NOISE ADAPTIVE FRONT-END NORMALIZATION BASED ON VECTOR TAYLOR SERIES FOR DEEP NEURAL NETWORKS IN ROBUST SPEECH RECOGNITION

Bo, Li and Khe Chai, Sim

National University of Singapore, School of Computing
Computing 1, Singapore 117417

ABSTRACT

Deep Neural Networks (DNNs) have been successfully applied to various speech tasks during recent years. In this paper, we investigate the use of DNNs for noise-robust speech recognition and demonstrate their superior capabilities of modeling acoustic variations over the conventional Gaussian Mixture Models (GMMs). We then propose to compensate the normalization front-end of the DNNs using the GMM-based Vector Taylor Series (VTS) model compensation technique, which has been successfully applied in the GMM-based ASR systems to handle noisy speech. To fully benefit from both the powerful modeling capability of the DNN and the effective noise compensation of the VTS, an adaptive training algorithm is further developed. The preliminary experimental results on the AURORA 2 task have demonstrated the effectiveness of our approach. The adaptively trained system has been shown to outperform the GMM-based VTS adaptive training by relatively 18.8% using the MFCC features and 21.9% using the FBank features.

Index Terms— Noise Robustness, Vector Taylor Series, Deep Neural Networks

1. INTRODUCTION

Creating and developing systems that would be much more robust against variability and shifts in acoustic environments, reverberations, external noise sources, communication channels, speaker characteristics and language characteristics has always been the goal of speech recognition researchers. In recent years, Deep Neural Networks (DNNs), which are effectively multilayer perceptrons (MLPs) with many hidden layers, have been successfully applied to various speech tasks. The context-independent DNN-HMM hybrid systems [1, 2] have been initially proposed for the phoneme recognition. Later, a novel context-dependent (CD)-DNN-HMM system [3] has been successfully applied to large vocabulary speech recognition systems. The DNN system has been shown to reduce the word error rate by up to one third on the challenging conversational speech transcription tasks compared to the discriminatively trained conventional CD-GMM-HMM systems in [4]. It then intrigues much interest in adopting the DNNs for the noise-robust speech recognition. In [5], the Recurrent Neural Network (RNN) and the DNN have been shown to generalize much better than GMMs and MLPs on the AURORA 2 task [6]. In [7], a deep recurrent denoising autoencoder (DRADE) is trained on the stereo data to reconstruct the clean utterances from the noisy input features. It has been shown to outperform the SPLICE denoising algorithm [8] and the hand-engineered ETSI2 advanced front end (AFE) denoising system [9]. The DRADE makes no assumption on how the noise affects the signal, nor the existence of distinct noise environments. It is thus more dependent upon the training data to provide a reasonable sample of noise environments that could be possibly encountered at test time.

Model-based approaches that utilize explicit models of noise, channel distortion, and their interaction with speech are a well-established and continually-evolving research paradigm in noise-robust speech recognition. The Vector Taylor Series (VTS) compensation method [10] and the corresponding noise adaptive training (NAT) [11] have been widely adopted in GMM-HMM systems. Due to the many layers of non-linearities, the model-based compensation for the DNN is much harder than for the GMM. In [12], a Factorial Hidden Restricted Boltzmann Machine (FHRBM) is proposed to explicitly model the noise distribution and how the noise affects the speech. However, due to the un-observed noise parameters in the input layer of FHRBM, the inference is intractable and scaling exponentially with the number of hidden units. Variational approximations have to be used. In this paper, we propose to compensate only the normalization front-end of the DNN using the VTS.

2. METHODS

The effect of the acoustic noise on the feature vectors is usually modeled by the following mismatch function [13]

$$\begin{aligned}\boldsymbol{y}^s &= \boldsymbol{x}^s + \boldsymbol{h}^s + \boldsymbol{C}\log(\mathbf{1} + \exp(\boldsymbol{C}^\dagger(\boldsymbol{n}^s - \boldsymbol{x}^s - \boldsymbol{h}^s))) \\ &= \boldsymbol{x}^s + \boldsymbol{h}^s + \mathrm{g}(\boldsymbol{x}^s, \boldsymbol{n}^s, \boldsymbol{h}^s) \end{aligned} \quad (1)$$

where the superscript s indicating the static part of each variable. The $\boldsymbol{y}, \boldsymbol{x}, \boldsymbol{h}, \boldsymbol{n}$ are the cepstrum vectors corresponding to the distorted speech, clean speech, channel and additive

noise, respectively. $\boldsymbol{C}$ and $\boldsymbol{C}^{\dagger}$ are the discrete cosine transform and its pseudo-inverse.

