MULTI-SPEAKER MODELING AND SPEAKER ADAPTATION FOR DNN-BASED TTS SYNTHESIS

Yuchen Fan, Yao Qian, Frank K. Soong, and Lei He

Microsoft Research Asia, Beijing, China
{v-yufan, yaoqian, frankkps, helei}@microsoft.com

ABSTRACT

In DNN-based TTS synthesis, DNNs hidden layers can be viewed as deep transformation for linguistic features and the output layers as representation of acoustic space to regress the transformed linguistic features to acoustic parameters. The deep-layered architectures of DNN can not only represent highly-complex transformation compactly, but also take advantage of huge amount of training data. In this paper, we propose an approach to model multiple speakers TTS with a general DNN, where the same hidden layers are shared among different speakers while the output layers are composed of speaker-dependent nodes explaining the target of each speaker. The experimental results show that our approach can significantly improve the quality of synthesized speech objectively and subjectively, comparing with speech synthesized from the individual, speaker-dependent DNN-based TTS. We further transfer the hidden layers for a new speaker with limited training data and the resultant synthesized speech of the new speaker can also achieve a good quality in term of naturalness and speaker similarity.

Index Terms— statistical parametric speech synthesis, deep neural networks, multi-task learning, transfer learning

1. INTRODUCTION

Deep neural networks (DNNs) has shown its great power for acoustic modeling in TTS synthesis. Zen, et al. [1] investigated DNN-based TTS and pointed out comprehensively some intrinsic limitations of the conventional HMM-based speech synthesis, e.g. decision-tree based contextual state clustering. They showed that, on a rather large training corpus (˜35,000 sentences), DNN can yield better TTS performance than its GMM-HMM counterpart with a similar number of parameters. Qian, et al. [2] examined various aspects of DNN-based TTS training with a moderate size corpus (5,000 sentences), which is more commonly used for parametric TTS training. Fan, et al. [3] introduced LSTM-based RNN into parametric TTS synthesis, which uses deep structure for state transition modeling and upgrade the acoustic model from frame-level to sequence-level. However, compared with hidden Markov models (HMMs) in conventional parametric TTS synthesis [4], DNN is so complicate that it needs large amounts of phonetically and prosodically rich speech data to train a high-quality model. Due to the huge cost of recording, the available training data is always very limited, especially for one specific speaker.

In conventional HMM-based TTS synthesis, speaker adaptive training [5] uses multiple speakers' voice to train an average voice model and then adapt the average model to the speakers. This approach addresses the limitation of small-size corpus by joint training with various speakers' voice under various conditions. Similar technique is expected to improve DNN-based TTS synthesis.

Multi-task learning [6] and transfer learning [7] are both hot topics in machine learning, and they can also be applied into deep learning [8], such as in automatic speech recognition (ASR), DNN can learn knowledge across multiple languages and transfer the knowledge to another language [9]. Although the role of DNN in TTS is different from ASR, the ideas can still be used as reference.

In DNN-based TTS synthesis, DNN is used as regression model for linguistic and acoustic feature mapping. DNN can be viewed as a layer-structured model, that jointly learns a complicated linguistic feature transformation in hidden layers and a speaker-specific acoustic space in regression layer. With such structure understanding of DNN, we can decompose DNN into two parts (linguistic transformation and acoustic regression) to benefit DNN-based TTS synthesis by multi-speakers' data and solve the adaptation problem by shared hidden representation.

In this paper, we proposed a multi-speaker DNN, in which the hidden layers are shared across different speakers while the regression layers are speaker dependent. The shared hidden layers and the separate regression layer of each speaker are jointly trained with multiple speaker-dependent TTS corpuses. The shared hidden layers can be viewed as the global linguistic feature transformation that can be used for any speaker. Actually, the architecture and training procedure of multi-speaker DNN are instances of the multi-task learning, which combines the models with multiple related tasks and strengths them with shared knowledge.

Moreover, the shared linguistic feature transformation can be even transferred to a new speaker, which is a derivative of

transfer learning. For the new speaker with very limited training data, speaker adaptation can also be achieved by fixing the shared hidden layers and only updating the regression layer.

