FEATURE BASED ADAPTATION FOR SPEAKING STYLE SYNTHESIS

Xixin Wu[1], *Lifa Sun*[1], *Shiyin Kang*[3], *Songxiang Liu*[1], *Zhiyong Wu*[*1,2], *Xunying Liu*[1], *Helen Meng*[1,2]

[1]Department of Systems Engineering and Engineering Management,
The Chinese University of Hong Kong, Hong Kong SAR, China
[2]Tsinghua-CUHK Joint Research Center for Media Sciences, Technologies and Systems,
Graduate School at Shenzhen, Tsinghua University, Shenzhen, China
[3]Tencent AI Lab, Tencent, Shenzhen, China
{wuxx, lfsun, sxliu, zywu, xyliu, hmmeng}@se.cuhk.edu.hk, shiyinkang@tencent.com

ABSTRACT

Speaking style plays an important role in the expressivity of speech for communication. Hence speaking style is very important for synthetic speech as well. Speaking style adaptation faces the difficulty that the data of specific styles may be limited and difficult to obtain in large amounts. A possible solution is to leverage data from speaking styles that are more available, to train the speech synthesizer and then adapt it to the target style for which the data is scarce. Conventional DNN adaptation approaches directly update the top layers of a well-trained, style-dependent model towards the target style. The detailed local context-level mismatch between the original and the target styles is not considered. In order to address this issue, two frame-level input feature-based style adaptation techniques are investigated in this paper. We will use style features extracted from (1) a target-style data trained bottleneck DNN, and (2) a novel cross-style residual feature regression DNN. These features are used for top-layer adaptation of a well-trained style-dependent synthesis network. Experimental results on adapting the declarative style to the interrogative style demonstrate the effectiveness of our proposed style features in improving the expressiveness of synthesizing speech for the interrogative style, while maintaining speech quality.

Index Terms— style adaptation, speaking style, style feature, speech synthesis, expressiveness

1. INTRODUCTION

Recent progress of deep neural network (DNN) techniques has shown great promise in generating natural and intelligible speech [1–3]. However, the expressiveness of text-to-speech (TTS) synthesis still needs to be improved. There are many factors affecting the expressiveness of generated speech. One of the most important factors is speaking style [4]. A suitable speaking style can vividly express the information encoded in the speech signal and enhance the interactions between human and machines. Speech with same content but with different speaking styles may convey different meanings, for example, represented by interrogative and declarative styles. To train from scratch a TTS model with a specific style requires a large amount of matched training data. However, data of some specific speaking styles, for example, the interrogative style, are difficult to obtain in large quantities. One solution to this problem is to use speaking style adaptation techniques leveraging large quantities of data with widely available style(s) for generating speech with special style. In our daily lives, the declarative style is most commonly used and hence the data is widely available. In contrast, although the interrogative style is essentially for conveying the semantics of a question, data of the interrogative style is often very limited. We aim to conduct research in speaking style adaption for any style(s). As an initial step, we select the interrogative style for which we do not have much data, and apply adaptation to the DNN-based speech synthesis system, which is trained on declarative style data, for generating interrogative style output.

The differences between styles manifest themselves at multiple levels, i.e., at the global, utterance level and also the local, segmental level. For example, if we consider the differences between the declarative and interrogative styles at the global level—the mean pitch value of the interrogative style is much higher than that of declarative style in an utterance [5]. On the other hand, at the local level, words carrying interrogative information have pitch value raised more intensively [6].

