathbf{o}_t^{(u)})$ given each frame are first computed, for example, using variational EM [25] algorithm, by

$$P_{\text{LDA}}(z_k|\mathbf{o}_t^{(u)}) \propto P(\mathbf{o}_t^{(u)}|z_k) \exp\left\{ \Psi\left(P_{\text{LDA}}^{\text{old}}(z_k|\mathbf{o}_t^{(u)}) + \alpha_k \right) \right\}. \quad (2)$$

Then, the per-utterance cluster posterior probabilities can be obtained by averaging them, that is

$$P_{\text{LDA}}(z_k|u) = \frac{1}{T + \sum_{k'=1}^{K} \alpha_{k'}} \left(\sum_{t=1}^{T} P_{\text{LDA}}(z_k|\mathbf{o}_t^{(u)}) + \alpha_k \right) \quad (3)$$

which can be used to represent the latent acoustic conditions of speech from different domains [24, 21].

2.2. DNN adaptive training using LDA clusters

DNNs indicate neural networks consisting of many feed-forward or recurrent layers. For acoustic modeling, DNNs can be used in DNN-HMM architectures to model the posterior probabilities of HMM states given the acoustic features [27], or as feature extractor to generate robust auxiliary features for complement to acoustic features [2, 28]. A DNN can be seen as a non-linear transformation of the input features, thus has strong representation ability to variations captured in input features.

By using LDA clustering for feature based DNN adaptation, none of additional acoustic condition information and multi-pass decoding is required. The cluster label k with the maximized posterior probability $P_{\text{LDA}}(z_k|u)$ can be utilized [23, 21] as latent acoustic condition of utterance u and form a 1-of-K vector. This vector will be concatenated with every frame acoustic feature vector in the same utterance, and the concatenated feature vector will then be used as input to train the DNN.

3. RNN-LDA clustering using LSTM layer

For inference of LDA, each utterance is treated as a bag of discrete data, thus $P_{\text{LDA}}(z_k|\mathbf{o}_t^{(u)})$ is computed by ignoring the transition and correlation between neighbors. Although HMM can be combined with LDA to deal with temporal data [29], only the information from a very short history can be considered for each instant. Moreover, the quantification on acoustic features may also introduce new error. Another approach to deal with the continuous data may be using the GMMs to model $P(\mathbf{o}_t^{(u)}|z_k)$. However, this requires multi-level of iterations for training and inference, which may lead to high complexity of time. To address the issues, recurrent neural network (RNN) is used to re-estimate the clustering of LDA, which is called RNN-LDA.

A RNN is a neural network with recurrent layers, which can model the transition and correlation of continuous data. Inference of every frame on RNN will consider all information propagating from previous frames. However, considering long history on conventional RNNs may lead to ill convergence [30]. Long short term memory (LSTM) [31] recurrent layers overcome this weakness by employing input gates, output gates, forget gates and memory cells to preserve only the useful information from history.

Similar to the LDA inference, training of RNN-LDA consists of two stages. The first stage (left hand side of figure 1) exploits the per-utterance LDA clusters as supervised labels to train the every frame RNN posterior probabilities by

$$P_{\text{RNN}}(z_k|\mathbf{o}_{1:t}^{(u)}) = \text{Softmax}_k\left(\mathbf{W}\mathbf{h}_{\text{LSTM}}(\mathbf{o}_{1:t}^{(u)}) + \mathbf{b} \right) \quad (4)$$

where $\mathbf{W}$ and $\mathbf{b}$ denote weight matrix and bias vector of feed-forward layer, and $\mathbf{h}_{\text{LSTM}}(\mathbf{o}_{1:t}^{(u)})$ denotes the LSTM layer output on tth frame given all its history in utterance u. This inference on each frame makes use of the LDA cluster transition and correlation from their history. The second stage (right hand side of figure 1) trains and obtains the estimation of per-utterance cluster posterior probabilities by frame averaging on the above LSTM layer outputs, that is

$$P_{\text{RNN-LDA}}(z_k|u) = \text{Softmax}_k\left(\mathbf{W} \frac{1}{T} \sum_{t=1}^{T} \left(\mathbf{h}_{\text{LSTM}}(\mathbf{o}_{1:t}^{(u)}) \right) + \mathbf{b} \right). \quad (5)$$

This stage is essential for the RNN-LDA training, as the inference error on the earlier frame may propagate to all later frames in the first stage. However, directly training RNN-LDA from the second stage is easy to be stuck in local minimum and thus difficult to converge. Therefore, these two stages can be viewed as pre-training and fine-tuning of RNN-LDA respectively.

