TWO-STAGE NOISE AWARE TRAINING USING ASYMMETRIC DEEP DENOISING AUTOENCODER

Kang Hyun Lee, Shin Jae Kang, Woo Hyun Kang and Nam Soo Kim

Department of Electrical and Computer Engineering and INMC,
Seoul National University, Gwanak P.O.Box 34, Seoul 151-744, Korea
E-mail: {khlee, sjkang, whkang}@hi.snu.ac.kr, nkim@snu.ac.kr

ABSTRACT

Ever since the deep neural network (DNN)-based acoustic model appeared, the recognition performance of automatic speech recognition has been greatly improved. Due to this achievement, various researches on DNN-based technique for noise robustness are also in progress. Among these approaches, the noise-aware training (NAT) technique which aims to improve the inherent robustness of DNN using noise estimates has shown remarkable performance. However, despite the great performance, we cannot be certain whether NAT is an optimal method for sufficiently utilizing the inherent robustness of DNN. In this paper, we propose a novel technique which helps the DNN to address the complex connection between the input and target vectors of NAT smoothly. The proposed method outperformed the conventional NAT in Aurora-5 task.

Index Terms— Deep neural networks (DNNs), robust speech recognition, noise aware training (NAT), denoising autoencoder.

1. INTRODUCTION

In recent years, deep learning has been prevalent in signal processing and it has become an opportunity for automatic speech recognition (ASR) to progress. Especially in acoustic modeling, introduction of the deep neural network (DNN)-hidden Markov model (HMM) system which represents the relationship between the acoustic features and HMM states using DNN instead of Gaussian mixture model (GMM) is considered as a breakthrough. DNN-HMM system has outperformed the conventional GMM-HMM system in a variety of ASR tasks [1, 2, 3]. The remarkable performance of the DNN-HMM system is attributed to its capability in automatically learning complicated non-linear mapping from the input to the target vectors. If a sufficient amount of training data is available, more complicated input-target relationship can be easily learned by using wider and deeper neural network architectures [4].

Interest in DNN's efficient learning capability has also been expanded to the robust speech recognition area. DNN-based approaches to noise robustness can generally be divided into two categories: feature-based and model-based techniques. The feature-based techniques [5, 6] directly train an arbitrary unknown mapping from the noisy to the clean speech features unlike the conventional techniques [7, 8, 9, 10] which usually require some specific models or formulations to account for the relationship. Among these techniques, feature enhancement algorithms based on deep denoising autoencoder (DDAE) has demonstrated its superiority in reconstructing the clean features from noisy features [11, 12]. The model-based techniques leave the observations unaltered and instead let the DNN parameters automatically find out the relationship between the observed speech and the phonetic targets. This approach is referred to the multi-condition training which has been widely used in robust ASR.

Furthermore, some other techniques augment additional information such as the background noise estimate or speaker information to the input vector in order to improve the modeling power of the DNN [13, 14]. Particularly a technique referred to noise-aware training (NAT) attained the state-of-the-art results on Aurora-4 task [13]. An interesting property of NAT is that it follows the general procedure of the multi-condition DNN-based acoustic model training, except for the fact that it adds an input vector relevant to the environmental condition. NAT enables the DNN to learn the relationship among noisy input, noise features and target vectors corresponding to the phonetic identity by augmenting an estimate of the noise present in the input signal. Due to its easy implementation and good performance, NAT has already been applied actively in speech enhancement and robust ASR [15].

Despite its success in robust ASR, yet we cannot be certain whether NAT is an optimal method in taking advantage of the inherent robustness of the DNN framework. Although NAT somewhat contributes to the noise robustness of DNN,

This work was supported in part by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MEST) (NRF-2015R1A2A1A15054343), and by the MSIP (Ministry of Science, ICT and Future Planning), Korea, under the ITRC (Information Technology Research Center) support program (IITP-2015-H8501-15-1016) supervised by the IITP (Institute for Information & communications Technology Promotion).

its performance in adverse environment is still far from that shown in clean condition. One of the fundamental reasons for this phenomenon is that the current NAT framework is considered insufficient to make the DNN implement the mapping from noisy speech and noise estimates to phonetic targets as clearly as it addresses the relationship between clean speech and the corresponding phonetic targets. A promising way to improve the NAT is to extract some representation relevant to clean speech features and then to implement the mapping from this representation to the phonetic targets.