Speaker and style adaptation has been thoroughly investigated in hidden Markov model (HMM)-based synthesis approaches [7, 8]. Deep neural network (DNN)-based approaches have demonstrated effectiveness in speaker adaptation for automatic speech recognition (ASR) [9–13], which has also drawn increasing interests from TTS researchers [14–16]. Existing adaptation methods can be categorized into three main types: 1) Input feature-based adaptation and augmentation—utterance-level speaker features, for example, speaker codes [16], i-vector [15] and d-vector [17], are incorporated into input features to improve controllability over the target voice of the generated speech and have been shown effective. 2) Model-based adaptation—retraining the top regression layers of a well-trained model has been investigated in [14]. Learning hidden unit contribution (LHUC) [18] is applied to speaker adaptation in [15]. 3) Output feature transformation—a transformation function is built to transform the original predicted acoustic features to the target voice [15, 19].

A major issue with the existing adaptation approaches, as described above, is that the mismatch between source and target voices is only considered at a global level (e.g., speaker or utterance level), while the more detailed mismatch at the local context level are not explicitly modeled. Consequently, the methods do not perform well for style adaptation.

In order to address the above issue, frame level feature-based adaptation methods for speaking style are investigated in this paper. In contrast to the use of utterance or speaker level features, frame level features encoding the characteristics of interrogative style speech are fed into the top layers of a large, well-trained synthesis

* Corresponding author

DNN constructed with sufficient declarative style data. The frame level features are used as auxiliary input features to facilitate a more flexible local context level style adaptation. Two forms of style features are used: the first are bottleneck features (BNFs) extracted from a compact synthesis DNN trained using interrogative data only. The second explicitly encodes the difference between styles in the form of residual features (RFs) predicted by a special DNN, which is designed to learn such difference, using the offset between their acoustic features.

This paper is organised as follows: Section 2 reviews model-based style adaptation on retraining of DNN top layer parameters. Two forms of input feature-based style adaptation methods are proposed in Section 3. Section 4 presents the detail of experiments. The conclusion is drawn in Section 5.

2. MODEL BASED STYLE ADAPTATION

In DNN-based speech synthesis, the task is to train a network to map linguistic features to acoustic features. Large amounts of training data is required to robustly estimate model parameters. For the declarative style, the training data is often widely available and thus adequate for building a good TTS system. However, for the interrogative style, it is often more difficult to obtain training data in large amounts to develop a high-quality TTS system. One commonly used approach to handle this problem is to adapt the top layer parameters of a well trained and large sized network on sufficient quantities of declarative style data to the target interrogative style, while fixing the lower layer parameters [14].

Referencing previous work [14, 20, 21], the baseline TTS adaptation system in this paper is a bidirectional long short term-memory (BLSTM) recurrent neural network (RNN) model with its top layers adapted. The model architecture is illustrated on the left side of Fig. 1, consisting of lower-level fixed layers and top-level adapted layers. During interrogative style adaptation, the model is initially trained with large amounts of declarative data. In the subsequent adaptation stage, another set of interrogative top-level layers are stacked upon the lower-level fixed layers and retrained with interrogative style data. The lower-level fixed layers and the resulting top layers are then used to generate interrogative acoustic features. Using this technique, a good trade-off between modeling precision and robustness can be achieved. However, the detailed, frame level similarity and dissimilarity in style, and their variations over time, are not explicitly modeled in this framework, as discussed in Section 1.

3. FEATURE BASED STYLE ADAPTATION

To provide frame-various interrogative information for the adaptation, we propose to extract some compact style features with style information from the sparse and high-dimensional input linguistic features. The extracted style features are concatenated with the outputs of lower-level fixed hidden layers, and fed to the stacked top layers. The top layers are retrained as conventional approaches. The model architecture is shown in the right side of Fig. 1. The following subsections describe the extraction of two forms of proposed style features, BNFs and RFs.