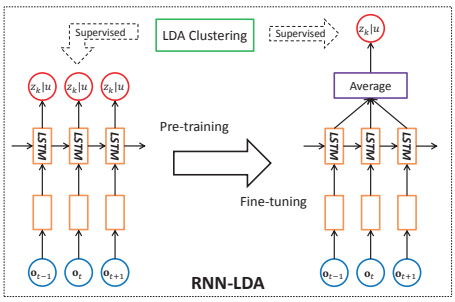

Figure 1: *An example of RNN-LDA.*

4. Feature based DNN adaptive training

4.1. Feature concatenation

For feature based DNN adaptive training, the RNN-LDA or LDA cluster labels can be used as appending features as mentioned in section 2.2. However, [24, 21] showed that acoustic condition variations can be represented by distribution of these clusters. Hence, using the cluster log posterior probabilities

$$\mathbf{f}^{(u)} = [\log P(z_1|u), \log P(z_2|u), ..., \log P(z_K|u)]^T \quad (6)$$

as appending features should be more appropriate than the cluster labels. As shown in the left part of figure 2, $\mathbf{f}^{(u)}$ is concatenated with each frame acoustic feature vector in utterance u to form the DNN input feature vector

$$\mathbf{o}_t'^{(u)} = [\mathbf{o}_{t-\tau}^{(u)}, ..., \mathbf{o}_t^{(u)}, ..., \mathbf{o}_{t+\tau}^{(u)}, \mathbf{f}^{(u)}]^T \quad (7)$$

where τ denotes the context window width for DNN inputs. Using the concatenated vector as input, DNN can be trained by discriminative pre-training followed by global fine-tuning. This method can also be treated as transfer learning [32] from unsupervised knowledge discovered by LDA and RNN-LDA.

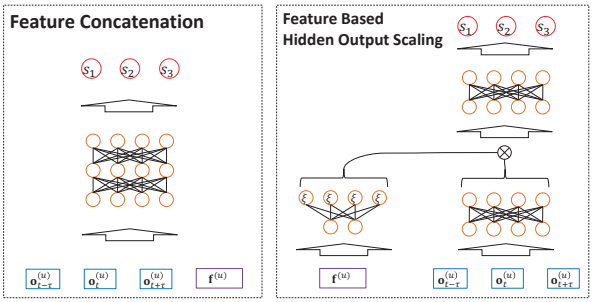

Figure 2: *Examples of feature concatenation (left) and feature based hidden output scaling (right).*

4.2. Feature based hidden output scaling

Scaling values on the hidden layer activation outputs can be used as acoustic condition dependent parameters for DNN adaptive training or adaptation. Borrowing from this idea, RNN-LDA or LDA clustering features $\mathbf{f}^{(u)}$ can be used to generate the scaling values for each utterance by using a feed-forward layer, which is shown in the right part of figure 2. Scaling values here can be located between zero and two by using a sigmoid function like the learning hidden unit contribution (LHUC) [10] adaptation, that is

$$\xi^{(u)} = 2\text{Sigmoid}\left(\mathbf{W}_{\text{LHUC}}\mathbf{f}^{(u)} + \mathbf{b}_{\text{LHUC}}\right). \quad (8)$$

Alternatively, the scaling values can also be linear as in [11], but this will not be investigated in the paper. Subsequently, the lth hidden layer activation outputs are computed by

$$\mathbf{h}_{l,t}^{(u)} = \xi^{(u)} \otimes \text{Sigmoid}\left(\mathbf{W}_l\mathbf{h}_{l-1,t}^{(u)} + \mathbf{b}_l\right) \quad (9)$$

where $\otimes$ denotes element-wise product.

