AN INVESTIGATION OF DEEP NEURAL NETWORKS FOR NOISE ROBUST SPEECH RECOGNITION

Michael L. Seltzer, Dong Yu

Microsoft Research
Redmond, WA 98052 USA
{mseltzer,dongyu}@microsoft.com

Yongqiang Wang[*]

Department of Engineering, Cambridge University
Cambridge, UK
yw293@eng.cam.ac.uk

ABSTRACT

Recently, a new acoustic model based on deep neural networks (DNN) has been introduced. While the DNN has generated significant improvements over GMM-based systems on several tasks, there has been no evaluation of the robustness of such systems to environmental distortion. In this paper, we investigate the noise robustness of DNN-based acoustic models and find that they can match state-of-the-art performance on the Aurora 4 task without any explicit noise compensation. This performance can be further improved by incorporating information about the environment into DNN training using a new method called noise-aware training. When combined with the recently proposed dropout training technique, a 7.5% relative improvement over the previously best published result on this task is achieved using only a single decoding pass and no additional decoding complexity compared to a standard DNN.

***Index Terms*—** noise robustness, deep neural network, adaptive training, Aurora 4

1. INTRODUCTION

Traditional speech recognition systems are derived from a HMM-based model of the speech production process in which each state is modeled by a Gaussian mixture model (GMM). These systems are sensitive to mismatch between the training and testing data, particularly the mismatch introduced by environmental noise. As a result, much effort has been spent improving the robustness of speech recognizers to such distortions.

Approaches to noise robustness generally fall into one of two approaches. Feature enhancement methods attempt to remove the corrupting noise from the observations prior to recognition. There are a tremendous number of algorithms that fall into this category, e.g. [1, 2]. Model adaptation methods leaves the observations unchanged and instead updates the model parameters of the recognizer to be more representative of the observed speech, e.g. [3, 4, 5]. Both of these approaches can be further improved by the use of multi-condition training data and adaptive training techniques. Both feature-space and model-space noise adaptive training methods have been proposed [6, 7, 8]. The combination of feature enhancement or model adaptation with adaptive training currently represents the state of the art in noise robustness.

Recently, a new form of acoustic model has been introduced based on deep neural networks (DNN). These acoustic models are closely related to the original ANN-HMM hybrid architecture [9] with two key differences. First, the networks are trained to predict tied context-dependent acoustic states called senones. Second, these networks have more layers than the networks trained in the past. While context-dependent deep neural networks (CD-DNN-HMM) have generated significant improvements over state of the art GMM-HMM systems on a variety of tasks [10, 11, 12], there has been no evaluation of the robustness of such systems to environmental distortion. Prior work in neural networks for noise robustness has primarily focused on tandem approaches which train neural networks to generate posterior features, e.g. [13, 14] and feature enhancement methods that use stereo data to train a network to map from noisy to clean features, e.g. [15, 16].

In this paper, we investigate the noise robustness performance of DNN-based acoustic models and propose three methods to improve accuracy. The first two methods can be considered DNN analogs to feature-space and model-space noise-adaptive training. These methods use information about the environmental distortion either via feature enhancement prior to network training or during network training itself. The third approach, called dropout training, is a recently proposed strategy for training neural networks on data sets where over-fitting is a concern [17]. While this method was not designed for noise robustness per se, we demonstrate that it is useful for noisy speech as it produces a network that is highly robust to variabilities in the input.

Through a series of experiments on the Aurora 4 task,we show that the DNN acoustic model has remarkable noise robustness, with comparable performance to several more complicated methods in the literature. By using the approaches proposed in this paper, performance is further improved, achieving the best published result on the Aurora 4 task. Unlike most robustness techniques for GMM-HMM acoustic models, the proposed methods do not add any decoding complexity and only require a single recognition pass.

The remainder of the paper is organized as follows. In Section 2 we review the DNN-HMM acoustic model. We then propose three strategies to improve noise robustness in Section 3. The performance of the proposed approaches are evaluated in Section 4 and finally, some conclusions are drawn in Section 5.

