ISCA Archive
http://www.isca-speech.org/archive

INTERSPEECH 2012
ISCA's 13th Annual Conference
Portland, OR, USA
September 9-13, 2012

Speech Restoration Based on Deep Learning Autoencoder with Layer-Wised Pretraining

Xugang Lu, Shigeki Matsuda, Chiori Hori, Hideki Kashioka

National Institute of Information and Communications Technology, Japan

Abstract

Neural network can be used to "remember" speech patterns by encoding speech statistical regularity in network parameters. Clean speech can be "recalled" when noisy speech is input to the network. Adding more hidden layers can increase network capacity. But when the hidden layer size increases (deep network), the network is easily to be trapped to a local solution when traditional training strategy is used. Therefore, the performance of using a deep network sometimes is even worse than using a shallow network. In this study, we explore the greedy layer-wised pretraining strategy to train a deep autoencoder (DAE) for speech restoration, and apply the restored speech for noisy robust speech recognition. The DAE is first pretrained using quasi-Newton optimization algorithm layer by layer in which each layer is regarded as a shallow autoencoder. And the output of the preceding layer is served as the input to the next layer. The pretrained layers are stacked and "unrolled" to be a DAE. The pretrained parameters are served as initial parameters of the DAE which are used to refine training. The trained DAE is used as a filter for speech restoration when noisy speech is given. Noisy robust speech recognition experiments were done to examine the performance of the trained deep network. Experimental results show that the DAE trained with pretraining process significantly improved the performance of speech restoration from noisy input.

Index Terms: Deep learning, autoencoder, noise reduction, speech recognition.

1. Introduction

Neural network can be trained to encode the statistical regularity of a training speech data set, thus to "remember" the trained speech patterns. The trained network can be used to restore a clean pattern from a noisy input pattern. In this sense, the trained network can be regarded as a noise reduction filter which can be used for noise robust feature extraction for many speech applications, e.g, automatic speech recognition (ASR), hearing aids.

In order to efficiently encode speech information, it is believed that a deep network (with multiple layers) is preferred than a shallow network (with single or less layers) [1]. However, training a deep network is much more difficult than training a shallow network. Because the training process is easily to be trapped to a local solution. In this situation, the performance of a deep network may be worse than a shallow network.

Recently, several deep learning algorithms that efficiently train a deep neural net in machine learning field have been proposed [2, 3, 4]. Different from traditional training process which trains the network as a whole with randomly initialized parameters, the strategy in these algorithms is to try to pretrain the network layer by layer, and using the pretrained parameters as initial parameters to further refine the network. This strategy is proved to significantly improve the performance of a deep neural net. Besides many applications in image processing [1], the deep learning is successfully applied in speech field [5, 6]. With layer-wised pretraining and discriminative fine tuning, the deep neural net showed improved discriminative performance for speech recognition [5]. Similar idea was used as a deep autoencoder which showed efficient encoding ability than traditional vector quantization (VQ) [6].

Different from others work, in this study, we focus on whether the pretraining process in deep learning autoencoder can be helpful or not for speech restoration. And we apply the restored speech for noisy robust speech recognition. The remainder of this paper is organized as follows. Section 2 gives a brief introduction of the deep learning idea, mainly the introduction of the architecture of deep autoencoder and its learning strategy. Section 3 introduces the training process of a deep autoencoder with a clean speech data set. In Section 4, we carry out experiments to evaluate the deep autoencoder for speech reconstruction and noisy robust speech recognition, and compare the performance with traditional encoding algorithms. Discussions and conclusion are given in section 5.

2. Deep learning autoencoder

Restrict Boltzmann machine (RBM) is widely used as a building module for deep belief network (DBN). It can be efficiently trained by using the divergence contrast algorithm [2]. A closely related learning module, i.e., autoencoder, also can be used to build a DBN by greedy, layer-by-layer learning [3]. In RBM based learning, Gibbs sampling algorithm is used in optimization, and it is difficult to use traditional optimization algorithms. In order to easily apply traditional optimization algorithms, we adopt a simple autoencoder module to build a DBN. In this sense, the DBN is named as deep autoencoder (DAE).