3.1. Interrogative style bottleneck features

BNFs are extracted from the activation outputs of the hidden layer with a smaller number of hidden units, compared to the other hidden layers (e.g., 64 vs. 512 in this paper) [22]. The small layer size is designed to constrict the feature representation learned inside the DNN to a compact, lower dimensional space. Bottleneck features have been successfully applied in ASR [22–25]. Doddipatla et al. [24] proposes to extract speaker dependent BNFs for speaker adaptation in ASR. Wu et al. [26] also demonstrates the effectiveness of BNFs in TTS as input features. In this work, to extract the BNFs, we first train a naive bottleneck DNN (bDNN) model with a bottleneck hidden layer based on the interrogative training data, as shown in Fig. 2(a). The activation outputs of the bottleneck hidden layer are then used as the BNFs for adapting the top layers. Though bDNN is trained with limited data and has a reduced ability to generate interrogative acoustic features compared with larger sized networks trained with sufficient data, it is hoped the resulting BNFs carry the style dependent information learned from the interrogative data.

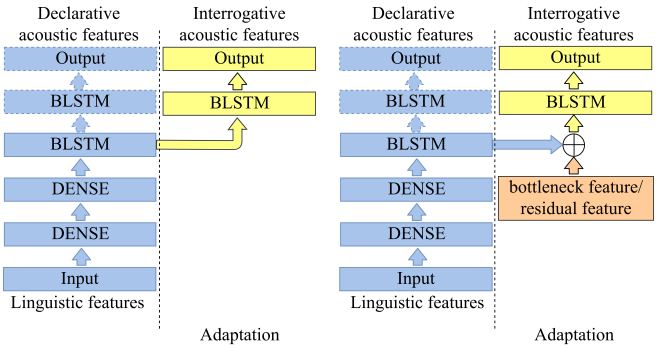

Fig. 1. The architecture of adaptation models. (Left: the conventional model-based style adaptation. Right: the proposed feature-based style adaptation)

3.2. Style difference residual features

To further improve the compactness of the style features, we investigate extracting features that can explicitly represent style differences. The frame level offsets between the desired interrogative style acoustic features and the acoustic features predicted using the unadapted declarative style DNN are used to encode such style difference in the RFs. These RFs are then used as style features to facilitate the top layer adaptation shown on the right side of Fig. 1. The extraction of RFs are conducted in the following stages, as shown in Fig. 2(b):

Stage 1: Train a DNN model, referred as decDNN, with large amounts of declarative data;

Stage 2: With the trained decDNN, generate the declarative acoustic features with the linguistic features extracted from interrogative data. Subtract the generated declarative features from the interrogative acoustic features to obtain the offset between the declarative and interrogative acoustic features. Train another DNN model, referred as rDNN, to map the extracted linguistic features to corresponding obtained acoustic feature offset.

Stage 3: The given linguistic features are fed to the rDNN to obtain the corresponding RFs.

4. EXPERIMENTS

4.1. Corpus

In our experiment, a corpus of one female Mandarin native speaker is used, consisting of 5,000 utterances (around 5 hours) with declarative style and 484 utterances (around 25 minutes) with interrogative