The model compensation scheme combines the clean trained model and noise distributions with the mismatch function to find the parameters for the noise-corrupted speech model. Due to the nonlinearity of the mismatch function, it is hard to directly incorporate Equ. (1) to ASR systems. A first-order VTS approximation [10] of Equ. (1) is proposed to estimate the corrupted static mean and covariance. To compensate the delta parameters, the continuous time approximation [14] is commonly used. The environment distortion (noise and channel) parameters are usually estimated per test utterance using an iterative EM algorithm. The standard VTS compensation assumes a clean speech model. To utilize the multi-condition data, a noise adaptive training (NAT) has been proposed [11].

2.1. VTS Compensation for DNN Front-end

A DNN is a multi-layer perceptron with many hidden layers. The main challenge in learning DNNs is to devise efficient strategies in order to escape poor local optimum of the complicated nonlinear error surface introduced by the large number of hidden layers. A common practice is to initialize the DNN weights layer by layer using generatively trained Restricted Boltzmann Machines (RBMs) before the discriminative joint fine-tuning of all the layers [1].

Directly applying the mismatch function Equ. (1) to compensate the DNN is similar to augment the DNN with one extra nonlinear input layer, which models the inverse of the mismatch function. Following the VTS convention, the weights of this layer should be estimated per test utterance in an unsupervised way, which is challenging for discriminative models especially when the mismatch is large. Instead, we propose to compensate the generative normalization front-end of the DNN per test utterance.

A common practice for neural network training is to first normalize the input features to have zero mean and unit variance in each dimension. Suppose $\boldsymbol{y}$ is a D-dimensional noisy input feature vector and the normalization parameters estimated from training data are

$$\boldsymbol{\mu}_m = \begin{bmatrix}\mu_{m,1} & \mu_{m,2} & \cdots & \mu_{m,D}\end{bmatrix}^{\top} \tag{2}$$

$$\boldsymbol{\Sigma}_m = \text{diag}\Big(\begin{bmatrix}\sigma^2_{m,1} & \sigma^2_{m,2} & \cdots & \sigma^2_{m,D}\end{bmatrix}^{\top}\Big) \tag{3}$$

The normalization process could be represented as a linear input layer in front of the DNN with the bias $\boldsymbol{b}$ and the weight matrix $\boldsymbol{W}$ as

$$\boldsymbol{b} = \begin{bmatrix}-\frac{\mu_{m,1}}{\sigma_{m,1}} & -\frac{\mu_{m,2}}{\sigma_{m,2}} & \cdots & -\frac{\mu_{m,D}}{\sigma_{m,D}}\end{bmatrix}^{\top} \tag{4}$$

$$\boldsymbol{W} = \text{diag}\Big(\begin{bmatrix}\frac{1}{\sigma_{m,1}} & \frac{1}{\sigma_{m,2}} & \cdots & \frac{1}{\sigma_{m,D}}\end{bmatrix}^{\top}\Big) \tag{5}$$

From the model perspective, this DNN normalization front-end captures the overall training data distribution and is generatively estimated. When applying the DNN model to the mismatched test data, we can do a global compensation to the overall data distribution by compensating this front-end. As it is effectively a single diagonal Gaussian, the GMM-based VTS compensation could be directly applied to $\boldsymbol{\mu}_m$ and $\boldsymbol{\Sigma}_m$ yielding the corrupted front-end parameters, $\hat{\boldsymbol{\mu}}_m$ and $\hat{\boldsymbol{\Sigma}}_{\boldsymbol{m}}$:

$$\hat{\boldsymbol{\mu}}^s_m = \boldsymbol{\mu}^s_m + \boldsymbol{\mu}^s_h + \text{g}(\boldsymbol{\mu}^s_m, \boldsymbol{\mu}^s_n, \boldsymbol{\mu}^s_h) \tag{6}$$