A training process of DAE is shown in Fig. 1. The basic autoencoder module is shown in panel (a) of Fig. 1. This autoencoder is a simple one hidden layer association neural net that tries to learn an encoder to reconstruct the input. The output of the hidden layer and reconstruction of the input are:

$$\begin{aligned} \mathbf{y} &= f(\mathbf{W}, \mathbf{b}) = \sigma(\mathbf{W}\mathbf{x} + \mathbf{b}) \\ \hat{\mathbf{x}} &= g(\mathbf{W}, \mathbf{b}, \mathbf{c}) = \sigma\left(\mathbf{W}^T \sigma(\mathbf{W}\mathbf{x} + \mathbf{b}) + \mathbf{c}\right) \end{aligned} \quad (1)$$

where $\mathbf{W}$ is a matrix of network weighting coefficient, σ is a sigmoid or hyperbolic tangent function as neural response function, $\mathbf{b}$ and $\mathbf{c}$ are the vectors of biases of input and output layers, respectively. Although there are several definitions of the objective function in learning the autoencoder parameter $\mathbf{W}$ with either smoothness or sparsity constraint [1], in our preliminary study, the basic autoencoder form is chosen for simplicity.

This work is support by MASTAR project of National Institute of Information and Communications Technology, Japan

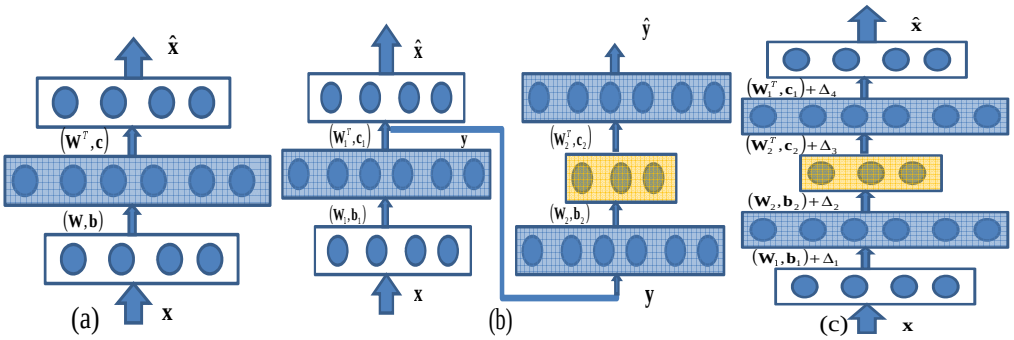

Figure 1: (a) Basic autoencoder module, (b) Two stacked autoencoder in pretraining stage, and (c) Deep autoencoder (two stacked autoencoders) in fine tuning stage.

The autoencoder parameter (weighting coefficient and bias) is learned based on minimizing an objective function defined as follows:

$$(\mathbf{W}^*, \mathbf{b}^*, \mathbf{c}^*) \stackrel{\Delta}{=} \arg\min_{\mathbf{W},\mathbf{b},\mathbf{c}} \sum_{\mathbf{x}} \|g(\mathbf{W}, \mathbf{b}, \mathbf{c}) - \mathbf{x}\|_2^2 \qquad (2)$$

The optimization of Eq. (2) can be solved by using many unconstrained optimization algorithms. In this study, a linear search based quasi-Newton optimization algorithm is used to estimate $(\mathbf{W}^*, \mathbf{b}^*, \mathbf{c}^*)$ [7]. For adding more hidden layers in pretraining, the input of next autoencoder is the output of the preceding hidden layer. Panel (b) in Fig. 1 shows an example of the pretraining of a two stacked autoencoder. After pretraining of each autoencoder layer by layer, all the layers are stacked to form a deep autoencoder for fine tuning as shown in panel (c) in Fig. 1. In fine tuning stage, the initial network parameters are fixed as the parameters obtained from pretraining stage. The parameter adjustment as $(\Delta_1, \Delta_2, \Delta_3, \Delta_4)$ are estimated based on traditional learning algorithms for neural network. Because the starting parameters for optimization of the DAE is set from pretraining of many shallow autoencoder, it is possible that the final solution is better than training the DAE with a random initialization.