4.3. Feature driven cluster adaptive training

Feature driven clustering adaptive training (CAT) interpolates DNN parameters by weight vector generated from RNN-LDA or LDA clustering feature $\mathbf{f}^{(u)}$ using a feed-forward layer, which is shown in figure 3. The layer activation should be softmax function to ensure the weights summing to one, that is

$$\lambda^{(u)} = \text{Softmax}(\mathbf{W}_{\text{CAT}}\mathbf{f}^{(u)} + \mathbf{b}_{\text{CAT}}) \quad (10)$$

where $\lambda^{(u)} \in \mathbb{R}^N$ is the weight vector for N interpolation bases. For initialization of this layer, parameters are pre-trained with supervised labels clustered on $\mathbf{f}^{(u)}$ by a GMM with N mixtures. Finally, similar to the model based CAT-DNNs, bases interpolation can occur, for example, on the weight matrix [9], before the non-linear activation [9], or following the non-linear activation outputs [8], which are computed respectively by

$$\left\{\mathbf{h}_{l,t}^{(u)}\right\}^{(\text{wgt-mat})} = \text{Sigmoid}\left(\sum_{n=1}^{N}\lambda_n^{(u)}\mathbf{W}_{l,n}\mathbf{h}_{l-1,t}^{(u)} + \mathbf{b}_l\right) \quad (11)$$

$$\left\{\mathbf{h}_{l,t}^{(u)}\right\}^{(\text{pre-act})} = \text{Sigmoid}\left(\sum_{n=1}^{N}\lambda_n^{(u)}\left(\mathbf{W}_{l,n}\mathbf{h}_{l-1,t}^{(u)} + \mathbf{b}_{l,n}\right)\right) \quad (12)$$

$$\left\{\mathbf{h}_{l,t}^{(u)}\right\}^{(\text{post-act})} = \sum_{n=1}^{N}\lambda_n^{(u)}\text{Sigmoid}\left(\mathbf{W}_{l,n}\mathbf{h}_{l-1,t}^{(u)} + \mathbf{b}_{l,n}\right). \quad (13)$$

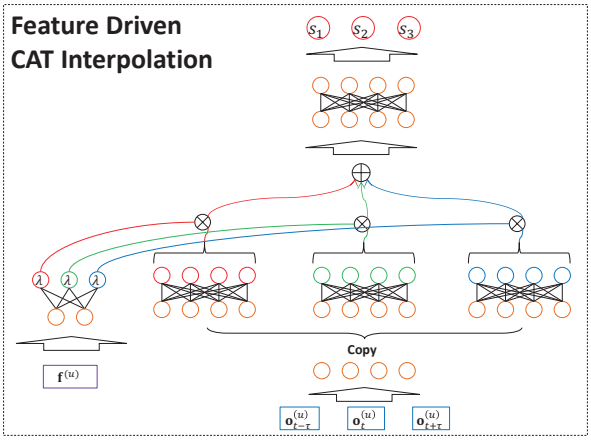

Figure 3: *An example of feature driven CAT interpolation.*

4.4. Combining feature and model based adaptation

Variation information captured by RNN-LDA clustering may not be considered by certain acoustic condition model based adaptation. Hence, RNN-LDA clustering feature based DNN adaptation can be used as a complement to model based adaptation, if multi-pass decoding is available. For example, model based episode adaptation with LHUC scaling [10] can be employed on the DNN adaptively trained with concatenation features $\mathbf{o}_t'^{(u)}$ mentioned in section 4.1.

5. Experiments

For experiments, the DARPA Gale Mandarin Chinese broadcast speech data was used to train and evaluate the proposed systems performance. The data set contained one 202.5 hours training set with 29 shows, 506 episodes and 150k utterances, one 07 development set and one 07 evaluation set. In the experiments, for more reliable evaluation, the 07 development and evaluation sets were merged to form a 4.7 hours evaluation set with 26 shows, 157 episodes and 3170 utterances. Moreover, speech

collected from the 10% most recent dates of each show in the training set was held out as the development set. The evaluation set could be divided into three classes: the official TV, NTD TV and Phoenix TV subsets. Among these subsets, speech in the official TV set was mainly spoken in standard Mandarin; the NTD TV shows were out-domain data that did not appear in the training set; the Phoenix TV is the most difficult part consisting of large amount of conversations and spontaneous speech.