2. DEEP NEURAL NETWORKS

A deep neural network (DNN) is simply a multi-layer perceptron (MLP) with many hidden layers between its inputs and outputs. In this section, we review fundamental ideas of the MLP, discuss the benefits of pre-training, and show a neural network can be used as an acoustic model for speech recognition.

[*] The author performed the work while at Microsoft Research

978-1-4799-0356-6/13/$31.00 ©2013 IEEE

2.1. Multi-Layer Perceptrons

In this work, an MLP is used to classify an acoustic observation $\mathbf{x}$ into one of a set of context-dependent phonetic states s. It is a nonlinear classifier that can be interpreted as a stack of log-linear models. Each hidden layer models the posterior probabilities of a set of binary hidden variables $\mathbf{h}$ given the input visible variables $\mathbf{v}$, while the output layer models the class posterior probabilities. Thus, in each of the hidden layers, the posterior distribution can be expressed as

$$p(\mathbf{h}_l|\mathbf{v}_l) = \prod_{j=1}^{N^l} p(h_{l,j}|\mathbf{v}_l), \quad 0 \le l < L \tag{1}$$

where

$$p(h_{l,j}|\mathbf{v}_l) = \frac{1}{1+e^{(-z_{l,j}(\mathbf{v}_l))}}, \qquad z_{l,j} = \mathbf{w}_{l,j}^T\mathbf{v}_l + b_{l,j} \tag{2}$$

Each observation is propagated forward through the network, starting with the lowest layer $(\mathbf{v}_0 = \mathbf{x})$. The output variables of each layer become the input variables of the next layer, i.e. $\mathbf{v}_{l+1} = \mathbf{h}_l$. In the final layer, the class posterior probabilities are computed using a soft-max layer, defined as

$$p(s|\mathbf{x}) = p(s|\mathbf{v}_L) = \frac{e^{(z_{L,s}(\mathbf{v}_L))}}{\sum_{s'} e^{(z_{L,s'}(\mathbf{v}_L))}} \tag{3}$$

Note that the equality between $p(s|\mathbf{v}_L)$ and $p(s|\mathbf{x})$ is valid by making a mean-field approximation [18].

In this work, networks are trained by maximizing the log posterior probability over the training examples, which is equivalent to minimizing the cross-entropy.

$$\mathcal{L} = \sum_t \log p(s_t|\mathbf{x}_t) \tag{4}$$

The objective function is maximized using error back propagation which performs an efficient gradient-based update

$$(\mathbf{w}_{l,j}, b_{l,j}) \leftarrow (\mathbf{w}_{l,j}, b_{l,j}) + \eta \frac{\partial \mathcal{L}}{\partial(\mathbf{w}_{l,j}, b_{l,j})}, \quad \forall\, l, j \tag{5}$$

where η is the learning rate.

2.2. Pre-training DNNs

Performing back propagation training from a randomly initialized network can result in a poor local optimum, especially as the number of layers increases. To remedy this, pre-training methods have been proposed to better initialize the parameters prior to back propagation. The most well-known method of pre-training grows the network layer by layer in an unsupervised manner. This is done by treating each pair of layers in the network as a restricted Boltzmann machine (RBM) that can be trained using an objective criterion called contrastive divergence. Details about the pre-training algorithm can be found in [19]

2.3. Integrating DNN into the HMM

To perform speech recognition using a DNN, the state emission likelihoods generated by the GMMs are replaced with likelihoods generated by the DNN. These likelihoods are obtained via Bayes rule using the posterior probabilities computed by the DNN and the class priors.

$$p(\mathbf{x}|s) \propto \frac{p(s|\mathbf{x})}{p(s)} \tag{6}$$

Here the network is trained to predict context-dependent states, in the form of tied states or senones.