3. Learning deep autoencoder for speech reconstruction

In order to recall clean speech pattern, the DAE must be trained from a clean speech data set. In our preliminary experiments, a clean speech data set consisting of 350 continuous speech utterances from ATR data corpus was used in training. And another 50 speech utterances was used in testing. The input of DAE is the Mel scale power spectrum patches which are concatenated to be a long vector from several consecutive frames. In order to explore the statistical regularity of the training data set by the DAE, we created a large Mel power spectral patches set from the training data samples (40000 was created in this study). The Mel power spectrum was extracted using 40 Mel filter bands. And the frame length is 16 ms with 8 ms frame shift. One spectral patch corresponds to 7 frames about 56 ms window length. Hence the dimension of the input to DAE is 40*7=280.

The architecture of a DAE is important for the final reconstruction performance, i.e., deep size of a DAE and layer size. However, there is no automatic method to optimally select the architecture. In this study, we manually set the architecture of a DAE as follows: in encoding stage, the size of input layer is 280, first hidden layer is 400, second hidden layer is 100, and the third hidden layer is 20. All the layers are stacked and unrolled to form a deep autoencoder (encoder-decoder), hence the corresponding autoencoder layer sizes are 280-400-100-20-100-400-280. Because the representation in final hidden layer of encoder is 20, the deep autoencoder also can be regarded as a dimension reduction encoder.

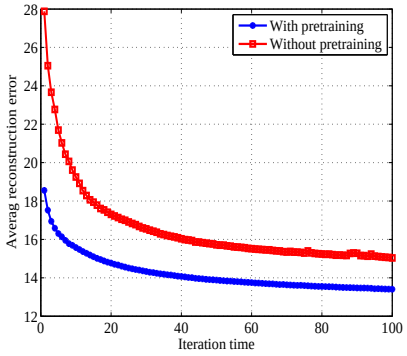

Figure 2: Learning convergence curves with and without pretraining stage.

In training of the DAE, a l-BFGS optimization algorithm [7], which is a batch based algorithm, was used in this study. In our experiments, a batch size of 1000 was used. In pretraining stage, the maximum number of iteration was set to 50 for the whole training data set. And in fine tuning stage, the maximum number of iteration was set to 100. The convergence curves of learning the DAE is shown in Fig. 2. From this figure, we can see that with pretraining, the average reconstruction error of the training data set (40000 spectral patches) from the DAE is reduced more than the training without pretraining.

4. Experiments and evaluations

In this section, we do experiments to test the performance of the DAE with and without pretraining for speech restoration from noisy speech input, an do noisy robust speech recognition experiments on the restored speech. Before examining the noisy robustness of the DAE representation, we first check how the DAE is used for speech reconstruction from clean speech input.

4.1. Qualitative analysis of reconstruction accuracy for clean speech

The purpose of learning the DAE is to reconstruct the clean speech. Therefore, the learned DAE should encode clean speech structure well, and the reconstruction of the input should be with high accuracy. An example of reconstruction of a clean

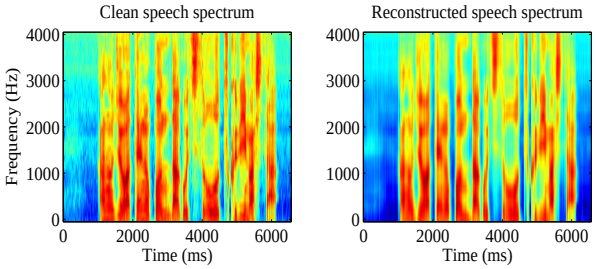

Figure 3: Clean speech spectrum (left), and reconstruction of it from trained DAE (right).