$$\hat{\boldsymbol{\Sigma}}^s_m = \boldsymbol{J}\boldsymbol{\Sigma}^s_m\boldsymbol{J}^{\top} + (\boldsymbol{I}-\boldsymbol{J})\boldsymbol{\Sigma}^s_n(\boldsymbol{I}-\boldsymbol{J})^{\top} \tag{7}$$

where the $\boldsymbol{\mu}_n$, $\boldsymbol{\mu}_h$ and $\boldsymbol{\Sigma}_n$ are the environment distortion parameters. The $\boldsymbol{I}$ is an identity matrix and the $\boldsymbol{J}$ is the Jacobian of the mismatch function with respect to the clean speech parameter. Dynamic parameters are commonly derived from the compensated static parameters using a continuous time approximation [14].

With $\hat{\boldsymbol{\mu}}_m$ and $\hat{\boldsymbol{\Sigma}}_{\boldsymbol{m}}$, the VTS compensated DNN front-end, $\hat{\boldsymbol{b}}$ and $\hat{\boldsymbol{W}}$, can be computed from the Equ. (4) and Equ. (5) by replacing each variable with its compensated version. The VTS-compensated normalized feature vector $\hat{\boldsymbol{x}}_{\text{NORM}}$, which is believed to match better to the training data, can then be computed by

$$\hat{\boldsymbol{x}}_{\text{NORM}} = \hat{\boldsymbol{W}}\boldsymbol{y} + \hat{\boldsymbol{b}} \tag{8}$$

To estimate the unknown environment distortion parameters, we could formulate them into the DNN's discriminative training framework. However, it may not work well with limited unsupervised data, *i.e.* the specific test utterance. In this work, we simply borrow the noise estimations from the VTS compensation of a conventional GMM-HMM system.

2.2. Feature-based VTS

Although our approach is formulated as a model based compensation, if we take the normalization front-end of the DNN as a feature processing step, it may look like a feature-based VTS. However, they are quite different. From [10], a GMM that represents the clean speech feature distribution has to be estimated additionally. The pseudo-clean features are estimated using the minimum mean square estimation (MMSE) from the noisy observations. With the first-order VTS approximation, they are computed as

$$\begin{aligned}\hat{\boldsymbol{x}}^s_{\text{MMSE}} &= \text{E}(\boldsymbol{x}^s|\boldsymbol{y}) = \int \boldsymbol{x}^s\ \text{p}(\boldsymbol{x}^s|\boldsymbol{y})d\boldsymbol{x}^s \\ &= \boldsymbol{y} - \sum_{k=0}^{K-1}\text{p}(k|\boldsymbol{y})(\boldsymbol{\mu}^s_h + \text{g}(\boldsymbol{\mu}^s_{x,k}, \boldsymbol{\mu}^s_n, \boldsymbol{\mu}^s_h))\end{aligned} \tag{9}$$

where $\text{p}(k|\boldsymbol{y})$ is the posterior probability for the kth Gaussian in the noise-compensated GMM given the noisy feature $\boldsymbol{y}$. The $\boldsymbol{\mu}_{x,k}$ is the mean of the kth Gaussian in the clean GMM. Comparing to this feature-based VTS, our approach (*cf.* Equ. (8)) not only shifts but also scales the noisy speech,. The additional scaling step actually captures the variance changes between the clean and noisy speech. Moreover, multiple Gaussians usually have to be estimated for the feature-based VTS, while only a single Gaussian is involved in our approach.

2.3. Adaptive Training

Similar to the GMM-HMM, the VTS compensation has the clean speech model assumption. When dealing with multi-condition data, the noise adaptive training (NAT) is commonly adopted. We also hope to take advantage of DNNs' powerful modeling capability to relieve the limitation of the single Gaussian based global compensation through the adaptive training. A NAT based on our front-end VTS compensation is thus developed. It is done as follows: 1) Train a DNN model from the multi-condition data and estimate the initial environment distortion parameters from the beginning and ending frames for each utterance (20 frames in our experiments); 2) Compensate the current DNN front-end and estimate a new set of distortion parameters with the current DNN hypotheses; 3) Re-train the DNN with the new noise compensated front-end; 4) Go back to step 2 until the recognition accuracy converges on the cross validation set. After the adaptive training, the distortion parameters are discarded and only the pseudo-clean DNN is kept for testing.