2. MULTI-SPEAKER MODELING

In the DNN-based TTS synthesis [2], as shown in Figure 1, DNN takes the converted linguistic features (binary & numeric) as input and acoustic features (LSP, F0 and U/V flag) as output, and learns the mapping between the linguistic and acoustic space. The model for each speaker was trained individually with their own voice.

Figure 2 shows the architecture of proposed multi-speaker DNN. In multi-speaker DNN, hidden layers are shared across all the speakers in training corpus, and can be considered as the global linguistic feature transformation shared by all the speakers. Conversely, each speaker has his own output layer, so-called regression layer, to modeling the specific acoustic space of himself. Compared with the conventional DNN, multi-speaker DNN takes the same input linguistic feature, which converted from text in the same manner, and the same output acoustic feature for each speaker.

Due to the changes in architecture, training algorithm also has some differences, but is still based on conventional back-propagation (BP) algorithm. For multi-speaker DNN, it's very crucial to train the network for all the speakers simultaneously, which means that each mini-batch should consider the data from all the speakers during the stochastic gradient decent (SGD) procedure, also training data also needs to be shuffled across all the speakers. Since each regression layer can only be used for its corresponding speaker, the error signal of one training sample can only be back-propagated to the specific regression layer and shared hidden layers. Fortunately, multi-speaker DNN can still be pre-trained by discriminative layer-wise pre-training [10] with multi-speaker corpus.

In synthesis, multi-speaker DNN can be decomposed. By only taking the specific regression layer and shared hidden layer, the sub-model can be used to synthesize speech of any speaker already trained in the multi-speaker DNN.

Multi-speaker DNN shares the hidden layers between different speakers, so that introduces a structural regularization to DNN model, which can be considered as an instance of multi-task learning. Multi-speaker DNN is joint optimized with multiple speakers' data, and supposed to benefit each speaker's synthesized speech from the knowledge of other speakers.

3. SPEAKER ADAPTATION

The shared hidden layers, lying in the multi-speaker DNN as Figure 2, can be treated as a global linguistic feature transformation applicable to multiple speakers. So the shared hidden layers can also be borrowed to transform linguistic feature for

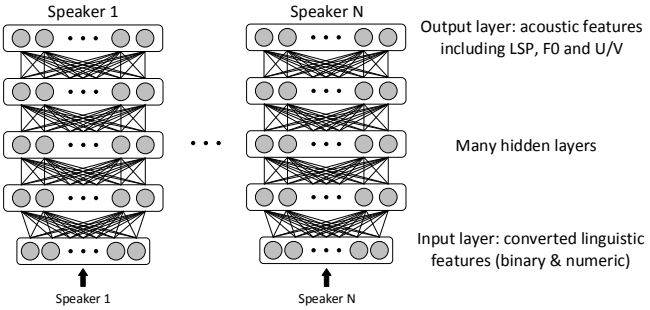

Fig. 1. DNN Architecture in DNN-based TTS Synthesis.

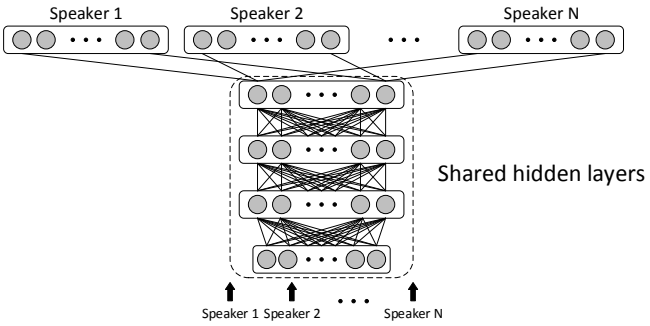

Fig. 2. Multi-speaker DNN Architecture in DNN-based TTS Synthesis.

new speakers. This procedure can be considered as a special case of transfer learning, also called speaker adaptation in TTS synthesis.

The training procedure for adaptation is quite straightforward. Due to training data for adaptation is very limited, the hidden layers transferred from multiple speakers' data should be fixed and only the regression layer will be updated. Considering there is only a linear regression between the shared hidden layers' output and target, parameter estimation is much simpler than the non-linear problem, such as DNN, and usually has closed-form solution. So the least squares method, instead of BP algorithm, can effectively and efficiently minimize the squared residuals between prediction and ground-truth.