3. APPROACHES TO NOISE ROBUSTNESS FOR DNNS

In this paper, we explore four approaches to incorporating noise robustness into the training of DNNs. The first three of these mirror the main approaches used to improve robustness in conventional GMM-HMM recognizers [6]. These approaches are 1) training with multi-condition data, 2) using feature enhancement to remove the distortions in the observations prior to training, and 3) incorporating a noise model or noise estimate into the network itself. As we'll describe, the latter two methods are analogous to feature-space and model-space noise adaptive training, respectively. In addition to these approaches, we'll explore a method of training called dropout that generates networks that are more robust to unseen variabilities.

In this section, we denote the observed noisy features as $\mathbf{y}$, the corresponding unknown clean features as $\mathbf{x}$, and the corrupting noise as $\mathbf{n}$.

3.1. Training with multi-condition speech

Training a DNN on multi-condition data enables the network to learn higher level features that are more invariant to the effects of noise with respect to classification accuracy. In this case, we can view the deep neural network as a combination of nonlinear feature extractor and nonlinear classifier where the lower layers are implicitly seeking discriminative features that are invariant across the many acoustic conditions present in the training data.

Thus in DNN training with multi-condition data, the input vector $\mathbf{v}_t$ is simply an extended context window of the noisy observations.

$$\mathbf{v}_t = [\mathbf{y}_{t-\tau}, \ldots, \mathbf{y}_{t-1},\ \mathbf{y}_t,\ \mathbf{y}_{t+1}, \ldots, \mathbf{y}_{t+\tau}] \tag{7}$$

While multi-condition training is conceptually the same for DNNs and GMMs, there is a significant difference between the two. In the GMM-HMM, the features are directly modeled by a mixture of Gaussians, and thus, because the Gaussians simply model the observed data, they end up modeling the additional variability introduced by the additive noise. This can be mitigated by the use of discriminative training but only to a degree. In the case of discriminative training, features corrupted by noise are ignored by the GMMs whereas the DNN can potentially extract some useful information from them through the layers of nonlinear processing.

3.2. DNN training with enhanced features

One obvious way to reduce the variability in the features caused by environmental distortion is to attempt to remove it from the observations. Thus, the simplest way to reduce the effect of noise on the DNN is to simply process the data using a feature enhancement algorithm prior to training the network. By processing both the training and testing data with the same algorithm, any consistent errors or artifacts introduced by the enhancement can be learned by classifier. In the context of GMM-HMMs, this approach is referred to as feature-space noise adaptive training [20, 6] and this approach can be directly applied to DNN acoustic model. In contrast to (7), the input vector to the DNN is now formed from the enhanced features as

$$\mathbf{v}_t = [\hat{\mathbf{x}}_{t-\tau}, \ldots, \hat{\mathbf{x}}_{t-1},\ \hat{\mathbf{x}}_t,\ \hat{\mathbf{x}}_{t+1}, \ldots, \hat{\mathbf{x}}_{t+\tau}] \tag{8}$$

In this work, we use an feature enhancement algorithm based on the Cepstral-domain Minimum Mean Squared Error (C-MMSE) criterion [2]. This enhancement algorithm is based on the classic Log-MMSE noise suppression algorithm proposed by Ephraim and Malah [21]. The C-MMSE algorithm has been shown to consistently improve speech recognition performance of GMM-HMM recognizers in noisy conditions without causing degradations in high SNR conditions.

3.3. DNN noise-aware training

The other main approach to noise robustness for GMM-HMMs is model adaptation. In methods such as Vector Taylor Series (VTS) adaptation [22], an estimated noise model is used to adapt the Gaussian parameters of the recognizer based on a physical model that defines how noise corrupts clean speech. The relationship between the $\mathbf{x}$, $\mathbf{y}$, and $\mathbf{n}$ in the log spectral domain is typically approximated as

$$\mathbf{y} = \mathbf{x} + \log(1 + \exp(\mathbf{n} - \mathbf{x})) \tag{9}$$

One of the biggest challenges of noise robustness for speech recognition is dealing with the fact that the relationship in (9) is nonlinear. However, because the DNN is composed of multiple layers of nonlinear processing, the network has the capacity to learn this relationship directly from data. To enable this, we augment each observation input to the network with a estimate of the noise present in the signal. Because this is done in both training and decoding, this is analogous to noise adaptive training without an explicit mismatch function. Instead, the DNN is being given additional cues in order to automatically learn the relationship between noisy speech and noise in a way that is beneficial to predict senone posterior probabilities. Because the DNN is being informed about the noise, but not explicitly adapted, we adopt slightly different terminology and refer to this method as *noise-aware training*.

