Constructing a Deep Neural Network Based Spectral Model for Statistical Speech Synthesis

Shinji Takaki and Junichi Yamagishi

Abstract This paper presents a technique for spectral modeling using a deep neural network (DNN) for statistical parametric speech synthesis. In statistical parametric speech synthesis systems, spectrum is generally represented by low-dimensional spectral envelope parameters such as cepstrum and LSP, and the parameters are statistically modeled using hidden Markov models (HMMs) or DNNs. In this paper, we propose a statistical parametric speech synthesis system that models high-dimensional spectral amplitudes directly using the DNN framework to improve modelling of spectral fine structures. We combine two DNNs, i.e. one for data-driven feature extraction from the spectral amplitudes pre-trained using an auto-encoder and another for acoustic modeling into a large network and optimize the networks together to construct a single DNN that directly synthesizes spectral amplitude information from linguistic features. Experimental results show that the proposed technique increases the quality of synthetic speech.

1 Introduction

Recently, deep neural networks (DNNs) with many hidden layers have been significantly improved in statistical speech synthesis researches. For instance, DNNs have been applied for acoustic modelling. Zen et al. [1] use DNN to learn the relationship between input texts and extracted features instead of decision tree-based

Shinji Takaki was supported in part by NAVER Labs.
Junichi Yamagishi—The research leading to these results was partly funded by EP/J002526/1 (CAF).

S. Takaki (✉) · J. Yamagishi
National Institute of Informatics, Tokyo, Japan
e-mail: takaki@nii.ac.jp

J. Yamagishi
The Centre for Speech Technology Research, University of Edinburgh, Edinburgh, UK
e-mail: jyamagis@nii.ac.jp

A. Esposito et al. (eds.), *Recent Advances in Nonlinear Speech Processing*,
Smart Innovation, Systems and Technologies 48,
DOI 10.1007/978-3-319-28109-4_12

state tying. Restricted Boltzmann machines or deep belief networks have been used to model output probabilities of hidden Markov model (HMM) states instead of GMMs [2]. Recurrent neural network and long-short term memory have been used for prosody modelling [3] and acoustic trajectory modelling [4]. In addition, an auto-encoder neural network has also been used to extract low dimensional excitation parameters [5].

However, the synthetic speech of the latest statistical parametric speech synthesis still sounds muffled, and averaging effects of statistical models are often said to remove spectral fine structure of natural speech. To improve the quality of synthetic speech, a stochastic postfilter approach has been proposed [6] where a DNN is used to model the conditional probability of the spectral differences between natural and synthetic speech. The approach was found to be able to reconstruct the spectral fine structure lost during modeling and has significantly improved the quality for synthetic speech [6]. In this experiment, the acoustic model was trained using lower dimensional spectral envelope features, while the DNN-based postfiler was trained using the spectral amplitudes obtained using the STRAIGHT vocoder [7]. From the experimental findings, we can hypothesize that the current statistical parametric speech synthesis may suffer from quality loss due to not only statistical averaging but also acoustic modeling using lower dimensional acoustic features.

On the basis of this hypothesis, in this paper we present a new technique for constructing a DNN that directly synthesizes spectral amplitudes from linguistic features without using spectral envelope parameters such as mel-cepstrum. It is well known that there are many problems for training a DNN such as the local optima, vanishing gradients and so on [8]. However, it has been reported in the ASR field that DNNs that deal with high-dimensional features, e.g. FFT frequency spectrum, can be appropriately constructed using an efficient training technique such as pre-training [9].

Thus, in this paper we propose an efficient training technique for constructing a DNN that directly synthesizes spectral amplitudes from input texts. A key idea is to stack two DNNs, an auto-encoder neural network for data-driven nonlinear feature extraction from the spectral amplitudes and another network for acoustic modeling and context clustering. The proposed technique is regarded as a function-wise pre-training technique for constructing the DNN-based speech synthesis system.

The rest of this paper is organized as follows. Section 2 reviews a DNN-based acoustic model for the statistical parametric speech synthesis. Section 3 describes a DNN-based acoustic feature extractor and spectrum re-generator. Section 4 explains the proposed technique for constructing a DNN that directly synthesizes the spectral amplitudes. The experimental conditions and results are shown in Sect. 5. Concluding remarks and future works are presented in Sect. 6.

