REVERBERANT SPEECH RECOGNITION COMBINING DEEP NEURAL NETWORKS AND DEEP AUTOENCODERS

Masato Mimura, Shinsuke Sakai, and Tatsuya Kawahara

Academic Center for Computing and Media Studies, Kyoto University
Sakyo-ku, Kyoto 606-8501, Japan
{mimura|sakai|kawahara}@ar.media.kyoto-u.ac.jp

ABSTRACT

We propose an approach to reverberant speech recognition adopting deep learning in front end as well as back end of the system. At the front end, we adopt a deep autoencoder for enhancing the speech feature parameters, and the recognition is performed using a DNN-HMM acoustic models trained on multi-condition data. The system was evaluated through the ASR task in Chime Challenge 2014. The DNN-HMM system trained on the multi-condition training set achieved a conspicuously higher word accuracy compared to the MLLR-adapted GMM-HMM system trained on the same data. Furthermore, feature enhancement with the deep autoencoder contributed to the improvement of recognition accuracy especially in the more adverse conditions. When the DNN-HMM was used without the deep autoencoder front end, it resulted in a better performance than the non-adapted GMM-HMM system, but was not as good as the adapted GMM-HMM system. However, it outperformed the adapted GMM-HMM system when combined with the deep autoencoder.

Index Terms— reverberant speech recognition, Deep Neural Network(DNN), deep autoencoder

1. INTRODUCTION

In recent years, the speech recognition technology based on statistical techniques achieved a remarkable progress supported by the ever increasing training data and the improvements in the computing resources. Applications such as voice search are now being used in our daily life. However, speech recognition in adverse conditions is still a difficult task and the recognition accuracies in adverse environments such as those with reverberation and background noise are still staying at low levels.

A key breakthrough for speech recognition technology to be accepted widely in the society will be the establishment of the methodology for easier speech interface with hands-free input. Speech reverberation adversely influences the speech recognition accuracy in such conditions and various efforts have been made to improve the recognition performance for the reverberant speech.

Reverberant speech recognition has so far been tackled by applying feature enhancement at the front end, and by attempting model adaptation and the use of more sophisticated recognition techniques. Speech enhancement techniques include deconvolution approaches that tries to reconstruct clean speech by inverse filtering the reverberant speech [1][2][3] and spectral enhancement approaches that estimate and remove the influences of the late reflection [4][5]. Since improvement measured by SNR may not be directly related to the speech recognition accuracy, there also are approaches to enhance speech based on speech recognition likelihoods in the back end [6]. One of the simplest approach to feature enhancement is the cepstral mean normalization (CMN) [7]. However, since reverberation time is usually larger than the frame window length for feature extraction, its effectiveness is rather limited. A major back end approach is the use of maximum-likelihood linear regression (MLLR) [8] that tries to adapt the acoustic model parameters to the corrupted speech.

In this paper, we take an approach to reverberant speech recognition based on deep learning, which has been drawing much attention in the speech research community in recent years. Recognition of reverberant speech is performed combining “standard” DNN-HMM [9] decoding and a feature enhancement through deep autoencoder [10][11]. The combination of the DNN classifier and the deep autoencoder can be regarded as a single DNN classifier with a very deep structure. However, we can expect a mutually complementary effects from the combination of two networks that are optimized toward different targets. We have so far seen few practices of applying deep neural network technology to LVCSR in the adverse conditions such as reverberant and noisy speech, and we expect to be able to contribute to the research community by presenting some interesting results.

2. ASR TASK IN REVERB CHALLENGE

The proposed system was evaluated following the instructions for the ASR task of the Reverb Challenge 2014 [12].

For training, we used the standard multi-condition data that is built by convolving clean WSJCAM0 data with room impulse responses (RIRs) and subsequently adding noise signals. Evaluation data consists of “SimData” and “RealData”. SimData is a set of reverberant speech simulated by convolving clean speech with various RIRs and adding measured noise signals to make the resulting SNR to be 20dB. RIRs were recorded in three different-sized rooms (small, medium, and large) and with two microphone distances (near=50cm and far=200cm). The reverberation time (T60) of the small, medium, and large rooms are about 0.25s, 0.5s, and 0.7s, respectively. These rooms are different from those for measuring RIRs used in generating multi-condition training data. RealData was recorded in a different room from those used for measuring RIRs for SimData. It has the reverberation time of 0.7s. There are two microphone distances with RealData, which are near (≈100cm) and far (≈250cm). Utterance texts for both SimData and RealData were chosen from WSJCAM0 prompts. All the reverberant speech recordings were made with eight microphones. In the experiments in this paper, however, we only use a single channel both for training and testing. The speech recognition performance is measured by word error rate.