In this paper, we propose a novel approach to DNN training which can be a solution to the aforementioned issue of NAT. The main idea of the proposed approach is to let the DNN clarify the relationship among noisy features, noise estimates and phonetic targets only after reconstructing the clean features. In order to accomplish this, the proposed technique cascades two individually fine-tuned DNNs into a single DNN. The first DNN performs reconstruction of the clean features from noisy features when noise estimates are augmented. In order to reflect information of the noise estimates in the reconstruction process effectively, we apply the DDAE with a little modification to its output structure. Then the next DNN attempts to learn the mapping between the reconstructed features and the phonetic targets. The performance of the proposed approach is evaluated on the Aurora-5 task and better performance is observed compared to the conventional NAT.

2. A BRIEF REVIEW ON NOISE AWARE TRAINING

In this work, for a simple problem formulation, we will only consider acoustic environments where the background noises are dominant factors of speech degradation. Let us denote an observed noisy feature, the corresponding unknown clean feature, the corrupting noise and a HMM state identity being extracted at the t-th frame as $\mathbf{y}_t$, $\mathbf{x}_t$, $\mathbf{n}_t$ and $\mathbf{s}_t$, respectively. Additionally, we denote a subsequence of vectors $\mathbf{x}_{m_1}\mathbf{x}_{m_1+1}\cdots\mathbf{x}_{m_2}$ from frame index m_1 to m_2 as $\mathbf{x}_{m_1}^{m_2}$. Under the general framework of HMM-based recognition, we assume that there exists an unknown underlying function that approximates the posterior probabilities of the HMM states given as follows:

$$p(\mathbf{s}_t|\mathbf{y}_1^T) \cong f(\mathbf{y}_{t-\tau}^{t+\tau}, \mathbf{n}_{t-\tau}^{t+\tau}) \tag{1}$$

where $f(\cdot)$ represents the function that maps the noisy and noise features to the corresponding HMM state identity which contains phonetic information, T denotes the length of the input feature, and the subscript τ represents the temporal coverage which is required for figuring out the contextual information of the speech signal.

Since the true noise features $\mathbf{n}_{t-\tau}^{t+\tau}$ in (1) are unknown, NAT replaces them with a single noise estimate. The input vector of NAT is formed by augmenting the noise estimate with a window of consecutive frames of noisy feature, i.e.,

$$\mathbf{v}_t = [\mathbf{y}_{t-\tau}^{t+\tau}, \widehat{\mathbf{n}}_t] \tag{2}$$

where $\mathbf{y}_{t-\tau}^{t+\tau}$ represents a window of 2τ + 1 frames of noisy speech features and $\widehat{\mathbf{n}}_t$ represents a noise estimate. The target vector of the NAT network is given as the one-hot encoding label concerned with the tied HMM states (senone) as in common DNN-based acoustic models. By applying this simple process to both training and decoding, the DNN can automatically learn the complex mapping from the noisy speech and noise estimate to the HMM state labels.

However, even though this approach guarantees a certain level of improvement in noise robustness, we need to check whether the non-linear mapping obtained from NAT can be generalized well. Although NAT aims to generate internal representations that are robust to noise, when comparing its recognition performance in noisy environment with that in clean environment, we can easily discover that there still exists a large performance gap. For this reason, we need a more sophisticated technique to improve the modeling power of the NAT.

