Sequence Error (SE) Minimization Training of Neural Network for Voice Conversion

Feng-Long Xie[1,2*], *Yao Qian*[2], *Yuchen Fan*[2], *Frank K. Soong*[2], *Haifeng Li*[1]

[1]Harbin Institute of Technology, Harbin, China
[2]Microsoft Research Asia, Beijing, China
{v-fxie,yaoqian,frankkps}@microsoft.com, lihaifeng@hit.edu.cn

10.21437/Interspeech.2014-448

Abstract

Neural network (NN) based voice conversion, which employs a nonlinear function to map the features from a source to a target speaker, has been shown to outperform GMM-based voice conversion approach [4-7]. However, there are still limitations to be overcome in NN-based voice conversion, e.g. NN is trained on a Frame Error (FE) minimization criterion and the corresponding weights are adjusted to minimize the error squares over the whole source-target, stereo training data set. In this paper, we use the idea of sentence optimization based, minimum generation error (MGE) training in HMM-based TTS synthesis, and modify the FE minimization to Sequence Error (SE) minimization in NN training for voice conversion. The conversion error over a training sentence from a source speaker to a target speaker is minimized via a gradient descent-based, back propagation (BP) procedure. Experimental results show that the speech converted by the NN, which is first trained with frame error minimization and then refined with sequence error minimization, sounds subjectively better than the converted speech by NN trained with frame error minimization only. Scores on both naturalness and similarity to the target speaker are improved.

Index Terms: voice conversion, neural network, pre-training, sequence error minimization

1. Introduction

The purpose of voice conversion is to modify the speech of one speaker (source speaker) to make it sound like another target speaker. Many statistical methods for voice conversion have been studied[1, 3, 8, 9]. Among these approaches, the joint density Gaussian mixture model (JD-GMM)-based mapping method [2] and neural network (NN) based mapping method [3][4] are widely used. Although JD-GMM can effectively convert source speech to target speech with a decent quality, there still exists some over smoothing problem due to the statistical averaging in training the mean and covariance of Gaussian components. This averaging step removes some detailed information in training samples and muffles voice quality in the converted speech. Different with GMM-based approach where conversion is performed by maximizing the conditional probability calculated from a joint probability of source and target speaker's speech, NN based approach directly train the conditional probability which converts source speech to target speech. Besides, the conversion function of NN based approach is non-linear which might be able to simulate some non-linear function in speech production/perception. So NN based approach can achieve relatively better performance than GMM based approach [4]. Recently Restricted Bolzamann Machine (RBM), Conditional Restricted Boltzmann Machine (CRBM) and Deep Belief Nets (DBN) have been successfully applied to voice conversion. The Gaussian distribution in each mixture of GMM is replaced by an RBM, which can better capture the inter-dimensional and inter-speaker correlations within the joint spectral features than the conventional GMM based transformation method in [6]. Wu [7] directly adopts CRBM to investigate a robust transformation function since CRBM can perform linear and non-linear transformations simultaneously. Nakashika [5] uses DBN to build a high-order eigen space of the source/target speaker, where it is easier to convert source speech to target speech than in the traditional cepstrum space.

In this paper, motivated by the success of minimum generation error training in HMM based speech synthesis [10] and the Deep Neural Network for parametric TTS [14][20] , we propose a "sequence error" minimization training for NN based voice conversion. By incorporating the parameter trajectory generation into the whole training procedure, the inconsistency between training and conversion is removed, and the constraints between static and dynamic features are also considered in NN training. With the definition of sequence error between target speech and converted speech, back propagation (BP) [3] is applied to minimize the sequence error until the error function is converged.

Pre-training was found useful to initialize the weight matrix of neural network recently. In this paper we adopt two kinds of pre-training methods. i.e., layer-wise back propagation pre-training [11] and Deep Belief Network (DBN) pre-training[12]. In layer-wise back propagation pre-training procedure, we train a one-hidden-layer neural network to full convergence with back propagation at first, and then the weights of first layer are fixed and the output layer is replaced by a new randomly initialized hidden layer and output layer. The deeper network is then fine tuned with back propagation. This procedure is repeated until the desired number of hidden layer is reached. However the DBN pre-training procedure treats each consecutive pair of layers in the network as a restricted Boltzmann machine (RBM). And the RBM parameters can be efficiently trained in an unsupervised fashion with the approximate contrastive divergence algorithm [13].