In this case the network's input vector is similar to (7) with a noise estimate appended.

$$\mathbf{v}_t = [\mathbf{y}_{t-\tau}, \ldots, \mathbf{y}_{t-1},\ \mathbf{y}_t,\ \mathbf{y}_{t+1}, \ldots, \mathbf{y}_{t+\tau},\ \hat{\mathbf{n}}_t] \tag{10}$$

In this work, we assume the noise is stationary and use a noise estimate that is fixed over the utterance, i.e. $\hat{\mathbf{n}}_t = \boldsymbol{\mu}_{\mathbf{n}}$.

3.4. DNN dropout training

One of the biggest problems in training DNNs is overfitting. This typically happens when a large DNN is trained using a relatively small training set. A training method called "dropout" has been recently proposed to alleviate this problem [17]. The basic idea of dropout is to randomly omit a certain percentage (e.g., α) of the neurons in each hidden layer during each presentation of the samples during training. In other words, each random combination of the (1- α) remaining hidden neurons needs to perform well even in the absence of the omitted neurons. This requires each neuron to depend less on other neurons. Since each higher-layer neuron gets input from a random collection of the lower-layer neurons, it receives noisier excitations. In this sense, dropout can be considered a technique that adds random noise to the training data. Dropout essentially reduces the capacity of the DNN and thus can improve the generalization of the resulting model. Note that when a hidden neuron is dropped out, its activation is set to 0 and so no error signal will pass through it. This means that other than the random dropout operation, no other changes to the training algorithm are needed to implement this feature.

At the test time, however, instead of using a random combination of the neurons at each hidden layer, we use the average of all the possible combinations. This can be easily accomplished by discounting all the weights involved in dropout training by (1- α) and use the resulted model as a normal DNN. Thus, dropout can also be interpreted as an efficient way of performing model averaging (similar to bagging) in the DNN framework.

Dropout was succesfully applied to TIMIT phoneme recognition in [17]. However, it has not yet been evaluated for word recognition, and in particular for word recognition in difficult environments.

4. EXPERIMENTS

To evaluate the speech recognition performance of the DNN-HMM, we performed a series of experiments on Aurora 4 [23]. Aurora 4 is a medium vocabulary task based on the Wall Street Journal (WSJ0) corpus.The experiments were performed with the 16 kHz multi-condition training set consisting of 7137 utterances from 83 speakers. One half of the utterances were recorded by the primary Sennheiser microphone and the other half were recorded using one of a number of different secondary microphones. Both halves include a combination of clean speech and speech corrupted by one of six different noises (street traffic, train station, car, babble, restaurant, airport) at 10-20 dB SNR.

The evaluation set is derived from WSJ0 5K-word closed-vocabulary test set which consists of 330 utterances from 8 speakers. This test set was recorded by the primary microphone and a secondary microphone. These two sets are then each corrupted by the same six noises used in the training set at 5-15 dB SNR, creating a total of 14 test sets. Notice that the types of noise are common across training and test sets but the SNRs of the data are not. These 14 test sets can then be grouped into 4 subsets: clean, noisy, clean with channel distortion, noisy with channel distortion, which will be referred to as A, B, C, and D, respectively.

The baseline GMM-HMM system consisted of context-dependent HMMs with 1206 senones and 16 Gaussians per state trained using maximum likelihood estimation. The input features were 39-dimensional MFCC features (static plus first and second order delta features) and cepstral mean normalization was performed. These models were also used to align the training data to create senone labels for training the DNN-HMM system. Decoding was performed with the task-standard WSJ0 bigram language model.