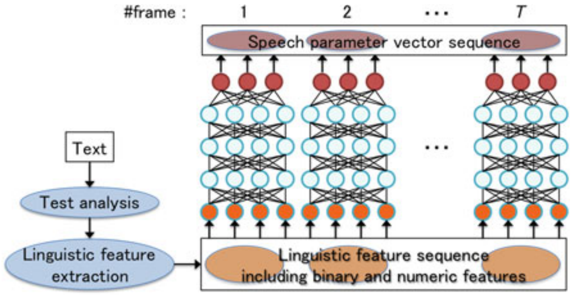
#frame :
1
2
…
T
Speech parameter vector sequence
Text
Test analysis
Linguistic feature extraction
Linguistic feature sequence including binary and numeric features

Fig. 1 A framework of DNN-based acoustic model

2 DNN-based Acoustic Model for Statistical Parametric Speech Synthesis

It is believed that the human speech production system has layered hierarchical structures to convert the linguistic information into speech. To approximate such a complicated process, DNN-based acoustic models that represent the relationship between linguistic and speech features have been proposed for statistical parametric speech synthesis [1–4] This section briefly reviews one of the state-of-the-art DNN-based acoustic models [1].

Figure 1 illustrates a framework of the DNN-based acoustic model. In this framework, linguistic features obtained from a given text are mapped to speech parameters by a DNN. The input linguistic features include binary answers to questions about linguistic contexts and numeric values, e.g. the number of words in the current phrase, the position of the current syllable in the word, and durations of the current phoneme. In [1], the output speech parameters include spectral and excitation parameters and their time derivatives (dynamic features). By using pairs of input and output features obtained from training data, the parameters of the DNN can be trained with a stochastic gradient descend (SGD) [10]. Speech parameters can be predicted for an arbitrary text by a trained DNN using forward propagation.

3 Deep Auto-encoder Based Acoustic Feature Extraction

An auto-encoder is an artificial neural network that is used generally for learning a compressed and distributed representation of a dataset. It consists of the encoder and the decoder. In the basic one-hidden-layer auto-encoder, the encoder maps an input vector $\mathbf{x}$ to a hidden representation $\mathbf{y}$ as follows:

$$\mathbf{y} = f_{\theta}(\mathbf{x}) = s(\mathbf{W}\mathbf{x} + \mathbf{b}), \tag{1}$$

where $\theta = \{\mathbf{W}, \mathbf{b}\}$. $\mathbf{W}$ and $\mathbf{b}$ represent an $m \times n$ weight matrix and a bias vector of dimensionality m, respectively, where n is the dimension of $\mathbf{x}$. The function s is a non-linear transformation on the linear mapping $\mathbf{Wx} + \mathbf{b}$. A sigmoid, a tanh, or a relu function is typically used for s. $\mathbf{y}$, the output of the encoder, is then mapped to $\mathbf{z}$, the output of the decoder. The mapping is performed by a linear mapping followed by an arbitrary function t that employs an $n \times m$ weight matrix $\mathbf{W}'$ and a bias vector of dimensionality n as follows:

$$\mathbf{z} = g_{\theta'}(\mathbf{y}) = t(\mathbf{W}'\mathbf{y} + \mathbf{b}'), \tag{2}$$

where $\theta' = \{\mathbf{W}', \mathbf{b}'\}$. An auto-encoder can be made deeper by stacking multiple layers of encoders and decoders to form a deep architecture.

Pre-training is widely used for constructing a deep auto-encoder. In pre-training, the number of layers in a deep auto-encoder increases twice as compare to a deep neural network (DNN) when stacking each pre-trained unit. It has been reported that fine-tuning with back-propagaqion through a deep auto-encoder is ineffective due to vanishing gradients at the lower layers [8]. To overcome this issue, we restrict the decoding weight as the transpose of the encoding weight following [10], that is, $\mathbf{W}' = \mathbf{W}^T$ where $\mathbf{W}^T$ denotes the transpose of $\mathbf{W}$. Each layer of a deep auto-encoder can be pre-trained greedily to minimize the reconstruction loss of the data locally. Figure 2 shows a procedure for constructing a deep auto-encoder using pre-training. In pre-training, a one-hidden-layer auto-encoder is trained and the encoding output of the locally trained layer is used as the input for the next layer. After all layers are pre-trained, they are stacked and are fine-tuned to minimize the reconstruction error over the entire dataset using error backpropagation [11]. We use the mean square error (MSE) for the loss function of a deep auto-encoder.