2. Neural Network Based Voice Conversion System

A block diagram of Neural Network (NN) based voice conversion system is shown in Figure 1, where both training and converting phases are illustrated.

In the training stage, a parallel database, which contains

*Work performed as an intern in the Speech Group, Microsoft Research Asia

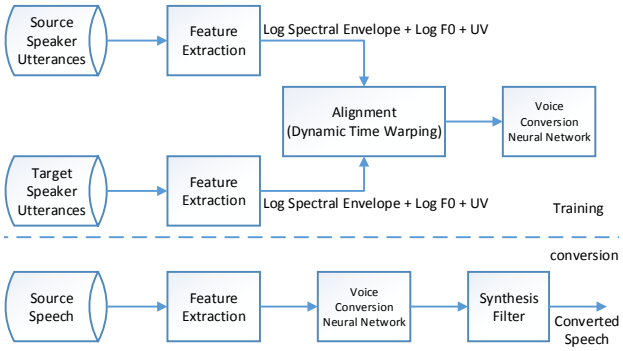

Figure 1: Neural Network based voice conversion system

source and target speakers' recordings of the same set of sentences, is needed. The feature extraction module is applied to represent the speech signals by vocal tract features,e.g., spectral envelope, and vocal fold features, e.g., fundamental frequency (F0) and unvoiced/voiced flag. The durations of the parallel utterances between source and target speakers are usually different, so dynamic time warping (DTW), a kind of dynamic programming, is used to align the feature vectors of the source and the target speakers. After alignment, each source feature vector is mapped to a target feature vector. Let $\boldsymbol{X}_t$ and $\boldsymbol{Y}_t$ denote the source and target feature vectors at frame t, respectively. A multi-layer feed forward neural network is used to train the mapping functions between the source and target feature vectors. For a 3-layer (with 2 hidden layers) neural network, the mapping function is

$$\widetilde{\boldsymbol{Y}_t} = F(\boldsymbol{X}_t) = \widetilde{f}(\boldsymbol{w}^{(3)} f(\boldsymbol{w}^{(2)} f(\boldsymbol{w}^{(1)} \boldsymbol{X}_t))) \tag{1}$$

where

$$\widetilde{f}(\theta) = \theta, f(\theta) = \frac{1}{1+e^{-\theta}} \tag{2}$$

$\boldsymbol{w}^1, \boldsymbol{w}^2, \boldsymbol{w}^3$ represents the weight matrices of the first, second, and third layer of the neural network, respectively. Sigmoid function is used as the activation function for two hidden layers, while linear activation function is employed at the final output layer. To capture the temporal change information, dynamic features, i.e., first and second order time derivatives, are usually appended to feature vectors. Let $\boldsymbol{X}$ and $\boldsymbol{Y}$ denote the source and target feature vector sequence, where $\boldsymbol{X} = [\boldsymbol{x}, \Delta\boldsymbol{x}, \Delta^2\boldsymbol{x}]$ and $\boldsymbol{Y} = [\boldsymbol{y}, \Delta\boldsymbol{y}, \Delta^2\boldsymbol{y}]$; $\boldsymbol{x}$, $\Delta\boldsymbol{x}$, $\Delta^2\boldsymbol{x}$ and $\boldsymbol{y}$, $\Delta\boldsymbol{y}$, $\Delta^2\boldsymbol{y}$ are the static, delta, and delta-delta coefficients sequence for the source and target, respectively. $\boldsymbol{x} = [\boldsymbol{x}_1^T, \boldsymbol{x}_2^T, ..., \boldsymbol{x}_t^T]$, $\boldsymbol{y} = [\boldsymbol{y}_1^T, \boldsymbol{y}_2^T, ..., \boldsymbol{y}_t^T]$. $\boldsymbol{X}$ and $\boldsymbol{Y}$ can be calculated from $\boldsymbol{x}$ and $\boldsymbol{y}$ through dynamic feature coefficient matrix $\boldsymbol{M}$[15].

$$\boldsymbol{X} = \boldsymbol{M}\boldsymbol{x}, \boldsymbol{Y} = \boldsymbol{M}\boldsymbol{y} \tag{3}$$

In NN training, all weights are trained by optimizing a cost function, i.e.,the average frame error (FE), between target feature vectors and the predicted output vectors with the Back-Propagation (BP) procedure [3]. The cost function is defined as frame error by

$$D(\boldsymbol{Y}, \widetilde{\boldsymbol{Y}}) = \frac{1}{2T} \sum_t \| \boldsymbol{Y}_t - \widetilde{\boldsymbol{Y}_t} \|^2 \tag{4}$$

Where $\widetilde{\boldsymbol{Y}}$ indicates the converted feature vector sequence.