With proposed speaker adaptation method, DNN-based TTS synthesis will be able to apply to the speakers, who have very limited training data. We also conjecture that, by borrowing knowledge from other speakers and only re-estimating a small portion of model parameters, the model for the speaker to adapt will be more robust.

4. EXPERIMENTS

4.1. Experimental Setup

A corpus of multiple native Mandarin speakers, both phonetically and prosodically rich, is used in our experiments. Speech signals are sampled at 16 kHz, windowed by a 25-ms window, and shifted every 5-ms. An LPC of 24th order is transformed into static LSPs and their dynamic counterparts.

The phonetic and prosodic contexts include quin-phone, the position of a phone, syllable and word in phrase and sentence, the length of word and phrase, stress of syllable, POS of word.

In the corpus, there are two male and two female standard Mandarin speakers for multi-speaker training. To evaluate our proposed multi-speaker DNN, We design three groups of training set:

- Set A: two male speakers; one hour speech for each; with the same transcriptions
- Set B: two male and two female speakers; one hour speech for each; with the same transcriptions
- Set C: two male and two female speakers; one hour speech for each; with different transcriptions among different speakers

We also choose 100 utterances with the same transcriptions for each speakers, and the transcriptions of these utterances are never covered in the training set. And there is another male accented Mandarin speakers for adaptation.

In the baseline DNN-based TTS, the input feature vector contains 585 dimensions, where 549 are binary features for categorical linguistic contexts and the rest are numerical linguistic contexts. The output feature vector contains a voiced/unvoiced flag, log F0, LSP, gain, their dynamic counterparts, totally 79 dimensions. Voiced/unvoiced flag is a binary feature that indicates the voicing of the current frame. DNN is set with 3 hidden layers and 512 nodes for each layer. An exponential decay function is used to interpolate F0 in unvoiced speech regions. 80% of silence frames are removed from the training data to balance the training data and to reduce the computational cost. Removing silence frames in DNN training was found useful for avoiding DNN overlearning silence label in speech recognition task. Both input and output features of training data are normalized to zero mean and unity variance. The weights are trained by back-propagation procedure with a mini-batch based stochastic gradient descent algorithm.

For the testing, DNN outputs are firstly fed into a parameter generation module to generate smooth feature parameters with dynamic feature constraints. Then formant sharpening based on LSP frequencies is used to reduce the over-smoothing problem of statistic parametric modeling and the resultant "muffled" speech. Finally speech waveforms are synthesized by an LPC synthesizer by using generated speech parameters.

Objective and subjective measures are used to evaluate the performance of TTS systems on testing data. Synthesis quality is measured objectively in terms of distortions between natural test utterances of the original speaker and the synthesized speech frame-synchronously where oracle state durations (obtained by forced alignment) of natural speech are used. The objective measures are F0 distortion in the root mean squared error (RMSE), voiced/unvoiced (V/U) swapping errors and normalized spectrum distance in log spectral distance (LSD). The subjective measure is an AB preference test between speech sentence pairs synthesized by different systems. In each preference test, we invite 10 native Mandarin subjects and each subject evaluates 50 pairs by using headsets. There are three preference choices: 1) the former is better; 2) the latter is better; 3) no preference or neutral (The difference between the paired sentences can not be perceived or can be perceived but difficult to choose which one is better).

Table 1. Objective Measures on Set A with and without Multi-speaker Modelling

Speaker	Measures	Baseline	Multi-speaker
Male #1	LSD (dB)	4.02	3.92 (-2.5%)
	V/U Err rate (%)	4.26	4.19 (-1.6%)
	F0 RMSE (Hz)	22.6	21.7 (-4.0%)
Male #2	LSD (dB)	3.96	3.85 (-2.8%)
	V/U Err rate (%)	7.36	6.78 (-7.8%)
	F0 RMSE (Hz)	16.6	15.5 (-6.6%)