Two DNNs were trained using different input features: the same MFCC features used in the GMM-based system and the corresponding 24-dimensional log mel filterbank (FBANK) features. In both cases, utterance-level mean normalization was performed and first- and second-order derivative features were used. The input layer was formed from a context window of 11 frames creating an input layer of 429 visible units for the MFCC network and 792 visible units for the FBANK network. Both DNNs had 5 hidden layers with 2048 hidden units in each layer and the final soft-max output layer had 1206 units, corresponding to the senones of the HMM system. The networks were initialized using layer-by-layer generative pre-training and then discriminatively trained using twenty-five iterations of back propagation. A learning rate of 0.16 was used for the first 15 epochs and 0.004 for the remaining 10 epochs, with a momentum of 0.9. Back propagation was done using stochastic gradient descent in minibatches of 512 training examples.

The performance of these systems is shown in Table 1. As the results in the table indicate, the DNN produces substantial improvements in all test conditions compared to the baseline GMM-HMM system. In addition, further gains are achieved by using log mel filterbank features instead of cepstra. This is similar to the findings in [10].

Next, we examined the performance as a function of the number of senones and the number of hidden layers. The GMM-HMM system was retrained with a different state-tying threshold, resulting

System/ Features	A	B	C	D	AVG
GMM-HMM (MFCC)	12.5	18.3	20.5	31.9	23.0
DNN-HMM (MFCC)	5.7	10.4	10.9	22.6	15.3
DNN-HMM (FBANK-24)	5.0	9.2	9.0	20.6	13.8

Table 1. Comparison of WER (%) for GMM and DNN acoustic models on Aurora 4 using 1206 senones

in a system with 3202 senones. With this system, the WER of the GMM-HMM system decreased slightly from 23.0% to 22.5%. The performance of the DNN-HMM is shown in Table 2. Increasing the hidden layers resulted in reductions in WER until 9 hidden layers were used. At this point, a degradation in performance is observed as the network overfits to the training data. Similar to the GMM-HMM system, modest improvements are obtained by increasing the number of senones.

# of Senones	# of Hidden Layers			
	3	5	7	9
1206	14.2	13.8	13.7	13.9
3202	–	13.6	13.4	–

Table 2. WER (%) as a function of the number of senones and hidden layers

To evaluate the proposed techniques designed to increase the noise robustness of these systems, a series of experiments were performed using the 7-layer DNN with 3202 senones and FBANK features. We first evaluated the results of training and testing the DNN using features that have been preprocessed using the C-MMSE feature enhancement algorithm modified to operate in the log mel filterbank domain. In a second experiment, we evaluated the performance of proposed noise-aware training. The context window of features input to the DNN was augmented with an estimate of the noise. This noise estimate for each utterance was computed simply by averaging the first and last ten frames and fixed for the entire utterance. Finally, we evaluated the impact of dropout training on the performance of noise robustness. In this experiment, a dropout percentage of 20% was used and the original unprocessed multi-condition features were used as the input.

The results of these three experiments are shown in Table 3. The baseline performance for the 3202-senone DNN is shown for comparison. As the table indicates, feature enhancement improves performance on the clean speech test sets (A,C) but degrades performance on the noisy test sets (B,D). We conjecture that enhancing the features causes the network to be less robust to mismatched conditions, e.g. SNR or channel variations, because it sees fewer variations in the data during training. Incorporating the noise estimate into the network via noise-aware training reduces the WER from 13.4% to 13.1%. The use of dropout training provides a larger gain, dropping the WER to 12.9%. Finally, the best performance is obtained from the combination of noise-aware training and dropout. This results in an error rate of 12.4%, a 7.5% relative improvement.