For acoustic modeling, a range of DNN-HMM systems were built on 11 successive frames of 42 dimensional linear discriminative transformed vector with PLP features and pitch. The supervised labels were 9205 tri-phone states aligned using a GMM-HMM system. The DNNs consisted of six hidden layers with 2048 nodes and sigmoid activation. For DNN training, the scheme with discriminative pre-training followed by global fine-tuning using error back-propagation was utilized. All systems were trained and evaluated using modified versions of HTK [33] and Kaldi toolkit [34].

The LDA clustering was implemented by variational EM algorithm following the GMM based LBG vector quantification with 512 mixtures. The cluster number was selected using method in [26] and equal to 64. The RNN-LDA model consisted of three feed-forward layers with 2048 nodes in the first and 512 nodes in the rest layers, followed by one LSTM layer with 128 cells. After trained, the 64 dimensional log probability vector was used for the feature based DNN adaptation systems.

Table 1 shows performance of the baseline DNN systems with and without LHUC episode adaptation [10] on test set, and LDA features adapted DNN systems using conventional cluster labels (sec 2.2) [21] or log probabilities (sec 4.1). It shows that cluster log probability vector (Sys (4)) obtained better performance than cluster labels (Sys (3)). Moreover, the LDA features adapted DNN systems (Sys (3) and (4)) performed worse than adaptation systems using LHUC episode adaptation (Sys (2)).

Table 1: *Performance of baseline DNN systems with and without episode adaptation, and LDA features adapted DNN systems using cluster labels or log probabilities.*
(∗ LHUC*: LHUC episode adaptation on test set.)*

Sys	Adaptation		CER (%)			
	Feat	Mod	Offi.	NTD.	PHNX.	Avg.
(1)	×	×	4.1	8.6	14.5	8.1
(2)	×	LHUC [10]	**3.8**	**8.3**	14.1	**7.8**
(3)	$\text{LDA}^{\text{label}}$ conc (sec 2.2) [21]	×	4.2	8.8	14.4	8.2
(4)	LDA^{prob} conc (sec 4.1)		4.2	8.7	**13.8**	7.9

The second experiment shown in table 2 compared different approaches of feature based DNN adaptive training using the LDA-RNN or LDA clustering, including feature concatenation (sec 4.1), feature based LHUC scaling (sec 4.2), and feature driven CAT interpolation (sec 4.3) with 3 bases. On the Phoenix TV subset, the adapted DNN system with RNN-LDA cluster log probability feature concatenation (Sys (3)) performed the best, which outperformed the un-adapted baseline DNN system (Sys (1)), conventional LDA cluster label features adapted DNN system (Sys (3) in table 1) and LDA cluster log probability features adapted DNN system (Sys (2)) by relative CER reductions of 8%, 8% and 3% respectively. The comparison between system (3) and (4) showed that using adaptation information on lower layers performed better than on higher layers. Moreover, the feature based LHUC scaling approach (Sys (5)) performed comparably to the feature concatenation approach (Sys (3)), and achieved the best performance on the out-domain NTD TV set. Both of these two approaches outperformed the feature driven CAT interpolation approaches (Sys (6), (7), (8)).

Table 2: *Performance of feature based adapted DNN systems.*
(∗ f-LHUC*: Feature based LHUC scaling.)*
(∗ f-CAT*: Feature driven CAT interpolation.)*

Sys	Adapt	Adapting Training		CER (%)			
	Feat	Method	Layer	Offi.	NTD.	PHNX.	Avg.
(1)	×	×	×	4.1	8.6	14.5	8.1
(2)	LDA^{prob}	Conc	Input	4.2	8.7	13.8	7.9
(3)	RNN-LDA^{prob}	Conc (sec 4.1)	Input	**3.9**	8.6	**13.4**	**7.7**
(4)			$\text{Hidden}^{\text{2nd}}$	4.2	8.9	14.3	8.1
(5)		f-LHUC (sec 4.2)	$\text{Hidden}^{\text{1st}}$	**3.9**	**8.2**	13.7	**7.7**
(6)		$\text{f-CAT}^{\text{post-act}}$ (sec 4.3)		4.1	9.1	14.4	8.1
(7)		$\text{f-CAT}^{\text{pre-act}}$ (sec 4.3)		4.1	8.7	14.2	8.0
(8)		$\text{f-CAT}^{\text{wgt-mat}}$ (sec 4.3)		4.1	8.6	14.2	8.0