Table 2. Objective Measures on Set B with and without Multi-speaker Modelling

Speaker	Measures	Baseline	Multi-speaker
Male #1	LSD (dB)	4.02	3.84 (-4.5%)
	V/U Err rate (%)	4.26	4.19 (-1.6%)
	F0 RMSE (Hz)	22.6	21.2 (-6.2%)
Male #2	LSD (dB)	3.96	3.77 (-4.8%)
	V/U Err rate (%)	7.36	6.69 (-9.1%)
	F0 RMSE (Hz)	16.6	15.3 (-7.8%)
Female #1	LSD (dB)	3.77	3.60 (-4.5%)
	V/U Err rate (%)	5.96	5.89 (-1.2%)
	F0 RMSE (Hz)	29.5	26.8 (-9.2%)
Female #2	LSD (dB)	4.02	3.83 (-4.7%)
	V/U Err rate (%)	7.54	7.32 (-2.9%)
	F0 RMSE (Hz)	25.9	23.8 (-8.1%)

4.2. Evaluation Results and Analysis

To evaluate whether multi-speaker modeling can benefit DNN-base TTS synthesis, we firstly try to apply the proposed method on Set A. As shown in Table 1, multi-speaker modelling outperforms baseline system in all kinds of objective measures. The results indicate that shared hidden layers for linguistic feature transformation can effectively exploit many commonalities between speakers.

As we known that there are a lot of acoustic differences between male and female, multi-speaker modeling is expected to be evaluated with Set B which is a cross-gender corpus. The results shown in Table 2 indicate that shared hidden layers can transform the linguistic information into an universal space for both male and female, and make the synthesized waveform better.

For better linguistic modelling, multi-speaker system should be further improved by more widely covered linguis-

Table 3. Objective Measures on Set C with and without Multi-speaker Modelling

Speaker	Measures	Baseline	Multi-speaker
Male #1	LSD (dB)	4.34	4.13 (-4.8%)
	V/U Err rate (%)	4.26	4.09 (-4.0%)
	F0 RMSE (Hz)	23.4	21.4 (-8.5%)
Male #2	LSD (dB)	3.96	3.43 (-13.4%)
	V/U Err rate (%)	7.36	5.84 (-20.7%)
	F0 RMSE (Hz)	16.6	13.9 (-16.3%)
Female #1	LSD (dB)	3.91	3.74 (-4.3%)
	V/U Err rate (%)	5.88	5.82 (-1.0%)
	F0 RMSE (Hz)	28.9	27.7 (-4.2%)
Female #2	LSD (dB)	4.00	3.81 (-4.8%)
	V/U Err rate (%)	7.64	7.05 (-7.7%)
	F0 RMSE (Hz)	25.2	22.5 (-10.7%)

tic data. Therefore, we train the model on Set C, which contains speech with the totally different transcripts among different speakers. As shown in Table 3, comparing with Table 2, the distance between synthesis and natural speech can be further reduced with the same size of training corpus. We also evaluate the pair of system by perceptual test. The preference score, shown in Figure 3, indicates multi-speaker modelling can significantly (at $p < 0.01$ level) improve the DNN-based TTS synthesis with multiple speakers' corpus.

Fig. 3. Preference score on Set C with and without multi-speaker modelling.

Speaker adaptation technique is usually used for the speaker with very limited data or whose speech is not very friendly to the synthesis system training. So in this experiments, we use the corpus from a male Chinese speaker, whose Mandarin is not very standard and makes it very hard to train a good synthesis model.

Speaker adaptation is build on the top of well-trained multi-speaker system. Shared hidden layers are borrowed from the multi-speaker system and keep their parameters fixed. The new speaker's data is only used to train a new regression layer.

As adaptation task is very sensitive to the size of training data, we firstly try to use different size of training data to investigate the relationship between data size and performance. As shown in Figure 4, it's not a surprise that objective measures will be reduced with more training data, but we can find that the decrease becomes much slower when using more than 100 training utterances, which size is commonly used for speaker adaptation.

To examine the performance of our proposed adaptation method for DNN-based TTS, we add other two systems as references, except using 100 utterances training data for adaptation. One system (multi-speaker) directly puts the 100 utterances training data into the multi-speaker training as a new speaker. The other (baseline) uses 1000 utterances training data (10 times to adaptation) to train a mono-speaker DNN-based TTS synthesis. Table 4 shows that the performance of proposed adaptation method is very similar to multi-speaker training, whereas proposed adaptation method can be trained in only few minutes. However, due to the limitation of training data size, the baseline system training with 1000 utterances achieves the best objective performance.