3. EXPERIMENTS

To justify the effectiveness of our proposed VTS compensation for the DNN, we conduct a series of experiments on the AURORA 2 task. It contains 8,440 sentences of clean training data and 8,440 sentences of multi-condition training data. The test sets comprise 8 different noises at 7 different noise levels, totally 56 different test scenarios. They are further grouped into three broad test sets, namely Set A with noise types seen in the training data, Set B with noise types unseen and Set C with both additive noise and channel distortion. The HMM baseline system has 16 states per digit and 20 Gaussian per state following the standard "complex back-end" AURORA 2 recipe [15]. A simple equal-probability digit loop language model is used for decoding. For all the VST compensations in this study, only the first-order approximation is used. Word Error Rates (WERs) averaged over SNR values of 0-20dB for each test set are reported.

3.1. Clean Training

In this experiment, we first test our proposed approach on the clean trained models. The 39D MFCC features consisting of 13 cepstral features projected from 26 log filter-banks (FBanks) with delta and accelerator features are used. The cepstral coefficient of order zero (C0) is used instead of the log energy. Besides the GMM-HMM model, an 8-hidden-layer DNN-HMM model is also trained on the clean training data. A context window of totally nine frames and 512 units per hidden layer are adopted. The front-end normalization parameters are estimated from the clean training data only once. The WERs for each test set of these two systems are tabulated in the row "-" (*i.e.* without any compensation) in Table 1. The performance for the clean test data is also presented. From the results we can see that both the clean GMM and DNN dramatically degrade in the noisy environment.

Table 1. Average WER (%) of AURORA 2 recognition results based on clean trained models using MFCCs.

System		Clean	Test Set			
			A	B	C	Avg
GMM	-	0.4	39.1	40.0	39.1	39.5
	VTS_F	0.4	14.1	13.2	14.8	13.9
	VTS_M	0.4	8.9	8.4	9.7	8.8
DNN	-	0.3	39.3	40.2	37.3	39.2
	UTT	0.2	15.2	12.6	14.5	14.0
	VTS_F	0.2	10.4	9.4	10.6	10.0
	VTS_M	0.2	15.7	13.6	15.8	14.9

A 2048-component GMM is estimated for the feature-based VTS compensation ("VTS_F" in Table 1) and 4 iterations of noise estimations are done for the model-based VTS compensation ("VTS_M" in Table 1) of the GMM-HMM system. The WERs on all the three test sets are greatly reduced especially using model-based VTS, from 39.5% to 8.8%. As feature-based VTS has no assumption of the recognition systems, we could directly apply it to the DNN, which yields an average WER of 10.0%. Compensating the DNN front-end with the distortion parameters borrowed from the GMM-HMM system reduces the baseline WER from 39.2% to 14.9%. Although it is not as effective as VTS on GMMs, the simple DNN front-end VTS compensation still reduces the DNN baseline WER by more than a half. One probable explanation would be that with thousands of Gaussians in the GMM systems the VTS compensation is more effective. As "VTS_M" is effectively estimating the testing normalization parameters from the training ones using VTS, we also train a DNN using per-utterance normalization and the results are listed under the column "UTT". It is slightly better than "VTS_M" due to the true normalization parameters per test utterance.

3.2. Multi-condition Training

Similarly, the 39D MFCC features of the multi-condition data are used to train the GMM-HMM and the DNN-HMM with the same model structures as before. The recognition performance is listed in the Table 2 with the same naming convention. The baseline DNN system has a relative 31.5% error reduction over the GMM baseline system, clearly indicating its superior acoustic modeling capability. However, comparing the three test sets, the DNN performs much better on the test set with seen noises (the Set A) and degrades on the Set B with different noises, especially on the Set C with additional channel distortions.