The NN is trained frame by frame with a mini-batch based stochastic gradient descent algorithm [16],

$$\boldsymbol{w}_{i,j}^k = \boldsymbol{w}_{i,j}^k - \rho \cdot \Delta\boldsymbol{w}_{i,j}^k \tag{5}$$

where $\boldsymbol{w}_{i,j}^k$ is the weight between j-th neural node of $(k-1)$-th layer and i-th neural node of k-th layer, ρ is the learning rate.

In conversion stage, after feature extraction, the feature vector sequence of the source speaker $\boldsymbol{X}$ is firstly converted to the feature vector sequence $\widetilde{\boldsymbol{Y}}$ with the well trained neural network model, and then the smooth acoustic parameter $\widetilde{\boldsymbol{y}}$ is generated by parameter generation algorithm with delta constrains[10],

$$\widetilde{\boldsymbol{y}} = (\boldsymbol{M}^T\boldsymbol{U}^{-1}\boldsymbol{M})^{-1}\boldsymbol{M}^T\boldsymbol{U}^{-1}\widetilde{\boldsymbol{Y}} \tag{6}$$

where $\boldsymbol{M}$ and $\boldsymbol{U}$ are the dynamic feature coefficient matrix and the covariance matrix, respectively. We set the converted features, $\widetilde{\boldsymbol{Y}}$, from NN as mean vectors and pre-computed (global) variances of target features from all training data as $\boldsymbol{U}$. Finally the converted speech is synthesized by a vocoder with the converted spectral envelope and F0.

3. Minimum Sequence Error For Neural Network Based Voice Conversion

The goal of voice conversion is to convert the source speaker's speech to be as similar to the target speaker as possible without losing voice quality or naturalness of converted speech. The straight-forward cost function should be sequence error as

$$\boldsymbol{e} = D(\boldsymbol{y}, \widetilde{\boldsymbol{y}}) = \| \boldsymbol{y} - \widetilde{\boldsymbol{y}}) \|^2 = \sum_t \| \boldsymbol{y}_t - \widetilde{\boldsymbol{y}}_t \|^2 \tag{7}$$

instead of firstly using Eq. 4 and then generating $\widetilde{\boldsymbol{y}}$ by Eq. 6. The gradient in Eq. 5 should be

$$\Delta\boldsymbol{w}_{i,j}^k = \frac{\partial \boldsymbol{e}}{\partial \boldsymbol{w}_{i,j}^k} = \frac{\partial \boldsymbol{e}}{\partial \boldsymbol{P}_i^k} \cdot \frac{\partial \boldsymbol{P}_i^k}{\partial \boldsymbol{w}_{i,j}^k} \tag{8}$$

$$\boldsymbol{P}_i^k = \sum_j \boldsymbol{w}_{i,j}^k \cdot \boldsymbol{Q}_j^{k-1} \tag{9}$$

$\boldsymbol{Q}_j^{k-1}$ is the output of the j-th neural node in $(k-1)$-th layer and $\boldsymbol{P}_i^k$ is the input of the i-th node in k-th layer.

$$\frac{\partial \boldsymbol{P}_i^k}{\partial \boldsymbol{w}_{i,j}^k} = \frac{\sum_j \boldsymbol{w}_{i,j}^k \cdot \boldsymbol{Q}_j^{k-1}}{\partial \boldsymbol{w}_{i,j}^k} = \boldsymbol{Q}_j^{k-1} \tag{10}$$

Here we define

$$\boldsymbol{d}_i^k = \frac{\partial \boldsymbol{e}}{\partial \boldsymbol{P}_i^k} \tag{11}$$