The final experiment evaluated the performance of LDA-RNN features adapted DNN system combining with model based LHUC episode adaptation. Here, all feature based DNN adaptation systems used the approach of RNN-LDA cluster log probability feature concatenation. Table 3 shows that, on the most difficult Phoenix TV set, the RNN-LDA features adapted DNN system (Sys (4)) performed better than the baseline system with LHUC episode adaptation (Sys (2)), and comparably to that system with LHUC episode adaptive training (Sys 3). However, on the out-domain NTD TV set, the episode adapted systems performed better, which might be caused by the use of multi-pass decoding. Moreover, after further combination with LHUC episode adaptation, consistent improvements on all evaluation subsets were obtained by the RNN-LDA features adapted DNN systems (Sys (5), (6)). This implied that RNN-LDA feature based adaptation could be used as a complement to episode adaptation. Finally, the best proposed system (Sys (6)) outperformed the corresponding adapted and un-adapted baseline systems (Sys (3), (1)) by relative 5% and 12% respectively.

Table 3: *Performance of baseline and RNN-LDA features adapted DNN systems with and without episode adaptation.*
(∗ + adaptive training*: LHUC episode adaptive training.)*

Sys	Adaptation		CER (%)			
	Feat	Mod	Offi.	NTD.	PHNX.	Avg.
(1)	×	×	4.1	8.6	14.5	8.1
(2)		LHUC [10]	3.8	8.3	14.1	7.8
(3)		+ adaptive training	3.8	8.0	13.4	7.5
(4)	RNN-LDA^{prob} conc	×	3.9	8.6	13.4	7.7
(5)		LHUC [10]	**3.6**	7.9	13.1	7.3
(6)		+ adaptive training	**3.6**	**7.7**	**12.6**	**7.1**

6. Conclusions

This paper proposed RNN-LDA clustering taking LDA cluster transition and correlation into consideration. Moreover, a range of feature based DNN adaptive training approaches were investigated. On a DARPA Gale Mandarin Chinese broadcast speech transcription task, the best proposed RNN-LDA cluster features adapted DNN system outperformed both the baseline un-adapted DNN system and conventional LDA features adapted DNN system by 8% relative on the most difficult Phoenix TV subset. After combining with episode adaptation, consistent overall improvements of 5% and 12% relative were obtained against the corresponding adapted and un-adapted baseline systems respectively. Future works will focus on considering position contribution and longer temporality in RNN-LDA.

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

[2] H. Hermansky, D. P. Ellis, and S. Sharma, "Tandem connectionist feature extraction for conventional hmm systems," in *Acoustics, Speech, and Signal Processing, 2000. ICASSP'00. Proceedings. 2000 IEEE International Conference on*, vol. 3. IEEE, 2000, pp. 1635–1638.

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

[4] V. V. Digalakis, D. Rtischev, and L. G. Neumeyer, "Speaker adaptation using constrained estimation of gaussian mixtures," *IEEE Transactions on speech and Audio Processing*, vol. 3, no. 5, pp. 357–366, 1995.

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

[6] ——, "Cluster adaptive training for speech recognition." in *ICSLP*, vol. 1998, 1998, pp. 1783–1786.

[7] ——, "Cluster adaptive training of hidden markov models," *IEEE transactions on speech and audio processing*, vol. 8, no. 4, pp. 417–428, 2000.

[8] C. Wu and M. J. 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.

[9] T. Tan, Y. Qian, and K. Yu, "Cluster adaptive training for deep neural network based acoustic model," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 3, pp. 459–468, 2016.

[10] P. Swietojanski and S. 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.