3. TWO-STAGE NAT

In this section, we propose a novel approach to improve NAT. The basic idea of the proposed approach starts from the assumption that the underlying function $f(\cdot)$ in (1) can be expresses as a composition of two separate functions as follows:

$$p(\mathbf{s}_t|\mathbf{y}_1^T) \cong f(\mathbf{y}_{t-\tau}^{t+\tau}, \mathbf{n}_{t-\tau}^{t+\tau}) \cong h \circ g(\mathbf{y}_{t-\tau}^{t+\tau}, \mathbf{n}_{t-\tau}^{t+\tau}) \tag{3}$$

where the output of $g(\cdot)$ is a clean feature vector stream,

$$\mathbf{x}_{t-\tau}^{t+\tau} \cong g(\mathbf{y}_{t-\tau}^{t+\tau}, \mathbf{n}_{t-\tau}^{t+\tau}), \tag{4}$$

and

$$p(\mathbf{s}_t|\mathbf{y}_1^T) \cong h(\mathbf{x}_{t-\tau}^{t+\tau}). \tag{5}$$

In (3)-(5), $g(\cdot)$ represents a function which deals with the mapping from the noisy and noise features to the clean speech features and $h(\cdot)$ is a function predicting the phonetic target based on the clean speech feature stream. To mimic this function structure, we propose a DNN as shown in Fig. 1. The whole DNN is constructed by concatenating two individually fine-tuned DNNs and each separate DNN approximates the function $g(\cdot)$ and $h(\cdot)$ in (3). The first DNN which is based on DDAE is applied to separate the clean speech features from the corruption noises. We call this DNN the lower DNN since it is placed in the lower part of the DNN in Fig. 1. The second DNN which is called the upper DNN, deals with modeling the relationship between the output vector generated by the lower DNN and the phonetic target.

3.1. Lower DNN

For training the lower DNN, we apply the DDAE which has proven its capability of reducing the distortion in the original noisy feature [11]. Although it is a common method that

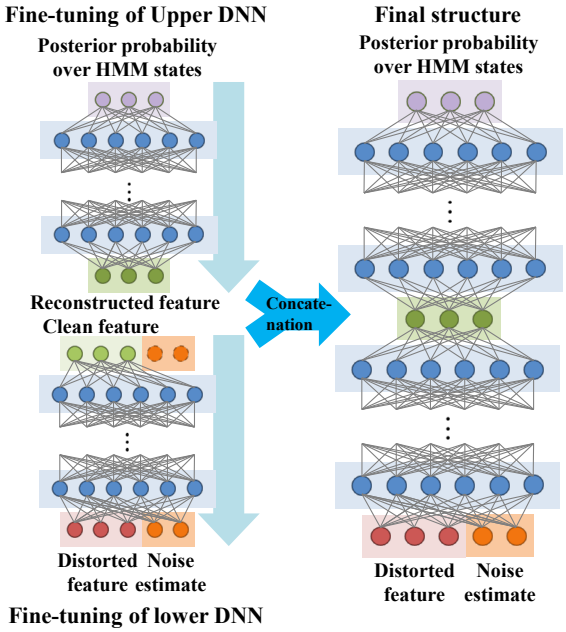

Fig. 1. DNN structure of proposed technique.

DDAE is used to reconstruct clean features from only noisy speech features [16], the DDAE in this technique is designed somewhat differently. The output layer of the DDAE corresponds to the clean speech features and the noise features and the input layer is given by (2). From this structure we can see that the DDAE extracts not only the clean speech and noise features from the noisy input features augmented with noise estimate.

Since the purpose of the lower DNN is to extract only the clean features, we exclude the noise-related nodes in the output layer after training as depicted in Fig. 1. Therefore, the DDAE is designed to have an asymmetric structure where the dimensions of the input and output vector are different and the output vector of the lower DNN can be represented as follows:

$$\widehat{\mathbf{v}}_t = [\widehat{\mathbf{x}}_{t-\tau}^{t+\tau}] \tag{6}$$

i.e, a window of $2\tau + 1$ frames of clean speech feature estimates. To obtain the noise estimate $\widehat{\mathbf{n}}_t$ in (2), a time-varying environmental estimation approach based on the interacting multiple model (IMM) algorithm is utilized [17]. By reflecting the dynamic environmental estimate through the IMM technique to the input of the network at each frame, we can expect the lower DNN to reconstruct the clean features considering the real-time noise at each frame.