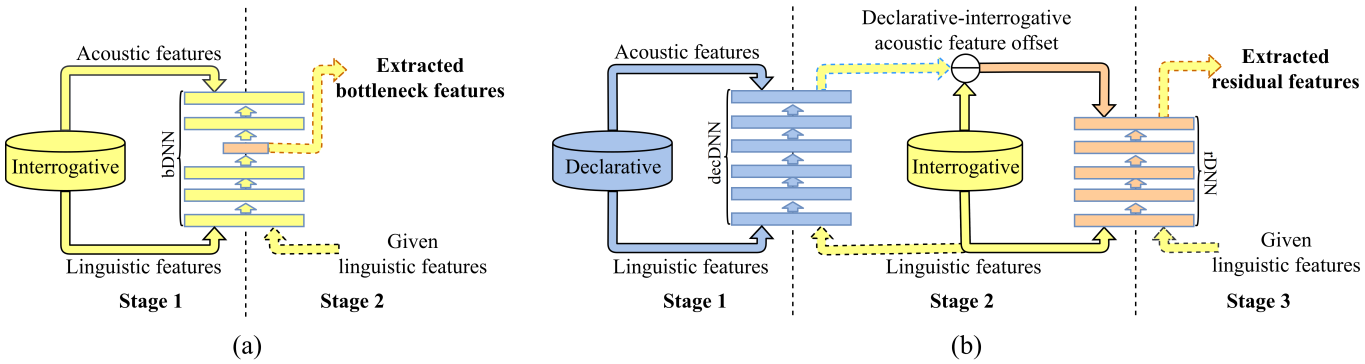

Fig. 2. The procedure of extraction of BNFs and RFs. (a) BNF extraction. Stage 1: Training of the interrogative bDNN; Stage 2: Extracting the bottleneck layer outputs as BNFs. (b) RF extraction. Stage 1: Training of the decDNN; Stage 2: Obtaining of declarative-interrogative acoustic feature differential and training of the rDNN; Stage 3: Prediction of RFs of the given linguistic features.

style. There are totally 98,084 syllable samples in total, covering 1,660 tonal syllable classes. The corpus is originally prepared for concatenative speech synthesis. The syllable boundary information is manually labeled by checking the forced alignment result with HMM. We use all the 5,000 utterances as training data to train the TTS system with declarative style. 400 utterances with interrogative style are used as adaptation data and the remaining 84 utterances as testing data. The speech signals are sampled at 16 kHz, windowed by a 25-ms window and shifted every 5 ms. 80% of the starting and ending silence frames are removed. We extract 39-dimensional mel-cepstral coefficients (MCEPs) plus log energy, 25-dimensional band-aperiodicity parameters (BAPs), logarithmic fundamental frequency (LF0), their delta and voice/unvoiced (V/UV) decision as frame-level acoustic features. LF0 is interpolated in unvoiced frames. For linguistic features, we use syllable features and prosodic features, e.g., prosodic word and prosodic phrase features. The input linguistic features for each syllable include 291 binary features for categorical linguistic contexts (e.g., initial or final of syllable), and 35 numerical features for numerical linguistic information (e.g., the position of current syllable in a prosody word). In this work, we directly use the manually labeled *syllable duration information* in the input linguistic features. All numerical linguistic features and acoustic features are normalised to have zero mean and unit variance. After the mapping from linguistic features to acoustic features is learned, the generated acoustic features are fed to STRAIGHT [27] to synthesize speech waveform.

4.2. Experimental setup

We train three systems as baseline systems to evaluate the naturalness and style expressiveness of our proposed systems. The first one is a TTS system with interrogative style, referred as INT. It is trained with only the adaptation data (i.e., 400 interrogative utterances). INT is composed of one feed-forward (FF) layer with 256 units with hyperbolic tangent activations (all FF layers in this work use hyperbolic tangent function unless specified), one BLSTM layer with 256 memory blocks per direction, and one output layer with linear activation. The second baseline system is a TTS system trained with the training data (i.e., 5,000 declarative utterances), denoted as DEC. DEC has five layers: two FF layers with 256 units per layer, two BLSTM layers with 256 memory blocks per direction, and one linear output layer. The third baseline system, DEC-INT, is a TTS adaptation system as described in Section 2, as shown in the left side of Fig. 1. DEC-INT directly borrows the lower three layers from the trained DEC, including two FF layers and one BLSTM layer. Two top layers are stacked upon the lower layers, including one BLSTM layers with 256 memory blocks per direction and one linear output layer. The top layers are then trained with the adaptation data with the lower layers fixed.

For style feature extraction, the bDNN in Fig. 2(a) consists of four hidden FF layers and one linear output layer. The last but one hidden layer is the bottleneck layer with 64 units, and the other hidden layers have 512 units. The bDNN is trained with the adaptation data. The decDNN has the same architecture as DEC and is trained with the declarative training data. The rDNN is composed of three FF hidden layers with 512 units and one linear output layer. The declarative-interrogative acoustic feature offset is calculated on the numerical acoustic features of the adaptation data and its dimension is 66. The feature offset data is then used to train the rDNN. With the trained bDNN and rDNN, we extract the BNFs and RFs for each utterance in the adaptation data and the testing data.