Figure 3 shows an example of original and reconstructed spectrograms using the standard mel-cepstral analysis and a deep auto-encoder. Both mel-cepstral analysis and the deep auto-encoder produced 120-dimensional acoustic features. We can clearly see that the deep auto-encoder reconstructs spectral fine structures more precisely than that of the mel-cepstral analysis. Log spectral distortions between natural spectrum and reconstructed spectrum calculated using 441 sentences were

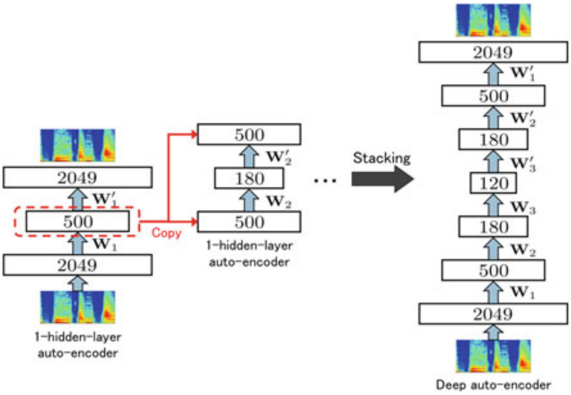

Fig. 2 Greedy layer-wise pre-training for constructing a deep auto-encoder

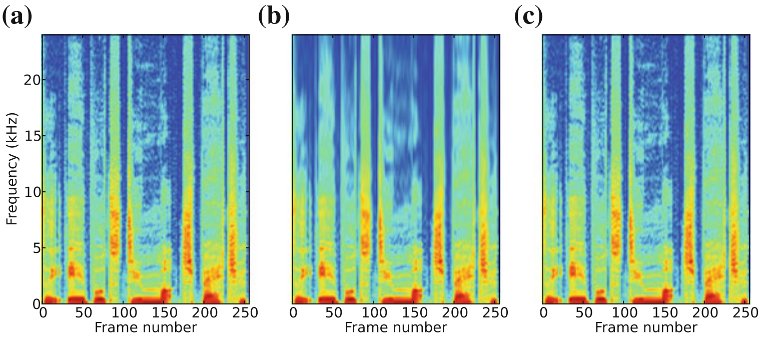

Fig. 3 Original and reconstructed spectra using mel-cepstral analysis and a deep auto-encoder. **a** Original, **b** mel-cepstrum, **c** deep auto-encoder

2.53 and 1.19 dB for the mel-cepstral analysis and deep auto-encoder, respectively. Similar auto-encoder based bottleneck features were tested for a ClusterGen speech synthesizer [12]. Our idea is different from [12] and we stack the decoder part of the deep auto-encoder onto another DNN for acoustic modeling.

4 Proposed DNN-based Spectral Modeling

A DNN-based acoustic model described in Sect. 2 may be used for the direct spectral modeling by substituting an output of the network from mel-cepstrum to the spectrum. However, the dimension of spectrum is much higher than that of mel-cepstrum. For a speech signal at 48 kHz, the mel-cepstral analysis order typically used is around 50-dim, whereas the dimension of spectrum corresponds to FFT points such as 2049. Because of this high dimensional data, a more efficient training technique is needed to construct a DNN that directly represents the relationship between linguistic features and spectra. In this paper, we hence propose a function-wise pre-training technique where we explicitly divide the general flow of the statistical parametric speech synthesis system into a few sub-processes, construct and optimize a DNN for each task individually, and stack the individual networks for the final optimization.

Figure 4 shows a procedure for constructing the proposed DNN-based spectral model. Details of each step of the proposed technique are as follows:

Step 1. Train a deep auto-encoder using spectra and extract bottleneck features for a DNN-based acoustic model used in Step 2. Layer-wise pre-training or other initialization may be used for the learning of the deep auto-encoder.

Step 2. Train a DNN-based acoustic model using the bottleneck features extracted in Step 1. Layer-wise pre-training or other initialization may be used for learning the DNN.

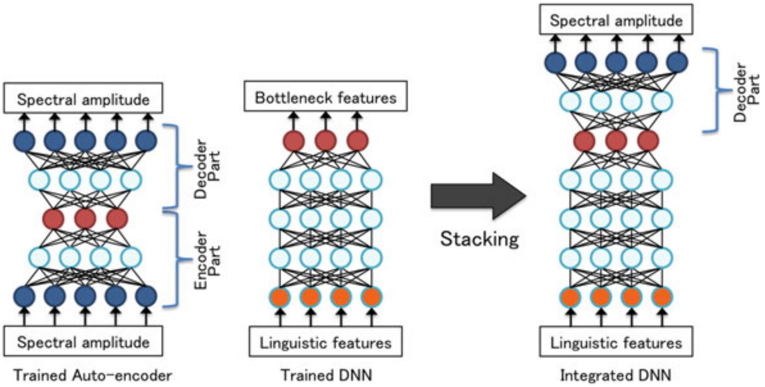

Fig. 4 Constructing a DNN-based spectral model based on a deep autoencoder and a DNN-based acoustic model

Step 3. Stack the trained DNN-based acoustic model for bottleneck features and the decoder part of the trained deep auto-encoder as shown in Fig. 4 and optimize the whole network.