3.2. Upper DNN

In the training stage of the upper DNN training, the network learns the mapping between the output vector of the lower DNN $\widehat{\mathbf{v}}_t$ in (6) and the corresponding one-hot encoding label which contains information of the HMM states. Through the mapping, prediction of the posterior probabilities of the HMM states from the reconstructed features is performed. This training method can be seen as feature-space noise adaptive training which is demonstrated to show worse performance than multi-condition DNN-HMM [13]. However, $\widehat{\mathbf{v}}_t$ here has different characteristic with the feature vector obtained from the conventional feature enhancement techniques. Since $\widehat{\mathbf{v}}_t$ is acquired by the lower DNN, the reconstructed vector is free from information loss caused by using linear approximations which are used in the conventional techniques [7, 8, 9, 10]. Especially, since the initial values of the lower DNN including its output layer parameters are set elaborately through the asymmetric DDAE utilizing environmental estimates, it is possible to generate a feature vector with phonetic discriminative information.

Table 1. WERs (%) of *Baseline*, *NAT* and *TS-NAT* for non-filtered and g. 712 filtered test data sets averaged over noise types on Aurora-5 task without drop training.

SNR (dB)	Non-filtered			G.712 filtered		
Method	*Baseline*	*NAT*	*TS-NAT*	*Baseline*	*NAT*	*TS-NAT*
Clean	1.32	1.25	**0.89**	0.90	0.87	**0.71**
15	1.88	1.95	**1.51**	1.28	1.21	**0.94**
10	3.33	3.42	**2.88**	2.09	1.94	**1.60**
5	7.83	8.09	**7.14**	4.71	4.36	**4.06**
0	20.85	20.67	**19.64**	13.13	11.94	**11.92**
Avg.	7.04	7.08	**6.41**	4.42	4.06	**3.85**

Table 2. WERs (%) of *Baseline*, *NAT* and *TS-NAT* for non-filtered and g. 712 filtered test data sets averaged over noise types on Aurora-5 task with dropout training.

SNR (dB)	Non-filtered			G.712 filtered		
Method	*Baseline*	*NAT*	*TS-NAT*	*Baseline*	*NAT*	*TS-NAT*
Dropout	20%	20%	20%	20%	20%	20%
Clean	1.32	1.05	**0.91**	0.84	0.78	**0.85**
15	1.87	1.78	**1.52**	0.90	1.15	**0.92**
10	3.29	3.18	**2.59**	1.89	1.88	**1.31**
5	7.77	7.62	**6.63**	4.33	3.97	**3.68**
0	20.60	19.92	**19.30**	11.92	11.57	**11.36**
Avg.	6.97	6.71	**6.19**	3.98	3.87	**3.62**

4. EXPERIMENTS

The performance of the proposed method was evaluated on Aurora-5 task [18]. In order to compare the performance of the proposed technique (*TS-NAT*), two different versions of DNN-HMM were trained. The first one was the basic multi-condition DNN-HMM (*Baseline*) and the second one was the DNN-HMM based on NAT (*NAT*). Also, the performance evaluation with dropout technique [19] which is widely used in the DNN training was also investigated.

4.1. Aurora-5 task and GMM-HMM system

The Aurora-5 task was developed to investigate the performance of speech recognition for speech recorded with hands-free device in noisy room environments. The test data con-

sisted of two sets: G. 712 filtered and non-filtered sets. The G. 712 filtered set comprised clean speech utterances where randomly selected car or public space noise samples were added at signal-to-noise ratio (SNR) levels 0 to 15 dB. The non-filtered set consisted of clean speech utterances where randomly selected interior noises were augmented at the same SNR range above.

In these experiments, we used multi-condition training data for training all the DNN-based techniques and the GMM-HMM systems were built based on the clean speech data provided by the G. 712 filtered and non-filtered data sets. The number of utterances used for HMM training was 8,623 for each data set. The input features were 39-dimensional MFCC features (static plus first and second order delta features) and cepstral mean normalization was performed. Each word in the vocabulary, which was designed based on TI-Digits DB, was modeled by a left-to-right structured HMM consisting of 16 states and 4 Gaussian mixture components per state. The training of the HMM parameters and Viterbi decoding for speech recognition was carried out using HTK software [20]. Also, the state labels for the frames were obtained from the forced alignment of clean speech data with HVite command of HTK 3.4.1 using the GMM-HMM acoustic models.