To evaluate the proposed framework with style features, we build three systems, i.e., DEC-b-INT, DEC-r-INT, DEC-br-INT, with different style feature configurations. 'b', 'r', 'br' stand for BNFs, RFs and the concatenation of BNFs and RFs. For these three systems, the configurations of lower-level fixed layers and top stacked adapted layers are the same as DEC-INT. The concatenation of lower fixed layer outputs and the style features is fed to the top adapted layers. The adaptation data is used to train the top adapted layers, with the lower layers fixed.

4.3. Objective evaluation

To objectively evaluate the performance of the above systems, we adopt four measures, mel-cepstral distortion (MCD), BAP distortion, F0 distortion in the root mean squared error (RMSE), and V/UV error rate for the four kinds of acoustic features we used. As shown in Table 1, our proposed systems outperform the three baseline systems, i.e., INT, DEC, and DEC-INT, on all the four metrics. We also investigate the effect of adaptation data size on the system performance. Experimental results show that the proposed systems achieve better performance on various data size, as shown in Fig. 3. To optimize the bottleneck layer size, we evaluate the results of different BNF sizes in DEC-b-INT. From the results in Fig. 3, we set the BNF size as 64.

Table 1. Objective evaluations for the six systems.

Systems	MCD (dB)	BAP (dB)	F0 RMSE (Hz)	V/UV Error Rate (%)
INT	6.02	4.81	30.01	12.11
DEC	5.63	4.49	30.21	7.74
DEC-INT	5.31	4.42	28.19	7.15
DEC-b-INT	4.98	**4.29**	27.41	**6.83**
DEC-r-INT	5.07	4.32	27.56	7.07
DEC-br-INT	**4.97**	**4.29**	**27.08**	6.84

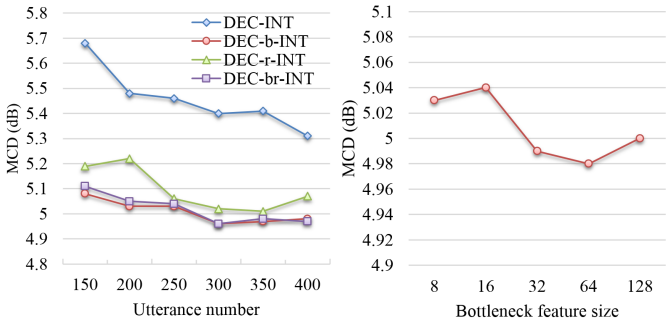

Fig. 3. Objective evaluation of various adaptation data sizes and bottleneck feature sizes.

4.4. Subjective evaluation

We conduct mean opinion score (MOS) test and AB preference tests to subjectively evaluate the above systems. 17 utterances are randomly selected from the testing data and synthesized by the six systems respectively, thus we have 102 utterances to be evaluated. We invite 20 subjects without listening impairment to participate in the tests. In the MOS test, each subject listens to each utterance and give a 5-point scale score of naturalness (5: excellent, 4: good, 3: fair, 2: poor, 1: bad). Fig. 4 presents the MOS results. The proposed systems, DEC-r-INT, DEC-b-INT, and DEC-br-INT, achieve significantly ($p < 0.001$) better performance than the baseline systems DEC-INT and INT, and obtain comparable naturalness with DEC. This demonstrates that the style features can help to maintain the naturalness.