[11] C. Zhang and P. C. Woodland, "Dnn speaker adaptation using parameterised sigmoid and relu hidden activation functions," in *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*. IEEE, 2016, pp. 5300–5304.

[12] T. Anastasakos, J. McDonough, R. Schwartz, and J. Makhoul, "A compact model for speaker-adaptive training," in *Spoken Language, 1996. ICSLP 96. Proceedings., Fourth International Conference on*, vol. 2. IEEE, 1996, pp. 1137–1140.

[13] B. Li and K. C. Sim, "Comparison of discriminative input and output transformations for speaker adaptation in the hybrid nn/hmm systems," 2010.

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

[15] A. Senior and I. 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.

[16] V. Gupta, P. Kenny, P. Ouellet, and T. Stafylakis, "I-vector-based speaker adaptation of deep neural networks for french broadcast audio transcription," in *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*. IEEE, 2014, pp. 6334–6338.

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

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

[19] S. Xue, O. Abdel-Hamid, H. Jiang, L. Dai, and Q. Liu, "Fast adaptation of deep neural network based on discriminant codes for speech recognition," *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, vol. 22, no. 12, pp. 1713–1725, 2014.

[20] M. L. Seltzer, D. Yu, and Y. Wang, "An investigation of deep neural networks for noise robust speech recognition," in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*. IEEE, 2013, pp. 7398–7402.

[21] M. Doulaty, O. Saz, R. W. Ng, and T. Hain, "Latent dirichlet allocation based organisation of broadcast media archives for deep neural network adaptation," in *Automatic Speech Recognition and Understanding (ASRU), 2015 IEEE Workshop on*. IEEE, 2015, pp. 130–136.

[22] Y. Qian, T. Tan, and D. Yu, "Neural network based multi-factor aware joint training for robust speech recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 12, pp. 2231–2240, 2016.

[23] O. Saz, M. Doulaty, S. Deena, R. Milner, R. W. Ng, M. Hasan, Y. Liu, and T. Hain, "The 2015 sheffield system for transcription of multi-genre broadcast media," in *Automatic Speech Recognition and Understanding (ASRU), 2015 IEEE Workshop on*. IEEE, 2015, pp. 624–631.

[24] M. Doulaty, O. Saz, R. W. Ng, and T. Hain, "Automatic genre and show identification of broadcast media," *arXiv preprint arXiv:1606.03333*, 2016.

[25] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *Journal of machine Learning research*, vol. 3, no. Jan, pp. 993–1022, 2003.

[26] M. Doulaty, O. Saz, and T. Hain, "Unsupervised domain discovery using latent dirichlet allocation for acoustic modelling in speech recognition," *arXiv preprint arXiv:1509.02412*, 2015.

[27] H. A. Bourlard and N. Morgan, *Connectionist speech recognition: a hybrid approach*. Springer Science & Business Media, 2012, vol. 247.

[28] X. Xie, R. Su, X. Liu, and L. Wang, "Deep neural network bottleneck features for generalized variable parameter hmms." in *INTERSPEECH*, 2014, pp. 2739–2743.

[29] E. B. Fox, E. B. Sudderth, M. I. Jordan, and A. S. Willsky, "A sticky hdp-hmm with application to speaker diarization," *The Annals of Applied Statistics*, pp. 1020–1056, 2011.

[30] Y. Bengio, P. Simard, and P. Frasconi, "Learning long-term dependencies with gradient descent is difficult," *IEEE transactions on neural networks*, vol. 5, no. 2, pp. 157–166, 1994.

[31] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[32] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on knowledge and data engineering*, vol. 22, no. 10, pp. 1345–1359, 2010.

[33] S. Young, G. Evermann, M. Gales, T. Hain, D. Kershaw, X. Liu, G. Moore, J. Odell, D. Ollason, D. Povey *et al.*, "The htk book," *Cambridge university engineering department*, vol. 3, p. 175, 2002.

[34] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz *et al.*, "The kaldi speech recognition toolkit," in *IEEE 2011 workshop on automatic speech recognition and understanding*, no. EPFL-CONF-192584. IEEE Signal Processing Society, 2011.