4.2. Structure and training of DNNs

For training all the DNN-based acoustic models, log mel filterbank (FBANK) feature of 23-dimension was used as an input. As in the case of MFCC feature above, both the first and second-order derivative of FBANK features were used. The input layer for *Baseline* was formed from a context window of 11 frames having 759 visible units for the network and that of *NAT* had total 828 visible units by augmenting the input vector of *NAT* with the IMM-based noise estimate. Both DNNs had 11 hidden layers with 2,048 hidden units in each layer and the final soft-max output layer had 179 units, each corresponding to the states of the HMM systems. Both networks were initialized using stack of RBMs and each RBM was trained using contrastive divergence [21]. The fine-tuning of the two networks were performed using cross entropy as the loss function by error back propagation supervised by state IDs for frames. The mini-batch size for the stochastic gradient descent algorithm was set to be 256. The learning rate was initially set to be 0.01 and exponentially decayed over each epoch with decaying factor 0.95. The momentum was set to be 0.9. The training was stopped after 50 epochs.

The DDAE for training the lower DNN had six layers in total consisting of three encoding layers and three decoding layers. The number of nodes in each layer was set to be 2,048 except for the input and output layers. The input layer of the DDAE was equal to that of *NAT*. Each layer of encoding layers was initialized using the weights and hidden unit biases of RBM. Then the decoding layers were initialized using the transpose of the weights and the visible unit biases of encoding layers. After the initialization, 759 nodes related with clean features were chosen from total 828 output nodes. The fine-tuning of the asymmetric DDAE was performed by error back propagation with squared error between output vector and clean feature as the loss function. The learning rate was initially set to be 0.005 and exponentially decayed over each epoch with decaying factor 0.9 until the training was stopped after 30 epochs.

The upper DNN had 5 hidden layers with 2048 hidden units. And the final soft-max output layer had 179 units in common with the other DNN-HMMs above. The rest of the training configurations were the same with those of the other DNN-HMMs. All the techniques attempted in this experiments were based on wide and very deep DNN structures. In training and optimizing these heavy networks, overfitting can be a serious problem. Considering this issue, we used the dropout technique which has already proved its regularization capability [19]. The dropout percentage of 20% were applied to the three different techniques with the other settings for DNN training unchanged.

4.3. Performance evaluations on Aurora-5 task

We compared performance of *TS-NAT* with those of *Baseline* and *NAT* on Aurora-5 task. The word error rates (WERs) of the three approaches are shown in Table I. We can see that both *NAT* and *TS-NAT* outperformed *Baseline* in almost every condition. It demonstrates that the dynamic noise estimate obtained from IMM technique obviously helps the DNN-based acoustic models to reflect environmental factors. Also, comparing *TS-NAT* with *NAT*, the performance of *TS-NAT* is superior to that of *NAT* irrespective of SNRs. When the dropout training is applied, the degree of improvement of the proposed technique is enlarged. With dropout training performed, the average relative error rate reductions (RERRs) of *TS-NAT* over *NAT* at SNRs were 7.75% and 6.36% in non-filtered and G.712 filtered set. This confirms that our proposed approach which intervenes NAT through information of reconstructed clean speech can be effective in making the DNN learn the complex relationship among noise, noisy and phonetic information.

5. CONCLUSION

In this paper, we have proposed a novel technique of DNN-based acoustic model designed for effective usage of multi-condition data and its noise estimate. The proposed technique addresses the mapping from noisy speech and noise estimates to phonetic targets effectively by concatenating two DNNs which take role of clean feature reconstruction and prediction of posterior probability over HMM states respectively. Through a series of experiments on Aurora-5 task, we have found that the proposed technique outperforms NAT in word accuracy. Future study will deal with techniques considering other environmental factors such as reverberation.

6. REFERENCES

[1] A. Mohamed, G. E. Dahl, and G. E. Hinton, "Acoustic modeling using deep beliefs networks," *IEEE Trans. Audio, Speech, Language Process.*, vol. 20, no. 1, pp. 14-22, Jan. 2012.