3. DNN-HMM

Pattern recognition by neural networks has a long history [13]. In recent years, deep structured neural networks (DNN) has been drawing much attention again in the pattern recognition field due to the establishment of an effective pre-training methodology [14] and the dramatic improvement of computing power and the increase of available training data. It has also been applied to speech recognition combined with hidden Markov models (HMM) and reported to achieve significantly higher accuracy than conventional GMM-HMM technology in various task domains [9][15][16][17].

There has so far been two typical ways to combine DNNs and HMMs. In one approach, the state emission probabilities are computed using DNNs instead of the conventional Gaussian mixture models (GMMs). In another approach, the output from DNNs are utilized as input to conventional GMM-HMMs. The former is called the hybrid approach [9][16][17] and the latter is called the TANDEM approach [18][19][20]. In this paper, we build acoustic models adopting the hybrid approach, which has a simple structure and therefore is easy to handle and has been shown to be effective in many task domains. We call these acoustic models built with hybrid approach as *DNN-HMM* hereafter in this paper.

In the training of deep neural networks, the standard error backpropagation training from randomly initialized states often does not yield the expected results due to the very little changes especially in the shallow layer parameters caused by the repeated multiplications of the values smaller than one. Therefore, we opt to initialize the network weights in a better way by unsupervised generative training before the supervised discriminative training [14].

In the first place, each layer of the network is trained as a restricted Boltzmann machine (RBM) independently. Next, these RBMs are stacked together to constitute a deep belief network (DBN). An initial deep neural network is then established by adding a randomly initialized softmax layer. This DNN is trained in a supervised way through error backpropagation using HMM state IDs as labels.

It has been a standard practice to train the neural networks for DNN-HMMs independent of other components of the HMM models. The model parameters other than the DNN components are usually copied from well-trained GMM-HMMs, for example, those trained according to the minimum phone error criterion. The state labels are also usually generated by the forced alignments using those GMM-HMMs.

4. SPEECH FEATURE ENHANCEMENT BY DEEP AUTOENCODERS

The deep network structure described in the last section can be utilized as a deep autoencoder when trained for a different target [21]. In this case, the lower layers are regarded as an encoder to obtain an efficient code and the upper layers are regarded as a decoder that "reverses" the encoder. As a whole, a deep autoencoder has a vertically symmetric network structure.

Initialization by RBM training is very important with deep autoencoders as well. However, each of the networks in the decoder layers are initialized with the same RBM in the encoder-layer counterpart. In decoder layers, network weights are initialized as the transpose of those used for the correspondent encoder layer network and biases are initialized using visible biases from RBMs rather than hidden biases that are used for the encoder layers.

This deep network with a symmetric structure can be used as a *denoising autoencoder* when input is a corrupted data and the target is the clean data [22]. It is trained to recover the clean data from the corrupted data. Other than the input and the target, the training algorithm is the same as the ordinary autoencoders [23].

5. EXPERIMENTAL EVALUATIONS

Experimental evaluations were performed for DNN-HMMs and denoising autoencoders (DAEs) described in the previous sections using evaluation data for Reverb Challenge [12].

In all of the experiments presented below, single channel data was only used for training and testing. For training, we used the 7,861 utterances of multi-condition data, which was also the training data for multi-condition baseline GMM-HMM models. For decoding, we used the HVite command from HTK-3.4 with a small modification to handle DNN output. The language model we used is the baseline language model supplied in the Reverb Challenge. Decoding parameters such as beam widths are set to be the same for GMM-HMM system and DNN-HMM system. Since the "likelihood" scores have different ranges, the language model weights and insertion penalties are independently optimized for each system.

The evaluation results obtained with the baseline GMM-HMM system are shown in Table 1, rows 1 through 3.

5.1. DNN-HMM

Here we describe the details of the DNN-HMM system we used for the evaluation experiments.

A 1,320-dimensional feature vector consisting of eleven frames of 40-channel log Mel-scale filter bank outputs and their delta and acceleration coefficients is used as the input to the network. The targets are chosen to be the 3,113 shared states of the baseline GMM-HMMs. The six-layer network consists of five hidden layers and a softmax output layer. Each of the hidden layers consists of 2,048 nodes. The network is initialized using the RBMs trained with reverberant speech.