Both the feature-based VTS, "VTS_F", and the model-based VTS, "VTS_M", are applied. All the distortion parameters used for the DNN compensation are borrowed from the corresponding GMM-HMM systems. Although the VTS compensation has the clean speech model assumption, it still

works well for the multi-condition models. This may imply that the VST is not restricted to the additive noise and channel distortions but the more general data mismatch between the training and the testing. "VTS_M" consistently outperforms "VTS_F" on the GMM-HMM system; however for the DNN, our simple front-end based model compensation is slightly worse than the "VTS_F", 7.0% *vs.* 6.7%.

After the VTS compensation, the GMM has similar performance among the three test sets, while the WERs of our approach, *i.e.* the DNN "VTS_M", increase with the difficulty levels of the test sets. This may attribute to the fact that the distortion parameters are not directly optimized for the DNN and the global compensation may not be capable to capture all the variations. We then further investigate the noise adaptive training for both the two systems. For the feature-based NAT, "NAT_F", the canonical models are re-estimated on the pseudo-clean features after the distortion parameter estimation. From our experiments, one full iteration of re-training gives the best recognition performance for both the "NAT_F" and the model-based NAT, "NAT_M", which is listed in Table 2. After re-training the DNN with the VTS compensated normalization front-end we could achieve an average WER of 5.2%, which is relatively 18.8% lower than the GMM NAT's 6.4% and 8.8% than the DNN UTT's 5.7%. This could be attributed to the superior modeling capability of the DNN which relieves the limitation of the single Gaussian based front-end compensation. While for the feature-based NAT, a slightly degradation over the "VTS_F" only has been observed, 6.9% *vs.* 6.7%. However the DNN frame accuracy on the training data improves a lot. It may be explained by the fact that the imperfect pseudo-clean feature estimated by the feature-based VTS does not maintain all the necessary variations causing the over-fitting of the DNN on the training data.

All the current experiments are based on the MFCC features. MFCCs attempt to reduce the dimensionality of the input and decorrelate the feature dimensions such that diagonal Gaussians are sufficient for ASR systems. With a powerful model, less pre-processed input representations, such as FBank features [1] and waveforms [16], could yield better recognition performance. We then use the 40-dimensional log FBank features and the log energy with the delta and the accelerate features. Similarly, nine contextual frames are employed and totally 8 hidden layers are trained. Due to the much higher input feature dimension, 123 *vs.* 39, the size of each hidden layer is set to 1024 instead of the 512 used for MFCC features. The recognition performance for FBank features (Table 2) is the lowest for both the uncompensated model and our proposed simple model-based compensation. Due to the correlations among each FBank feature dimension, which are not well modeled by the diagonal GMM, the feature-based VTS compensation performs worse and degrades greatly in the "NAT_F" due to over-fitting. With the proposed model-based adaptive training, our method achieves a WER of 5.0%, which is a relatively 21.9% error reduction than the GMM-based NAT model.

Table 2. Average WER (%) of AURORA 2 recognition results based on multi-condition trained models.

System		Clean	Test Set			
			A	B	C	Avg
GMM MFCC	-	0.6	12.3	10.4	17.9	12.7
	VTS_F	0.5	8.0	7.7	8.0	7.9
	VTS_M	0.5	7.0	6.9	7.2	7.0
	NAT_F	0.5	6.7	6.4	7.0	6.6
	NAT_M	0.5	6.5	6.1	6.8	6.4
DNN MFCC	-	0.4	6.4	8.5	13.7	8.7
	UTT	0.5	5.1	6.3	5.8	5.7
	VTS_F	0.6	6.6	6.9	6.5	6.7
	VTS_M	0.3	6.7	6.8	8.3	7.0
	NAT_F	0.7	6.5	7.5	6.8	6.9
	NAT_M	0.2	4.7	5.7	5.3	5.2
DNN FBank	-	0.3	5.7	7.8	12.1	7.8
	UTT	0.3	4.6	5.7	5.6	5.2
	VTS_F	0.7	7.3	7.8	7.9	7.6
	VTS_M	0.2	5.9	7.4	6.7	6.6
	NAT_F	0.9	9.9	11.6	11.0	10.8
	NAT_M	0.2	4.2	5.7	5.3	5.0