[2] G. E. Dahl, D. Yu, L. Deng, and A. Acero, "Context-dependent pre-trained deep neural networks for large vocabulary speech recognition," *IEEE Trans. Audio, Speech, Language Process.*, vol. 20, no. 1, pp. 30-42, Jan. 2012.

[3] B. Kingsbury, T. N. Sainath, and H. Soltau, "Scalable minimum Bayes risk training of deep neural network acoustic models using distributed Hessian-free optimization," in *Proc. Interspeech*, 2012, pp. 10-13.

[4] D. Yu, M. L. Seltzer, J. Li, J. Huang, and F. Seide, "Feature learning in deep neural networks - A study on speech recognition tasks," *CORR*, vol. abs/1301.3605, 2013.

[5] A. Narayanan, and D. Wang, "Investigation of speech separation as a front-end for noise robust speech recognition," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 22, no. 4, pp. 826-835, Apr. 2014.

[6] W. Li, L. Wang, Y. Zhou, J. Dines, M. Magimai.-Doss, H. Bourlard, and Q. Liao, "Feature mapping of multiple beamformed sources for robust overlapping speech recognition using a microphone array ," *IEEE/ACM Trans. Audio, Speech, Language Process*., vol. 22, no. 12, pp. 2244-2255, Dec. 2014.

[7] Y. Ephraim and D. Malah, "Speech enhancement using a minimum mean-square error short-time spectral amplitude estimator," *IEEE Trans. Acoust., Speech, Signal Process*., vol. 32, no. 6, pp. 1109-1121, Dec. 1984.

[8] D. Yu, L. Deng, J. Droppo, J. Wu, Y. Gong, and A. Acero, "Robust speech recognition using a cepstral minimum-meansquare-error-motivated noise suppressor," *IEEE Trans. Audio, Speech, Language Process*., vol. 16, no. 5, pp. 1061-1070, Jul. 2008.

[9] N. S. Kim, "IMM-based estimation for slowly evolving environments," *IEEE Signal Process. Lett*., vol. 5, no. 6, pp. 146-149, Jun. 1998.

[10] L. Deng, J. Droppo, and A. Acero, "Recursive estimation of non-stationary noise using iterative stochastic approximation for robust speech recognition," *IEEE Trans. Speech, Audio, Process*., vol. 11, no. 6, pp. 568-580, Nov. 2003.

[11] X. Feng, Y. Zhang, and J. Glass, "Speech feature denoising and dereverberation via deep autoencoders for noisy reverberant speech recognition," in *Proc. ICASSP*, 2014, pp. 1759-1763.

[12] M. Mimura, S. Sakai, and T. Kawahara, "Exploring deep neural networks and deep autoencoders in reverberant speech recognition," in *HSCMA*, 2014, pp. 197-201.

[13] M. Seltzer, D. Yu, and Y. Wang, "An investigation of deep neural networks for noise robust speech recognition," in *Proc. ICASSP*, 2013, pp. 7398-7402.

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

[15] Y. Xu, J. Du, L.-R. Dai, and C.-H. Lee, "Dynamic noise aware training for speech enhancement based on deep neural networks," in *Proc. Interspeech*, 2014, pp. 2670-2674.

[16] P. Vincent, H. Larochelle, Y. Bengio, and P. A. Manzagol, "Extracting and composing robust features with denoising autoencoders," in *Proc. ICML*, 2008, pp. 1096-1103.

[17] C. W. Han, S. J. Kang, and N. S. Kim, "Reverberation and noise robust feature compensation based on IMM," *IEEE Trans. Audio, Speech, Language Process*., vol. 21, no. 8, pp. 1598-1611, Aug. 2013.

[18] H. G. Hirsch, AURORA-5 experimental framework for the performance evaluation of speech recognition in case of a hands-free speech input in noisy environments Niederrhein Univ. of Appl. Sci., Nov. 2007.

[19] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929-1958, 2014.

[20] S. Young *et al*., *The HTK book*. Cambridge, U.K.: Cambridge Univ. Eng. Dept., 2006.

[21] G. E. Hinton, S. Osindero, and Y. W. Teh, "A fast learning algorithm for deep belief nets," *Neural Computation*, vol. 18, no. 7, pp. 1527-1554, 2006.