System	A	B	C	D	AVG
DNN Baseline	5.6	8.8	8.9	20.0	13.4
DNN + FE	4.8	9.1	8.6	20.8	13.8
DNN + NAT	5.4	8.8	7.8	19.6	13.1
DNN + Dropout	5.1	8.4	8.6	19.3	12.9
DNN + NAT + Dropout	5.4	8.3	7.6	18.5	12.4

Table 3. A comparison of the WER (%) of DNN-HMM systems trained with feature enhancement (FE), noise-aware training (NAT), and dropout on Aurora 4. All networks have 7x2048 hidden layers and use 3202 senones.

Finally, in Table 4, the results obtained using the DNN-HMM are compared with several other systems in the literature. These systems are representative of the state of the art in acoustic modeling and adaptation for noise robustness and to the authors' knowledge, are the best published results on Aurora 4. The first system combines MPE discriminative training and noise adaptive training using VTS to compensate for noise and channel mismatch [24]. The second system uses hybrid generative/discriminative classifier [25]. An adaptively trained HMM with VTS adaptation is used to generate features based on state likelihoods and their derivatives. These features are then used in a discriminative log-linear model to obtain the final hypothesis. Finally, the VAT-Joint system is an adaptively trained HMM system and combines VTS adaptation for environment compensation and MLLR for speaker adaptation [26]. The last two rows of the table show the performance of the two DNN-HMM systems. The first system has no explicit noise compensation algorithm and is simply a direct application of the DNN-HMM. Nevertheless, it outperforms all but the VAT-Joint system. Finally, the DNN-HMM system with noise-aware training and dropout has the best performance. In addition, all the DNN-HMM results were obtained in the first pass, while the other three systems required two or more recognition passes for noise, channel, or speaker adaptation. These results clearly demonstrate the inherent robustness of the DNN to unwanted variability from noise and channel mismatch.

Systems	A	B	C	D	Avg.
MPE-VAT [24]	7.2	12.8	11.5	19.7	15.3
VAT+ deriv kernels [25]	7.4	12.6	10.7	19.0	14.8
VAT-Joint [26]	5.6	11.0	8.8	17.8	13.4
DNN (FBANK, 7x2048)	5.6	8.8	8.9	20.0	13.4
DNN + NAT + dropout	5.4	8.3	7.6	18.5	12.4

Table 4. WER (%) of several systems in the literature to the proposed DNN systems on Aurora 4.

5. CONCLUSION

In this paper, we have evaluated the performance of a DNN-based acoustic model for noise robust speech recognition. A DNN trained on multi-condition acoustic data without any explicit noise compensation achieves a level of performance equivalent to or better than the best published results on the Aurora 4 task. This is especially remarkable given that the DNN uses simple spectral-domain features and a simple frame-level objective function and only requires a single decoding pass. In contrast, the GMM-HMM state-of-the-art algorithms are far more complex, requiring multiple recognition passes and in some cases, multiple classifiers. We also introduced two methods, noise-aware training and dropout training, that further improved the performance of the DNN-HMM. Combining these two methods resulted in an improvement of 7.5% over the previously best published result without introducing any additional complexity compared to standard DNN decoding.

6. REFERENCES

[1] D. Macho, L. Mauuary, B. Noé, Y.M. Cheng, D. Ealey, D. Jouvet, H. Kelleher, D. Pearce, and F. Saadoun, "Evaluation of a noise-robust DSR front-end on Aurora databases," in *Proc. of ICSLP*, Denver, Colorado, 2002.

[2] D. Yu, L. Deng, J. Droppo, J. Wu, Y. Gong, and A. Acero, "A Minimum-mean-square-error noise reduction algorithm on mel-frequency cepstra for robust speech recognition," in *Proc. of ICASSP*, Las Vegas, NV, 2008.

[3] J. Li, L. Deng, D. Yu, Y. Gong, and A. Acero, "High-performance HMM adaptation with joint compensation of additive and convolutive distortions via Vector Taylor Series," in *Proc. of ASRU*, Kyoto, Japan, 2007.

[4] Y. Hu and Q. Huo, "An HMM compensation approach using unscented transformation for noisy speech recognition," in *Proc. ISCSLP*, 2006, pp. 346–357.