For the output layer:

$$\begin{aligned} \boldsymbol{d}_i^k &= \frac{\partial \boldsymbol{e}}{\partial \boldsymbol{P}_i^k} = \left[\frac{\partial \boldsymbol{e}}{\partial \widetilde{\boldsymbol{Y}}}\right]_i = \left[\frac{\partial \boldsymbol{e}}{\partial \widetilde{\boldsymbol{y}}} \cdot \frac{\partial \widetilde{\boldsymbol{y}}}{\partial \widetilde{\boldsymbol{Y}}}\right]_i \\ &= \left[-2 \cdot (\boldsymbol{y} - \widetilde{\boldsymbol{y}})^T \left(\boldsymbol{M}^T\boldsymbol{U}^{-1}\boldsymbol{M}\right)^{-1} \boldsymbol{M}^T\boldsymbol{U}^{-1}\right]_i \end{aligned} \tag{12}$$

where $[\cdot]_i$ indicate the i-th dimension of vector.
For the hidden layer:

$$\begin{aligned} \boldsymbol{d}_i^k &= \frac{\partial \boldsymbol{e}}{\partial \boldsymbol{P}_i^k} = \frac{\partial \boldsymbol{e}}{\partial \boldsymbol{Q}_i^k} \cdot \frac{\partial \boldsymbol{Q}_i^k}{\partial \boldsymbol{P}_i^k} \\ &= \left(\sum_j \frac{\partial \boldsymbol{e}}{\partial \boldsymbol{P}_j^{k+1}} \cdot \frac{\partial \boldsymbol{P}_j^{k+1}}{\partial \boldsymbol{Q}_i^k}\right) \cdot \frac{f\left(\boldsymbol{P}_i^k\right)}{\partial \boldsymbol{P}_i^k} \\ &= \left(\sum_j \boldsymbol{d}_j^{k+1} \cdot \boldsymbol{w}_{j,i}^{k+1}\right) \cdot f\left(\boldsymbol{P}_i^k\right) \cdot \left(1 - f\left(\boldsymbol{P}_i^k\right)\right) \end{aligned} \tag{13}$$

where $f(\cdot)$ is sigmoid function defined in Eq. 2.

Mini-batch algorithms [16] are widely used in neural network training as a way to speed-up the stochastic convex optimization. However, it's not suitable for minimum sequence error (SE) based training where the size of batch for one iteration is equal to the length of one sentence instead of being fixed in a mini-batch [10].

Compared with minimum FE training, minimum SE training has a relatively larger computational cost because of the calculation of the inverse of $\boldsymbol{M^T U^{-1} M}$. Fortunately $\boldsymbol{M^T U^{-1} M}$ is a band matrix (diagonal covariance is adopted in our implementation), and its inverse can also be approximated to a block matrix which can significantly reduce the computational cost.

4. Experiments

4.1. Experimental Setup

Current voice conversion techniques need a parallel database. The research present here is carried out on the CMU ARCTIC database [17]. The ARCTIC corpus consists of four primary sets of recordings (2 male BDL and RMS, 2 female CLB and SLT), plus 3 other accents sets of recordings (3 male, Canadian JMK, Scottish AWB and Indian KSP). In our experiments, we do voice conversion in six pairs:

1) SLT (U.S. Female) to BDL (U.S. Male)
2) BDL (U.S. Male) to SLT (U.S. Female)
3) SLT (U.S. Female) to CLB (U.S. Female)
4) CLB (U.S. Female) to SLT (U.S. Female)
5) BDL (U.S. Male) to RMS (U.S. Male)
6) RMS (U.S. Male) to BDL (U.S. Male)

100 parallel utterances in CMU ARCTIC corpus were used for training, and 100 different parallel utterances in CMU ARCTIC corpus were used for testing. 256th-order log spectral envelopes calculated at 5ms frame shift with 512 point FFT and its delta and delta-delta coefficients were directly used as spectral features. Log F0 with delta and delta-delta and two contexts were used as the excitation features. Here a context size of two indicates the Log F0 from two left and two right adjacent frames were appended to Log F0 of the current frame. 1 dimension of UV was also added. So totally the number of dimension of the feature vector is 784. Adding contexts for F0 part can improve the pitch transformation result. But adding contexts for spectral part will lead to huge number of model parameters which will causes over-fitting problem with limited data in VC task seen from experiments. So in this paper we only add contexts for F0.

As shown in Figure 2, we investigate on different neural network structure and finally use a 3-layer neural network (2 hidden layers, each hidden layer contains 1,600 nodes) which achieves the best performance on the test set for all of the following experiments. The training samples were normalized to zero mean and unity variance before training. For Minimum SE based training method, the weight matrix of neural network was initialed with Minimum FE criterion until the frame error function was converged at first, and then we fine-tune the model with Minimum SE criterion until the sequence error function is converged. The learning rate is set as 0.01. 30 epochs were executed in model parameter estimation.