The fine-tuning of the DNN is performed using cross entropy as the loss function by error backpropagation supervised by state identifiers for frames. The training data is the same multi-condition data as the baseline GMM-HMM system. The mini-batch size for the stochastic gradient descent algorithms was set to be 256. The learning rate was set to be 0.08 initially and exponentially decayed over the sequence of mini-batches. The momentum was set to be 0.9. The training was stopped after 10 epochs. The state labels for the frames were generated with HVite command of HTK3.4 using the baseline GMM-HMM acoustic models trained on MFCC feature parameters of multi-condition data by maximum likelihood criterion. The HMM model parameters other than emission probabilities such as transition probabilities were copied from the baseline GMM-HMM models.

The word error rates for the evaluation data set obtained with the DNN-HMM system trained using multi-condition data are shown in the fourth row of Table 1. For all subsets of the "SimData" part of the evaluation set, the DNN-HMM system achieved drastically higher accuracies than the adapted GMM-HMM system. In the most adverse condition (Room 3, Far), word error rate was reduced by 8.6 points (from 39.28% to 30.64%).

With the "RealData" subsets, the DNN-HMM system achieved higher accuracies than the non-adapted GMM-HMMs, but did not reach the accuracies attained by the adapted GMM-HMMs.

The experimental results so far described (corresponding to Table 1, row 4) are our official results submitted in time for the Challenge deadline.

Table 1. System performances on the test data (word error rate (%))

		SimData							RealData		
		Room 1		Room 2		Room 3		Ave.	Room 1		Ave.
# of ch	Proc. Scheme	Near	Far	Near	Far	Near	Far		Near	Far	
1	Baseline(clean, w/o CMLLR)	18.26	25.60	41.87	82.20	53.59	87.99	51.73	89.91	87.58	88.74
1	Baseline(multicond, w/o CMLLR)	21.28	21.18	23.12	38.83	28.24	44.77	29.56	58.96	55.60	57.28
1	Baseline(multicond, w CMLLR)	16.57	18.21	20.31	32.43	24.86	39.28	25.27	50.37	48.01	49.19
1	DNN-HMM(multicond)	12.33	13.18	16.16	26.72	18.77	30.64	19.63	53.72	52.26	52.99
1	DAE + DNN-HMM(multicond)	16.96	16.57	15.90	22.61	16.33	21.75	18.35	45.96	45.51	45.74
1	DAE + DNN-HMM(retrain)	13.74	14.84	15.63	25.36	17.91	28.44	19.32	51.45	51.38	51.42

Table 2. System performances on clean data (word error rate(%))

		ClnData			
		Room 1	Room 2	Room 3	Ave.
# of ch	Proc. Scheme				
1	Baseline(clean, w/o CMLLR)	13.01	12.69	12.23	12.64
1	Baseline(multicond, w/o CMLLR)	30.92	30.28	30.17	30.46
1	Baseline(multicond, w CMLLR)	16.25	15.28	15.37	15.63
1	DNN-HMM(multicond)	12.41	12.40	13.63	12.81
1	DAE + DNN-HMM(multicond)	22.23	21.96	21.87	22.02
1	DAE + DNN-HMM(retrain)	13.81	13.89	14.69	14.03

We also performed evaluation experiments on clean speech. The word error rates for the clean versions of the evaluation set obtained with the baseline GMM-HMM systems are shown in rows 1 through 3 of Table 2. The results with the DNN-HMM system is shown in the fourth row. We see that the accuracies for clean speech deteriorate significantly with the GMM-HMMs trained using multi-condition data. Meanwhile, the results obtained by DNN-HMMs trained using multi-condition data was as good as those with the GMM-HMMs trained using clean data.

5.2. Denoising deep autoencoder[1]

The input and the target for the denoising autoencoder (DAE) were set to be the eleven-frame sequence of 40-channel log Mel-scale filterbank features with their delta and acceleration parameters. The DAE is fine-tuned using reverberant speech as the input and clean speech as the target. The input frames and the output frames for the training were adjusted to be time aligned in the multi-condition training data generation process. The last portions of reverberant speech utterance files exceeding the length of the clean speech were trimmed to equalize the lengths of input and output.