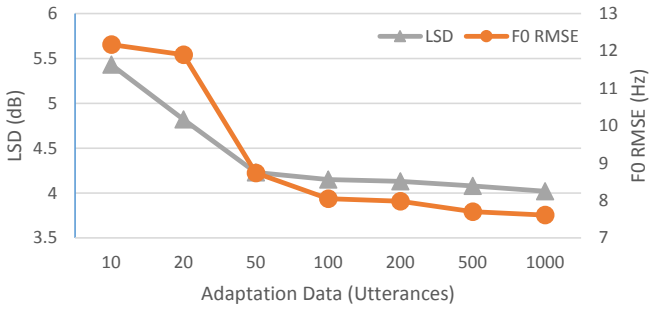

Fig. 4. Objective measures of adaptation with different training data size.

Table 4. Objective measures for speaker adaptation and its control group

Measures	Adaptation	Multi-speaker	Baseline
LSD (dB)	4.15	4.13	3.79
V/U Err rate (%)	5.25	5.30	4.78
F0 RMSE (Hz)	8.05	8.46	7.81

Subjective test are performed between the adaptation and baseline system, as shown in Figure 5. It's interesting that, opposite to the subjective measure, the adaptation approach can significantly (at $p < 0.01$ level) outperform the baseline system. One possible reason is that, although the speech from the speaker to adapt is not standard Mandarin, the knowledge of other speakers transferred by the shared hidden layers can correct some mispronunciations but not influence the similarity.

Fig. 5. Preference score of adaptation and baseline of DNN-based TTS.

5. CONCLUSIONS

In this paper, we investigate the multi-task learning into DNN-based TTS synthesis for multi-speaker modelling and achieve improvements on both objective and subjective measurements, comparing with individually modelling baseline. Also, we employ transfer learning for speaker adaptation and accomplish good quality for both naturalness and speaker similarity. Further, we will try to apply this methods onto some bigger corpus with more speakers.

6. REFERENCES

[1] Heiga Zen, Andrew Senior, and Mike Schuster, "Statistical parametric speech synthesis using deep neural networks," in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*. IEEE, 2013, pp. 7962–7966.

[2] Yao Qian, Yuchen Fan, Wenping Hu, and Frank K Soong, "On the training aspects of deep neural network (DNN) for parametric tts synthesis," in *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*. IEEE, 2014, pp. 3829–3833.

[3] Yuchen Fan, Yao Qian, Feng-Long Xie, and Frank K Soong, "TTS synthesis with bidirectional LSTM based recurrent neural networks," in *INTERSPEECH*, 2014.

[4] Heiga Zen, Keiichi Tokuda, and Alan W Black, "Statistical parametric speech synthesis," *Speech Communication*, vol. 51, no. 11, pp. 1039–1064, 2009.

[5] Junichi Yamagishi, Takashi Nose, Heiga Zen, Zhen-Hua Ling, Tomoki Toda, Keiichi Tokuda, Simon King, and Steve Renals, "Robust speaker-adaptive HMM-based text-to-speech synthesis," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 17, no. 6, pp. 1208–1230, 2009.

[6] Rich Caruana, *Multitask learning*, Springer, 1998.

[7] Sinno Jialin Pan and Qiang Yang, "A survey on transfer learning," *Knowledge and Data Engineering, IEEE Transactions on*, vol. 22, no. 10, pp. 1345–1359, 2010.

[8] Yoshua Bengio, "Deep learning of representations for unsupervised and transfer learning.," in *ICML Unsupervised and Transfer Learning*, 2012, pp. 17–36.

[9] Jui-Ting Huang, Jinyu Li, Dong Yu, Li Deng, and Yifan Gong, "Cross-language knowledge transfer using multilingual deep neural network with shared hidden layers," in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*. IEEE, 2013, pp. 7304–7308.

[10] Frank Seide, Gang Li, Xie Chen, and Dong Yu, "Feature engineering in context-dependent deep neural networks for conversational speech transcription," in *Automatic Speech Recognition and Understanding (ASRU), 2011 IEEE Workshop on*. IEEE, 2011, pp. 24–29.