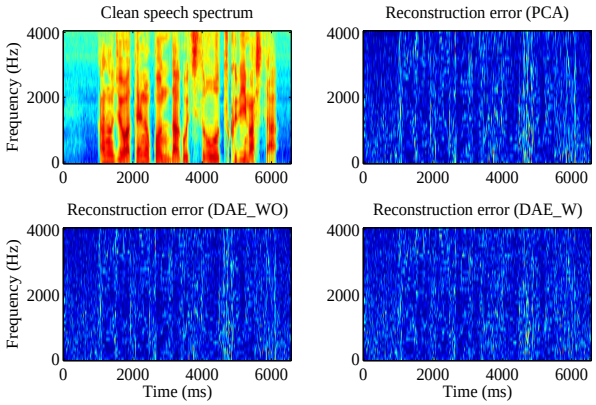

Figure 4: Clean speech spectrum (upper-left), and reconstruction error from PCA (upper-right), from DAE_WO (lower-left), from DAE_W (lower-right).

speech utterance is shown in Fig. 3. From this figure, we can see that most of the speech regularity structure is reconstructed well.

Now, we examine the reconstruction error of DAE without pretraining (DAE_WO) and with pretraining (DAE_W). In addition, we also provide the principal component analysis (PCA) based reconstruction error for comparison [8]. Because the dimension of final hidden layer in encoding stage of DAE is 20, the same number of principal component vectors are used in PCA reconstruction. The results of reconstruction error are shown in Fig. 4. From this figure, we can see that there leaves only random noise or no regular structure in the reconstruction residual signal (DAE_WO and DAE_W). However, in PCA based reconstruction, we can see a few temporal modulation structure in the residual signal. This suggest that the trained DAE can be used well to "remember" speech regularity structure.

4.2. Restoration of clean speech from noisy input

As we have discussed in section 1, after the DAE is trained on a clean speech data set, a clean speech pattern is expected to be recalled even when a noisy speech is input to the DAE. We show an example of reconstruction of a clean speech from noisy input in Fig. 5. In this figure, the power spectrum of a noisy speech (white random noise with SNR 5dB) is served as input to the DAE (left panel), the recalled power spectrum is shown in the right panel. From this figure, we can see that the trained DAE can be used as a filter for noise reduction.

We have showed that the DAE with pretraining can give more accurate reconstruction for clean input than that without

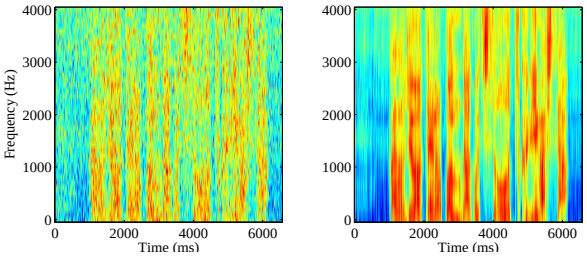

Figure 5: Noisy speech spectrum (left panel), and reconstruction from from DAE_W (right panel).

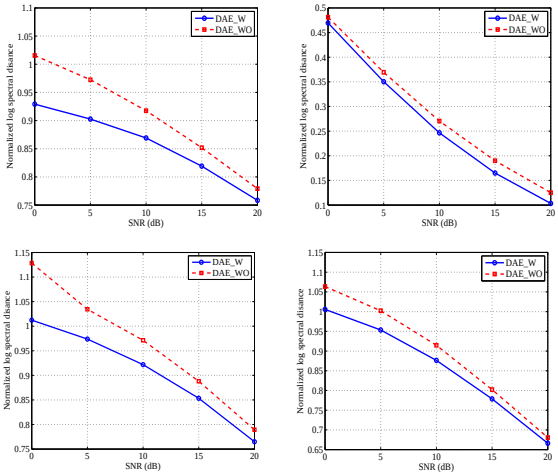

Figure 6: Average normalized log spectral distance in noisy conditions of white noise (upper-left), car noise (upper-right), factory noise (lower-left), and babble noise (lower-right).