4. CONCLUSIONS

In this paper, we propose a simple but effective model-based adaptive compensation approach for the DNN-based noise-robust speech recognition, which first compensates the normalization front-end of the DNN and then re-update all its back-end layers. We have demonstrated the effectiveness of our approach on the AURORA 2 task. It is more efficient than GMM-based VTS as only one Gaussian needs to be compensated per test utterance. With the adaptive training, the simple VTS-based DNN front-end compensation could yield a relatively 18.8% WER reduction over the GMM-based NAT system. Moreover, with log FBank features, we could achieve a relatively 21.9% improvement against the GMM NAT system. However, this approach is not as effective as VTS on clean trained speech models. One possible direction would be to estimate the environment distortion parameters directly from the DNN instead of borrowing from the GMM. Furthermore, instead of a global compensation of the single Gaussian based DNN normalization front-end, a GMM-based front-end may yield improved performance.

5. ACKNOWLEDGMENTS

This research is supported by the Singapore National Research Foundation under its International Research Centre @ Singapore Funding Initiative and administered by the IDM Programme Office.

6. REFERENCES

[1] A. Mohamed, G.E. Dahl, and G. Hinton, "Acoustic modeling using deep belief networks," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 20, no. 1, pp. 14–22, 2012.

[2] A. Mohamed, D. Yu, and L. Deng, "Investigation of full-sequence training of deep belief networks for speech recognition," in *Proc. Interspeech*. ISCA, 2010, pp. 2846–2849.

[3] G.E. Dahl, D. Yu, L. Deng, and A. 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.

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

[5] O. Vinyals, S.V. Ravuri, and D. Povey, "Revisiting recurrent neural networks for robust ASR," in *Proc. ICASSP*. IEEE, 2012, pp. 4085–4088.

[6] H.G. Hirsch and D. Pearce, "The Aurora experimental framework for the performance evaluation of speech recognition systems under noisy conditions," in *ASR2000-Automatic Speech Recognition: Challenges for the new Millenium ISCA Tutorial and Research Workshop (ITRW)*, 2000.

[7] A.L. Maas, Q.V. Le, T.M. ONeil, O. Vinyals, P. Nguyen, and A.Y. Ng, "Recurrent neural networks for noise reduction in robust ASR," in *Proc. Interspeech*. ISCA, 2012.

[8] L. Deng, A. Acero, L. Jiang, J. Droppo, and X. Huang, "High-performance robust speech recognition using stereo training data," in *Proc. ICASSP*. IEEE, 2001, vol. 1, pp. 301–304.

[9] ETSI, "Advanced front-end feature extraction algorithm," in *Technical Report. ETSI ES 202 050*, 2007.

[10] P.J. Moreno, B. Raj, and R.M. Stern, "A vector taylor series approach for environment-independent speech recognition," in *Proc. ICASSP*. IEEE, 1996, vol. 2, pp. 733–736.

[11] O. Kalinli, M.L. Seltzer, J. Droppo, and A. Acero, "Noise adaptive training for robust automatic speech recognition," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 18, no. 8, pp. 1889–1901, 2010.

[12] S.J. Rennie, P. Fousek, and P.L. Dognin, "Factorial hidden restricted boltzmann machines for noise robust speech recognition," in *Proc. ICASSP*. IEEE, 2012, pp. 4297–4300.

[13] A. Acero, *Acoustical and environmental robustness in automatic speech recognition*, Ph.D. thesis, Carnegie Mellon University, Pittsburgh, PA, 1990.

[14] R.A. Gopinath, M.J.F. Gales, P.S. Gopalakrishnan, S.B. Aiyer, and M.A. Picheny, "Robust speech recognition in noise - performance of the IBM continuous speech recogniser on the ARPA noise spoke task," in *Proceedings of the ARPA Spoken Language Systems Technology Workshop*, 1995, pp. 127–130.

[15] D. Pierce and A. Gunawardana, "Aurora 2.0 speech recognition in noise: Update 2," in *Proc. ICSLP*, 2002.

[16] N. Jaitly and G. Hinton, "Learning a better representation of speech soundwaves using restricted boltzmann machines," in *Proc. ICASSP*. IEEE, 2011, pp. 5884–5887.