Both objective and subjective measures are used to evaluate voice conversion system. The converted speech is measured objectively in terms of distortions between natural speech of the

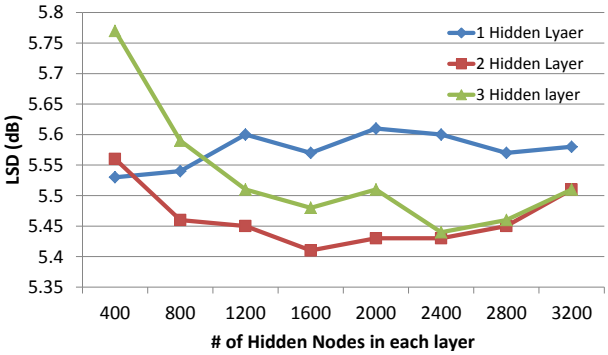

Figure 2: Neutral Network structure impact on LSD on test set

target speaker and the converted speech of the source speaker. Log spectral distance (LSD) in dB and root mean square error(RMSE) of F0 in Hz are used as the measurement. We also implement preference test on Amazon Mechanical Turk [18] to evaluate the naturalness and similarity between the converted speech and target speech.

4.2. Experimental Results and Analysis

4.2.1. NN vs. GMM

To make sure NN based VC can achieve relatively higher performance or at least as well as GMM and verify the effect of generation algorithm with delta features in VC task. We conduct a conversion comparison among NN based VC (with parameter generation), NN based VC(without parameter generation) in [4] and GMM based VC (with parameter generation) in [2]. 40th-order LSP with gain and its delta and delta-delta coefficients are used as spectral features here. All these three systems are trained with Minimum FE criterion. Table 1 shows the LSD on test set between converted speech and target speech of these three methods on transformation set SLT to BDL. NN_B can produce smoother parameter trajectory than NN_A due to that generation algorithm considers the relationships between static and dynamic features as constrains and results in the improvement of LSD.

Table 1: LSD comparison among NN_A, NN_B and GMM

	GMM	NN_A	NN_B
LSD (dB)	7.90	7.85	7.57

4.2.2. Spectral Feature Representation Analysis

LSP has the closet relevance to the vocal tract natural resonances or "formants" among all the LPC parameters and it has been widely used for representing spectral features [19]. But in this paper we adopt spectral envelope directly as spectral feature. A preference test between LSP based VC (LSP-VC) and spectral envelope based VC (SP-VC) was conducted. In SP-VC we use 256th-order log spectral envelope and its delta and delta-delta coefficients as spectral features (768-dim) while in LSP-VC we use 40th-order LSP with gain and its delta and delta-delta coefficients and two contexts (615-dim) as spectral features. Both of these two system are trained with Minimum FE based neural network until the error function is converged. A total of 10 native English speakers were asked to participate in this test and listen to 60 paired utterance (from SLT to BDL) synthesized by LSP-VC and SP-VC. The preference test result is shown in table 2. SP-VC achieves better performance than LSP-VC due to that spectral envelope has stronger inter-dimensional correlations than LSP. In the following experiments, we adopt log spectral envelope as spectral feature.

Table 2: naturalness preference score (%) between LSP-VC and SP-VC, where N/P denotes "No preference"; p, the p-value of t-test between the two systems

	LSP-VC	SP-VC	N/P	p
Naturalness	18	70	12	<0.001

4.2.3. *Pre-training*

Figure 3 shows the average LSDs of testing set with/without pre-training on the transformation set SLT to BDL with Minimum FE training criterion. Here no pre-training indicates we randomly initialize the weight matrix and fine-tune the network with back propagation. The LSD of of neural network with DBN pre-training is slightly better than that of neural network without pre-training mainly due to that DBN pre-training is an unsupervised learning procedure and only considers source speech information. And the LSD of neural network with layer-wise pre-training is relatively much better than that of neural network without pre-training because layer-wise pre-training is a supervised learning procedure and considers both source speech and target speech information. Both of these two pre-training methods can achieve a better initialization of weight matrix and lead to a better NN training result.