pretraining, we now do experiments to check whether the pretraining stage can help to reduce noise or not. Given a clean testing data set $\{\mathbf{x}^i_{\text{clean}}\}$ composed of N speech utterances with $i = 1, 2, ..., N$. The corresponding noisy set is $\{\mathbf{y}^i_{\text{noisy}}\}$. The reconstruction of clean speech from clean and noisy input are obtained through the trained DAE as:

$$\begin{aligned} \hat{\mathbf{x}}^i_{\text{clean}} &= \text{DAE}(\mathbf{x}^i_{\text{clean}}) \\ \hat{\mathbf{y}}^i_{\text{noisy}} &= \text{DAE}(\mathbf{y}^i_{\text{noisy}}) \end{aligned} \tag{3}$$

In order to measure the similarity of the reconstructed speech from clean and noisy input, we define a normalized log spectral distance as:

$$\text{Dist} \triangleq \sum_i \frac{\left\|\hat{\mathbf{x}}^i_{\text{clean}} - \hat{\mathbf{y}}^i_{\text{noisy}}\right\|_2^2}{\left\|\hat{\mathbf{x}}^i_{\text{clean}}\right\|_2^2} \tag{4}$$

The smaller the value is, the better noise reduction of the DAE. In this study, four types of noise, white noise, car noise, factory noise, and babble noise from NOISEX-92, are artificially added to the clean test speech data with various SNRs as 0 dB, 5 dB, 10 dB, 15 dB, and 20 dB. For each utterance, we obtain an output pair of the reconstruction of clean and noisy input to the DAE, i.e., the recalls of clean speech from clean and noisy input. Then the average of normalized log spectral distance is calculated. The results in four types of noise conditions are shown in Fig. 6. From these figures, we can see that, with pretraining in DAE, the encoding is more robust to noise interference. From these results, we expect a noisy robust speech

recognition performance from using pretrained DAE for speech feature reconstruction.

4.3. Noisy robust speech recognition

We carry out speech recognition experiments to test the performance of the noisy robustness of the DAE. It is a simple Japanese phone recognition task. Each phone is modeled using a three state hidden Markov model (HMM), and 5 gaussian mixture modeling (GMM) for each state output probability distribution. 350 clean utterances were used in training, and 50 utterances (under various noisy conditions) were used for recognition (the same training and testing data sets as used in section 3). Mel frequency cepstral coefficient (MFCC) feature is used in HMM modeling. Rather than directly using the Mel power spectrum from Mel filter band output for MFCC extraction, we use the reconstructed Mel power spectrum from pretrained DAE for both training and testing. The same four noisy conditions as used in section 4.2 were simulated for producing the noisy inputs to DAE. The baseline feature is from the discrete cosine transform (DCT) of Mel power spectrum which is the MFCC as used in traditional ASR. The DCT based feature can be regarded as projecting the Mel power spectrum on a set of fixed basis vectors. For comparison, a learned basis vector set, the principal component analysis (PCA) based reconstruction (20 principal components) were also applied. The experimental results (phone recognition accuracy) are shown in table 1 for white noise condition, table 2 for car noise condition, table 3 for factory noise condition, and table 4 for babble noise condition, respectively. In these four tables, the best performance

Table 1: Recognition rates in white noise condition (%)

SNR	DCT	PCA	DAE_WO	DAE_W
0 dB	8.86	13.51	1.51	**14.80**
5 dB	8.91	14.75	9.56	**18.15**
10 dB	11.67	15.45	15.07	**22.10**
15 dB	22.20	22.46	24.42	**31.93**
20 dB	33.93	31.66	37.66	**44.30**

Table 2: Recognition rates in car noise condition (%)

SNR	DCT	PCA	DAE_WO	DAE_W
0 dB	71.75	76.12	72.29	**76.39**
5 dB	79.85	81.52	77.09	**82.82**
10 dB	83.79	84.17	80.77	**84.36**
15 dB	85.20	84.82	83.20	**85.25**
20 dB	85.52	85.52	83.74	**85.68**

Table 3: Recognition rates in factory noise condition (%)