A DNN that represents the relationship between linguistic features and spectra is constructed based on a DNN-based spectral generator and a DNN-based acoustic model using the bottleneck features. After this proposed pre-training, we fine-tune the DNN to minimize the error over the entire dataset using pairs of linguistic features and spectra in training data with SGD.

5 Experiments

We have evaluated the proposed technique in the subjective experiment. The dataset we use consists of 4546 short audio waveforms uttered by a professional female native speaker of English and each waveform is around 5 s long. All data was sampled at 48 kHz.

We compared three techniques; *CEPSTRUM* is the DNN that synthesizes cepstrum vectors, *SPECTRUM* has the same network structure as that of *CEPSTRUM*, but it outputs the spectral amplitudes directly, and *INTEG* is the proposed DNN that synthesizes spectrum amplitudes with the proposed pre-training framework. In these techniques, the dynamic and acceleration features were not used. Figure 5 shows structures of constructed DNNs for each technique. We trained five-hidden-layer DNN-based acoustic models for each technique. The number of units in each of the hidden layers was set to 1024. Random initialization was used in a way similar to [1]. In *INTEG*, we trained the symmetric five-hidden-layer auto-encoder. The numbers of units of the hidden layers were 2049, 500, 60, 500 and 2049 As a result, we

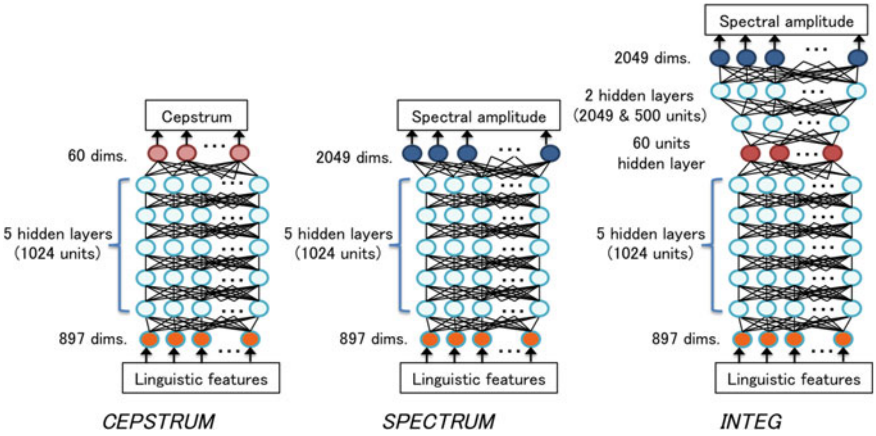

Fig. 5 Structures of constructed DNNs for each technique

constructed and fine-tuned the eight-hidden-layer (1024-1024-1024-1024-1024-60-500-2049) DNN for *INTEG*. We used a sigmoid function for all units of hidden and output layers of all DNNs.

For each waveform, we first extract its frequency spectra using the STRAIGHT vocoder with 2049 FFT points. For constructing the conventional system, 59 dimensional cepstrum coefficients were used. Spectrum and cepstrum were both frequency-warped using the Bark scale. Note that all the techniques synthesize only spectrum features and other requisite acoustic features; that is, F0 and aperiodicity measures were synthesized from the same HMM-based synthesis system [13]. Feature vectors for HMMs were comprised of 258 dimensions: 59 dimensional bark-cepstral coefficients (plus the 0th coefficient), log f0, 25 dimensional band aperiodicity measures, and their dynamic and acceleration coefficients. Phoneme durations were also estimated by HMM-based speech synthesis. The context-dependent labels were built using the pronunciation lexicon Combilex [14]. The linguistic features for DNN acoustic models were comprised of 897 dimensions: 858 dimensional binary features for categorical linguistic contexts, 36 numerical features for numerical linguistic contexts, and three numerical features for the position of the current frame and duration of the current segment. The linguistic features and spectral amplitudes in the training data were normalized for training DNNs. In the proposed technique, however, the bottleneck features are not normalized, and the normalization process is not used for hidden units in the integrated DNN. The input linguistic features were normalized to have zero-mean unit-variance, whereas the output spectral amplitudes were normalized to be within 0.0–1.0.