The autoencoder network has six layers in total consisting of three encoding layers and three decoding layers. The number of nodes in each layer is set to be 2,048 except for input and output layers. The network is initialized using the same RBMs as used for initializing the DNNs described in the last subsection which were trained using reverberant speech. The encoding layers were initialized using the weights of first three RBMs and the hidden unit biases. The decoding layers were initialized using the transpose of the weights mentioned above and the visible unit biases.

The fine tuning of the DAE was performed by error backpropagation with squared error as the loss function. The parameters such as the mini-batch size and the momentum are set to be the same as those for DNN training. However, initial learning rate was set to be 0.001, which is smaller than the one for DNN training.

The word error rates obtained using the DNN-HMM with the input enhanced by the DAE just described are shown in Table 1, row five. The accuracies were deteriorated with "SimData", "Room 1" by adding the DAE to the DNN-HMM system. However, we see that the accuracy is a little improved with "Room 2", near microphone condition and improved conspicuously with other more adverse conditions. These results suggest that the denoising autoencoders effectively reduce the influence of reverberation on the speech recognition accuracy.

For "RealData", the word accuracies are drastically improved for all conditions. Specifically, the word error was reduced by 4.4 points with "Near" microphone and 2.5 points with "Far" microphone compared to the adapted GMM-HMM system.

The results obtained using the combination of the DNN-HMM and the DAE with clean speech as input are shown in Table 2, row five. We see that the word accuracies are deteriorated drastically compared to the results just using the DNN-HMM system.

5.3. DNN-HMM trained with the DAE output

The speech feature parameters "enhanced" by the DAE are considered to have different characteristics from the original reverberant speech. Therefore, there is a possibility that the DNN-HMMs trained using the DAE output perform better than the DNN-HMMs trained using the multi-condition data, when the input speech is processed by the DAE. We retrained the DNN using the DAE output and performed speech recognition experiments. This time, the RBMs for

[1] We decided to join the challenge around the end of November 2013, two weeks and a few days before the result submission. Partly because of this limited time, the results in this section was not obtained before the submission deadline.

initializing the network were trained using the DAE output as training data.

The word error rates obtained using this retrained network are shown in Table 1, row six. We see that the deterioration of the word accuracies for "SimData", "Room 1" is somewhat ameliorated but improvement of the accuracies are limited for other conditions.

We also tested this system on clean speech input and the results are shown in Table 2, row six. Interestingly, word accuracies with clean speech were improved by retraining the DNN using the DAE-enhanced input, but was not as good as the condition when the input is not processed by the DAE (row four).

Overall, retraining of the DNN using the DAE-enhanced data was not effective and the combination of the DAE and the DNN trained using multi-condition data was more robust for severely reverberant speech.

6. CONCLUSION

In this paper, we proposed an approach to reverberant speech recognition adopting deep learning in front end as well as back end of the system and evaluated it through the ASR task (one channel) of Reverb Challenge 2014.

The DNN-HMM system trained on the multi-condition training set achieved a conspicuously higher word accuracy compared to the MLLR-adapted GMM-HMM system trained on the same data for all "SimData" conditions. Furthermore, feature enhancement with the deep autoencoder contributed to the improvement of recognition accuracy especially in the more adverse conditions. When the DNN-HMM was used without the deep autoencoder front end, it resulted in a better performance than the non-adapted GMM-HMM system, but was not as good as the adapted GMM-HMM system. However, it outperformed the adapted GMM-HMM system when combined with the deep autoencoder.

In this paper, due to a limited time for experiments, the autoencoder was initialized using the same set of RBMs as used for the DNN-HMM initialization. The input and output of the autoencoder was also defined to be the same set of feature parameters as the input for DNN-HMM. However, the network structure and the feature parameters for autoencoder may be optimized in some criterion to yield better results and we are looking at these issues as future work.

7. REFERENCES

[1] M.Gurelli and C.Nikias, "Evam: An eigenvector-based algorithm for multichannel blind deconvolution of input colored signals," *IEEE Trans. Audio, Speech & Language Process.*, vol. 43, no. 1, pp. 134–149, 1995.

[2] M.Delcroix, T.Hikichi, and M.Miyoshi, "On the use of lime dereverberation algorithm in an acoustic environment with a noise source," in *ICASSP*, 2006, vol. 1.

[3] S.Gannot and M.Moonen, "Subspace methods for multimicrophone speech dereverberation," in *EURASIP J.Appl.Signal Process.*, 2003, vol. 11, pp. 1074–1090.