SNR	DCT	PCA	DAE_WO	DAE_W
0 dB	10.97	10.97	3.08	**16.64**
5 dB	19.45	20.26	9.21	**21.07**
10 dB	31.44	**31.66**	12.97	31.01
15 dB	40.09	**41.06**	26.04	37.06
20 dB	49.27	49.86	44.03	**52.57**

values are marked in bold. From these tables, we can see that the performance of DAE with pretraining is better than that of without pretraining in almost all noisy conditions. Compared with DCT and PCA, the DAE with pretraining performed the best in white and car noise conditions. However, in factory and babble noise conditions, the DAE even with pretraining does not always perform well. Particularly in babble noise condition, it seems that the transform based on learned basis vectors performed worse than based on fixed basis vectors (such as DCT).

Table 4: Recognition rates in babble noise condition (%)

SNR	DCT	PCA	DAE_WO	DAE_W
0 dB	23.99	**24.37**	8.59	15.34
5 dB	**27.28**	26.63	8.54	14.05
10 dB	**34.31**	31.60	17.45	15.78
15 dB	**41.71**	39.01	26.36	31.55
20 dB	53.43	51.00	42.95	**55.48**

Our explanation is that babble noise has similar statistical regularity structure as that of speech. The learned vectors do not have discriminability to babble noise and speech.

5. Conclusion and discussions

Deep learning has been successfully used in speech signal processing and recognition. It can be used to improve speech recognition accuracy and encoding accuracy. In this study, we examined the "recall" ability of DAE for both clean and noisy inputs. Our results showed the pretraining of DAE helped to improve the "recall" robustness. In this sense the DAE can be used for noisy robust speech feature extraction.

Several issues need to be further investigated. In theoretical level, we need to think about what are good objective functions for building an autoencoder. In this study, we only tried the square loss of reconstruction error. Several constraints can be added in the objection functions, such as smoothness constraint, sparseness constraint [1, 9]. In practical level, in this study, for saving computation time, we only used small data sets to train and test the DAE. For further confirming the conclusion, a large training and testing data sets should be used. In addition, for simplicity, the architecture of the DAE is manually set without examining an optimal or suboptimal architecture, such as how many layers should be used, and what is the size of each layer. All these issues remain as our future work.

6. References

[1] Bengio, Y., "Learning deep architectures for AI," Foundations and Trends in Machine Learning, 2(1): 1-127, 2009.

[2] Hinton, G. E., and Salakhutdinov, R., "Reducing the Dimensionality of Data with Neural Networks," Science, 313: 504-507, 2006.

[3] Bengio, Y., Lamblin, P., Popovici, D., and Larochelle, H., "Greedy layer-wise training of deep networks," In Advances in Neural Information Processing Systems, 19: 153-160, MIT Press, Cambridge, 2007.

[4] Ranzato, M. A., Huang, F. J., Boureau, Y. L., LeCun, Y., "Unsupervised Learning of Invariant Feature Hierarchies with Applications to Object Recognition," IEEE conference on Computer Vision and Pattern Recognition, 1-8, 2007.

[5] Dahl, G., Yu, D., Deng, L., Acero, A., "Context-dependent pretrained deep neural networks for large vocabulary speech recognition," IEEE Transactions on Audio, Speech, and Language Processing, 20 (1): 30-42, 2011.

[6] Deng, L., Seltzer, M., Yu, D., Acero, A., Mohamed, A., Hinton, A., "Binary Coding of Speech Spectrograms Using a Deep Autoencoder," in Proc. of Interspeech, 1692-1695, 2010.

[7] Schmidt, M., Van Den Berg, E., Friedl, M. P., Murphy, K., "Optimizing costly functions with simple constraints: A limited-memory projected quasi-newton algorithm," in Proc. of Conf. on Artificial Intelligence and Statistics, 456-463, 2009.

[8] Loizou, P. C., Speech Enhancement: Theory and Practice, CRC Press, 2007.

[9] Lee, H., Ekanadham, C., and Ng, A. Y., "Sparse deep belief net model for visual area V2," in Advances in Neural Information Processing Systems (NIPS), 20: 873-880, 2008.