[5] M. L. Seltzer, K. Kalgaonkar, and A. Acero, "Acoustic model adaptation via linear spline interpolation for robust speech recognition," in *Proc. of ICASSP*, Dallas, TX, 2010.

[6] M. L. Seltzer, *Techniques for Noise Robustness in Automatic Speech Recognition*, chapter Acoustic Model Training for Robust Speech Recognition, John Wiley, 2013.

[7] 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, Nov. 2010.

[8] H. Liao and M. J. F. Gales, "Adaptive training with joint uncertainty decoding for robust recognition of noisy data," in *Proc. of ICASSP*, Honolulu, Hawaii, 2007.

[9] S. Renals, N. Morgan, H. Bourlard, M. Cohen, and H. Franco, "Connectionist probability estimators in HMM speech recognition," *IEEE Trans. Speech and Audio Proc.*, jan 1994.

[10] 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, jan. 2012.

[11] F. Seide, G. Li, and D. Yu, "Conversational speech transcription using context-dependent deep neural networks," in *Proc. Interspeech*, 2011.

[12] 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.

[13] O. Vinyals and S.V. Ravuri, "Comparing multilayer perceptron to deep belief network tandem features for robust asr," in *Proc. ICASSP*, may 2011, pp. 4596 –4599.

[14] S. Sharma, D. Ellis, S. Kajarekar, P. Jain, and H. Hermansky, "Feature extraction using non-linear transformation for robust speech recognition on the aurora database," in *Proc. ICASSP*, 2000, vol. 2, pp. II1117 –II1120 vol.2.

[15] S. Tamura and A. Waibel, "Noise reduction using connectionist models," in *Proc. ICASSP*, apr 1988, pp. 553 –556 vol.1.

[16] Andrew L. Maas, Quoc V. Le, Tyler M. ONeil, Oriol Vinyals, Patrick Nguyen, and Andrew Y. Ng, "Recurrent neural networks for noise reduction in robust ASR," in *Proc. Interspeech*, 2012.

[17] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov, "Improving neural networks by preventing co-adaptation of feature detectors," http://arxiv.org/abs/1207.0580, 2012.

[18] Lawrence Saul, Tommi Jaakkola, and Michael I. Jordan, "Mean field theory for sigmoid belief networks," *Journal of Artificial Intelligence Research*, vol. 4, pp. 61–76, 1996.

[19] G. Hinton, "A practical guide to training restricted boltzmann machines," Tech. Rep. UTML TR 2010-003, University of Toronto, 2010.

[20] Li Deng, A. Acero, Mike Plumpe, and Huang Xuedong, "Large-vocabulary speech recognition under adverse acoustic environments," *ICSLP 2000*, vol. 3, pp. 806–809, October 2000.

[21] Y. Ephraim and D. Malah, "Speech enhancement using a minimum mean square error log-spectral amplitude estimator," *IEEE Trans. on Acoust., Speech, Signal Processing*, vol. ASSP-33, no. 2, pp. 443–445, Apr. 1985.

[22] A. Acero, L. Deng, T. Kristjansson, and J. Zhang, "HMM Adaptation Using Vector Taylor Series for Noisy Speech Recognition," in *Proc. of ICSLP*, 2000.

[23] N. Parihar and J. Picone, "Aurora working group: DSR front end LVCSR evaluation AU/384/02," Tech. Rep., Inst. for Signal and Information Process, Mississippi State University.

[24] F. Flego and M. J. F. Gales, "Discriminative adaptive training with VTS and JUD," in *IEEE Workshop on Automatic Speech Recognition and Understanding*, 2009, pp. 170–175.

[25] A. Ragni and M. J. F. Gales, "Derivative kernels for noise robust ASR," in *IEEE Workshop on Automatic Speech Recognition and Understanding*, 2011.

[26] Y.-Q. Wang and M. J. F. Gales, "Speaker and noise factorisation for robust speech recognition," *IEEE transactions on audio speech and language processing*, vol. 20, no. 7, 2012.