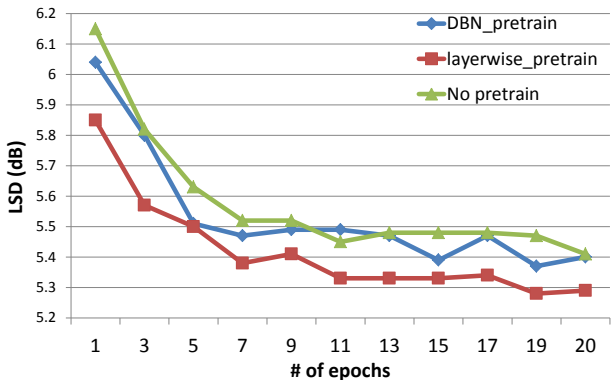

Figure 3: Pre-training (DBN or layer-wise BP) vs. no pre-training

4.2.4. *Convergence Property of Minimum SE Based Training*

Figure 4 shows the convergence property of Minimum SE based neural network training on transformation set from SLT to BDL.Seen from the results of the training and test set, the Minimum SE based Neural Network training is converged after 10∼15 iterations and the average sequence error of one dimension reduced about 11% after the Minimum SE based training.

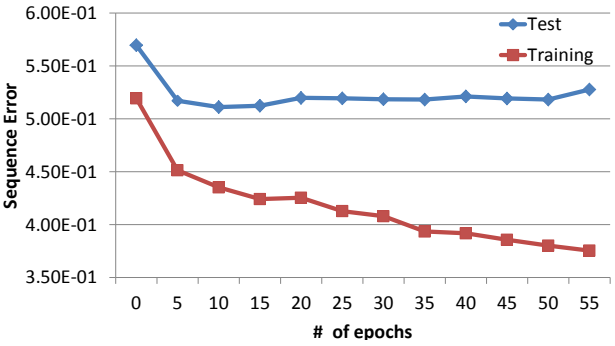

Figure 4: Convergence of Minimum SE based neural network training for voice conversion

4.2.5. *Minimum FE vs. Minimum SE*

Table 3 shows a LSD and RMSE of F0 comparison result between Minimum FE and Minimum SE based voice conversion system. On these six transformation sets Minimum SE based VC achieves better performance than Minimum FE based VC consistently. Minimum SE based NN training aims to minimize the sequence error instead of the error between NN output and target speech where the inconsistency between the training and converting is eliminated. And the constraints between static and dynamic features are also considered by incorporating the parameter generation in training procedure.

Table 3: LSD and RMSE of F0 comparison on test set between Minimum FE and Minimum SE based VC

Transformation sets	LSD (dB)		RMSE of F0 (Hz)	
	FE	SE	FE	SE
SLT to BDL	5.41	5.30	16.27	15.47
BDL to SLT	4.94	4.81	18.36	18.2
CLB to SLT	4.83	4.76	15.05	14.81
SLT to CLB	4.84	4.73	16.64	16.27
BDL to RMS	4.92	4.81	15.25	14.78
RMS to BDL	5.60	5.37	14.12	14.11

We conducted preference test for naturalness and similarity to evaluate the performance of the Minimum SE based transformation against the Minimum FE based transformation. A total of 20 native English speakers were asked to participate in the naturalness preference test and similarity preference test respectively. In naturalness preference test, each subject was asked to listen to 60 paired utterances (10 utterances for each transformation case) synthesized by Minimum FE and Minimum SE, and then choose the utterance which is more natural and intelligible. In similarity preference test, each subject was also asked to listen to 60 paired utterances synthesized by Minimum FE and Minimum SE, and then choose the utterance which is closer to the target speaker's natural utterance.

The naturalness and similarity preference results are shown in table 4 which indicates that Minimum SE based VC system is superior to Minimum FE based VC system in listening perception (Some samples of converted utterances are given on the web link: http://research.microsoft.com/en-us/projects/vcnn/default.aspx).

Table 4: naturalness and similarity preference score (%) between Minimum SE based and Minimum FE based Voice Conversion, where N/P denotes "No preference"; p, the p-value of t-test between two systems

	FE	SE	N/P	p
Naturalness	23	60	17	<0.001
Similarity	35	65	–	<0.001

5. Conclusions

In this paper, Minimum SE training criterion for NN based voice conversion was proposed and pre-training for neural network based voice conversion was studied. From the experimental results, the sequence error is reduced after Minimum SE based NN training. Two objective evaluation measures, LSD and RMSE of F0, both show that the proposed method Minimum SE based training outperforms the Minimum FE based training. The naturalness of the converted speech and the similarity between converted speech and target speech are both improved. DBN pre-training and layer-wise back propagation pre-training both help to better initialize the parameters of NN and lead to a better trained NN model.