We synthesized speech samples from spectrum amplitudes, F0 features and aperiodicity measures using the STRAIGHT vocoder in all techniques. In *CEPSTRUM*, synthesized cepstral vectors were converted into spectrum amplitudes for using the STRAIGHT vocoder.

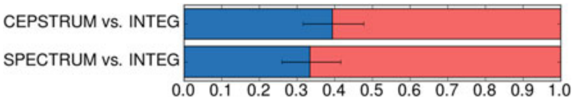

Fig. 6 Results of preference test

In subjective experiments, two preference tests were conducted. Seven subjects participated in both listening tests. Thirty sentences were randomly selected from the 180 sentences for each subject. The experiment was carried out using headphones in a quiet room.

5.1 Experimental Result

Figure 6 shows the results of the preference tests with 95 % confidence intervals. In the first preference test, they were asked to compare the DNN that synthesizes cepstrum vectors (*CEPSTRUM*) with the proposed DNN (*INTEG*). In the second preference test, they were asked to compare the DNN without the proposed pre-training technique that synthesizes spectrum amplitudes (*SPECTRUM*) with the proposed DNN (*INTEG*). The figure shows that the proposed technique produces more natural-sounding speech than other techniques. This indicates that the DNN that directly synthesizes spectra was efficiently trained using the proposed technique.

6 Conclusion

In this paper, we have proposed a technique for constructing a DNN that directly synthesizes spectral amplitudes. On the basis of the general flow for constructing the statistical parametric speech synthesis systems, a part of layers of a DNN could be efficiently pre-trained. Experimental results showed that the proposed technique increased the quality of synthetic speech.

In future work, we will investigate the effect of structures of a DNN-based acoustic model and a DNN-based spectrum auto-encoder more thoroughly. Time derivative features will also be interesting to investigate.

References

1. Zen, H., Senior, A., Schuster, M.: Statistical parametric speech synthesis using deep neural networks. In: Proceedings of ICASSP, pp. 7962–7966 (2013)
2. Ling, Z.-H., Deng, L., Yu, D.: Modeling spectral envelopes using restricted Boltzmann machines and deep belief networks for statistical parametric speech synthesis. IEEE Trans. Audio Speech Lang. Process. **21**, 2129–2139 (2013)

3. Fan, Y., Qian, Y., Xie, F., Soong, F.K.: TTS synthesis with bidirectional LSTM based recurrent neural networks. In: Proceedings of Interspeech, pp. 1964–1968 (2014)
4. Fernandez, R., Rendel, A., Ramabhadran, B., Hoory, R.: Prosody contour prediction with long short-term memory, bi-directional, deep recurrent neural networks. In: Proceedings of Interspeech, pp. 2268–2272 (2014)
5. Vishnubhotla, R., Fernandez, S., Ramabhadran, B.: An autoencoder neural-network based low-dimensionality approach to excitation modeling for hmm-based text-to-speech. In: Proceedings of ICASSP, pp. 4614–4617 (2010)
6. Chen, L.-H., Raitio, T., Valentini-Botinhao, C., Yamagishi, J., Ling, Z.-H.: DNN-based stochastic postfilter for HMM-based speech synthesis. In: Proceedings of Interspeech, pp. 1954–1958 (2014)
7. Kawahara, H., Masuda-Katsuse, I., Cheveigne, A.: Restructuring speech representations using a pitch-adaptive time-frequency smoothing and an instantaneous-frequency-based F0 extraction: Possible role of a repetitive structure in sounds. Speech Commun. **27**, 187–207 (1999)
8. Hochreiter, S., Bengio, Y., Frasconi, P., Schmidhuber, J.: Gradient flow in recurrent nets: the difficulty of learning long-term dependencies. Citeseer (2001)
9. Hinton, G.E.: Learning multiple layers of representation. Trends Cogn. Sci. **11**, 428–434 (2007)
10. Hinton, G.E., Salakhutdinov, R.: Reducing the dimensionality of data with neural networks. Science **313**(5786), 504–507 (2006)
11. Rumelhart, D.E., Hinton, G.E., Williams, R.J.: Parallel Distributed Processing: Explorations in the Microstructure of Cognition, vol. 1, pp. 318–362 (1986)
12. Muthukumar, P.K., Black, A.: A deep learning approach to data-driven parameterizations for statistical parametric speech synthesis (2014). arXiv:1409.8558
13. Zen, H., Tokuda, K., Black, A.W.: Statistical parametric speech synthesis. Speech Commun. **51**, 1039–1064 (2009)
14. Richmond, K., Clark, R., Fitt, S.: On generating combilex pronunciations via morphological analysis. In: Proceedings of Interspeech, pp. 1974–1977 (2010)