[4] M.Wu and D.Wang, "A two-stage algorithm for one-microphone reverberant speech enhancement," *IEEE Trans. Audio, Speech & Language Process.*, vol. 14, no. 3, pp. 774–784, 2006.

[5] K.Kinoshita, M.Delcroix, T.Nakatani, and M.Miyoshi, "Suppression of late reverberation effect on speech signal using long-term multiple-step linear prediction," *IEEE Trans. Audio, Speech & Language Process.*, vol. 17, no. 4, pp. 534–545, 2009.

[6] R.Gomez and T.Kawahara, "Robust speech recognition based on dereverberation parameter optimization using acoustic model likelihood," *IEEE Trans. Audio, Speech & Language Process.*, vol. 18, no. 7, pp. 1708–1716, 2010.

[7] A.E.Rosenberg, C.H.Lee, and F.K.Soong, "Cepstral channel normalization techniques for hmm-based speaker verification," in *ICSLP*, 1994, pp. 1835–1838.

[8] C.J.Leggetter and P.C.Woodland, "Maximum likelihood linear regression for speaker adaptation of continuous density hidden markov models," in *Computer Speech and Language*, 1995, vol. 9, pp. 171–185.

[9] 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 Proc.*, vol. 20, no. 1, pp. 30–42, 2012.

[10] T.Ishii, H.Komiyama, T.Shinozaki, Y.Horiuchi, and S.Kuroiwa, "Reverberant speech recognition based on denoising autoencoder," in *INTERSPEECH*, 2013, pp. 3512–3516.

[11] X.Lu, Y.Tsao, S.Matsuda, and C.Hori, "Speech enhancement based on deep denoising autoencoder," in *INTERSPEECH*, 2013, pp. 436–440.

[12] K.Kinoshita, M.Delcroix, T.Yoshioka, T.Nakatani, E.Habets, R.Haeb-Umbach, V.Leutnant, A.Sehr, W.Kellermann, R.Maas, S.Gannot, and B.Raj, "The reverb challenge: A common evaluation framework for dereverberation and recognition of reverberant speech," in *Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA-13)*, 2013.

[13] C. M. Bishop, *Neural Networks for Pattern Recognition*, Oxford University Press, 1995.

[14] G.E.Hinton, S.Osindero, and Y.Teh, "A fast learning algorithm for deep belief nets," *Neural Computation*, vol. 18, pp. 1527–1554, 2006.

[15] G.E.Hinton, L.Deng, D.Yu, G.Dahl, A.Mohamed, N.Jaitly, A.Senior, V.Vanhoucke, P.Nguyen, T.Sainath, and B.Kingsbury, "Deep neural networks for acoustic modeling in speech recognition," *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 82–97, 2012.

[16] A.Mohamed, G.Dahl, and G.Hinton, "Acoustic modelling using deep belief networks," *IEEE Trans. Audio, Speech, & Language Proc.*, vol. 20, no. 1, pp. 14–22, 2012.

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

[18] N.Morgan, "Deep and wide: Multiple layers in automatic speech recognition," *IEEE Trans. Audio, Speech, & Language Proc.*, vol. 20, no. 1, pp. 7–13, 2012.

[19] G.S.V.S.Sivaram and H.Hermansky, "Sparse multilayer perceptron for phoneme recognition," *IEEE Trans. Audio, Speech, & Language Proc.*, vol. 20, no. 1, pp. 23–29, 2012.

[20] P.J.Bell, M.J.F.Gales, P.Lanchantin, X.Liu, Y.Long, S.Renals, P.Swietojanski, and P.C.Woodland, "Transcriptions of multi-genre media archives using out-of-domain data," in *Proc. SLT*, 2012, pp. 324–329.

[21] G.E.Hinton and R.R.Salakhutdinov, "Reducing the Dimensionality of Data with Neural Networks," *Science*, vol. 313, pp. 504–507, 2006.

[22] P.Vincent, H.Larochelle, Y.Bengio, and P.A.Manzagol, "Extracting and composing robust features with denoising autoencoders," in *Proceedings of the Twenty-fifth International Conference on Machine Learning (ICML'08)*, 2008, pp. 1096–1103.

[23] Y.Bengio, P.Lamblin, D.Popovici, and H.Larochelle, "Greedy layer-wise training of deep networks," in *in Advances in Neural Information Processing Systems 19 (NIPS06)*, 2007, pp. 153–160.