6. References

[1] Y. Stylianou, O. Cappe, and E. Moulines, "Continuous probabilistic transform for voice conversion," IEEE Trans. on Audio Speech and Language Processing, vol. 6, no. 2, pp. 131-142, 1998.

[2] T. Toda, A. Black, and K. Tokuda,"Voice conversion based on maximum-likelihood estimation of spectral parameter trajectory," IEEE Trans. on Audio Speech and Language Processing, vol. 15, no. 8, pp. 2222-2235, 2007.

[3] M. Narendranath, H. A. Murthy, S. Rajendran, and B. Yegnanarayana, "Transformation of formants for voice conversion using artificial neural networks," Speech Commun., vol. 16, no. 2, pp. 207-216, 1995.

[4] S. Desai, A. Black, B. Yegnanarayana, K. Prahallad, "Spectral Mapping Using Artifical Neural Networks for Voice Conversion," IEEE Trans. on Audio Speech and Language Processing, vol. 18, no. 5, pp. 954-964, 2010.

[5] T. Nakashika, R. Takashima, T. Takiguchi, Y. Ariki, "Voice Conversion in High-order Eigen Space Using Deep Belief Nets," in Proc. Interspeech, pp. 369-372, 2013.

[6] L. Chen, Z. Ling, Y. Song, L. Dai, "Joint Spectral Distribution Modeling Using Restricted Boltzmann Machines for Voice Conversion," in Proc. Interspeech, pp. 3053-3056, 2013.

[7] Z. Wu, E. Chng, H. Li, "Conditional Restricted Boltzmann Machine For Voice Conversion," in Proc. ChinaSIP, pp. 104-108, 2013.

[8] H. Valbret, E. Moulines, and J. P. Tubach, "Voice transformation using PSOLA technique," in Proc. ICASPP, pp. 145-148, 1992

[9] M. Abe, S. Nakamura, K. Shikano, and H. Kuwabara, "Voice conversion through vector quantization," in Proc. ICASPP, pp. 655-658, 1988

[10] Y. Wu, R. Wang "Minimum Generation Error Training for HMM-Based Speech Synthesis" in Proc. ICASSP, pp. 89-92, 2006

[11] F. Seide, G. Li, X. Chen, D. Yu, "Feature engineering in Context-Dependent Deep Neural Networks for conversational speech transcription," IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU), pp. 24-29, 2011

[12] G. Hinton, L. Deng, D. Yu, G. Dahl, A. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. Sainath, B. Kingsbury, "Deep Neural Networks for Acoustic Modeling in Speech Recognition: The Shared Views of Four Research Groups," Signal Processing Magazine, IEEE , vol. 29, no. 6, pp. 82-97, Nov. 2012

[13] G. Hinton, S. Osindero, and Y. Teh, "A Fast Learning Algorithm for Deep Belief Nets", Neural Computation, vol. 18, pp. 1527-1554, 2006.

[14] Y. Qian, Y. Fan, W. Hu, F. K. Soong, "On the Training Aspects of Deep Neural Network (DNN) For Parametric TTS Synthesis", in Proc. ICASPP, 2014(Accepted).

[15] K. Tokuda, T. Yoshimura, T. Masuko, T. Kobayashi, T. Kitamura, "Speech Parameter Generation Algorithms For HMM-based Speech Synthesis," in Proc. ICASSP, pp. 1315-1318, 2000.

[16] M. Devetsikiotis, W. A. Al-Qaq, J. A. Freebersyser, J. K. Townsend, "Stochastic gradient techniques for the efficient simulation of high-speed networks using importance sampling," in Proc. GLOBECOM, pp. 751-756, 1993.

[17] J. Kominek and A. Black, "The CMU ARCTIC databases for speech synthesis," Tech. Rep. CMU-LTI-03-177, Language Technologies Institute, Carnegie Mellon University, 2003.

[18] I. McGraw, J. Glass and S. Seneff, "Growing a Spoken Language Interface on Amazon Mechanical Turk", in Proc. of Interspeech, 2011.

[19] J. H. Nirmal, S. Patnaik, Mukesh A. Zaveri, "Line Spectral Pairs Based Voice Conversion Using Radial Basis Function ", Int. J. on Signal & Image Processing, Vol. 4, No.2, pp. 26-33, 2013.

[20] Heiga Zen, Senior, A., Schuster, M., "Statistical parametric speech synthesis using deep neural networks," in Proc, ICASSP, pp. 7962-7966, 2013.