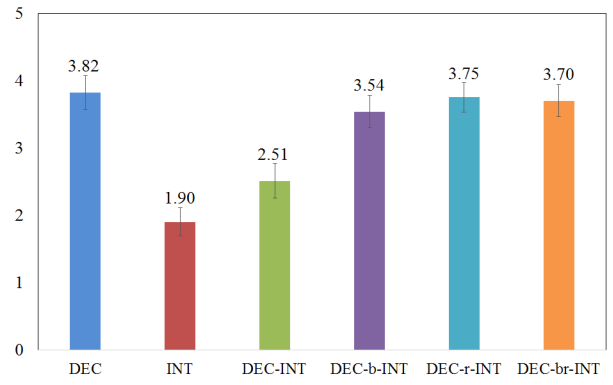

Fig. 4. MOS test results of various systems.

In each preference test, each subject listens to 17 pairs of utterances generated by two different systems, and provides a interrogative style preference choice: 1) the former is better; 2) the latter is better; 3) no preference or neutral (The difference between the paired utterances is difficult to be perceived). The results of preference tests are given in Fig. 5. DEC-b-INT can significantly ($p < 0.001$) outperform the baselines DEC and DEC-INT. The DEC-r-INT also outperforms the two baselines but is inferior to DEC-b-INT. Interestingly, DEC-br-INT achieves the best performance, which indicates combining BNFs and RFs gives better performance than individual ones.

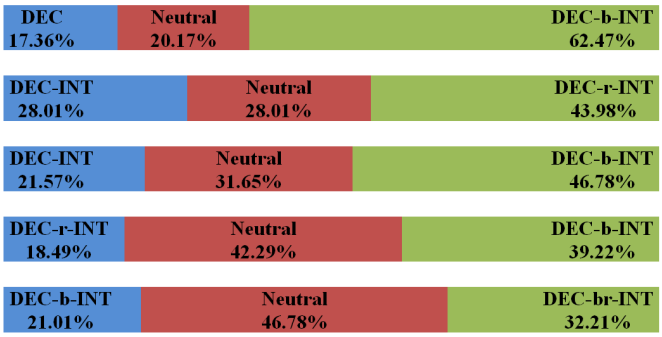

Fig. 5. Preference test results of various systems.

5. CONCLUSIONS

In this paper, we propose an input feature-based adaptation methods for DNN-based speech synthesis systems. Bottleneck features and residual features, which encode the finer, local context characteristics of the target style, are fed as auxiliary input features into the newly stacked top adapted layers of the adaptation model, where the lower layers are borrowed from a well trained DNN constructed using data with widely available style. As demonstrated by the experiments conducted on the data of declarative and interrogative styles, our methods can effectively improve the interrogative style in generated speech while maintaining a high quality in both objective and subjective evaluation tests. Since our method has no style-specific constraint, it can be flexibly applied to adaptation of other speaking styles. In future work, we will investigate duration adaptation techniques for speaking styles.

6. ACKNOWLEDGEMENT

This work is partially supported by National Natural Science Foundation of China-Research Grants Council of Hong Kong (NSFC-RGC) joint fund (61531166002, N_CUHK404/15).

7. REFERENCES

[1] Aaron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves, Nal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu, "Wavenet: A generative model for raw audio," *arXiv preprint arXiv:1609.03499*, 2016.

[2] Shiyin Kang, Xiaojun Qian, and Helen Meng, "Multi-distribution deep belief network for speech synthesis," in *Proc. ICASSP*. IEEE, 2013, pp. 8012–8016.

[3] Zhen-Hua Ling, Shi-Yin Kang, Heiga Zen, Andrew Senior, Mike Schuster, Xiao-Jun Qian, Helen M Meng, and Li Deng, "Deep learning for acoustic modeling in parametric speech generation: A systematic review of existing techniques and future trends," *IEEE Signal Processing Magazine*, vol. 32, no. 3, pp. 35–52, 2015.

[4] Wael Hamza, Ellen Eide, Raimo Bakis, Michael Picheny, and John Pitrelli, "The ibm expressive speech synthesis system," in *Eighth International Conference on Spoken Language Processing*, 2004.

[5] Jiahong Yuan, Chilin Shih, and Greg P Kochanski, "Comparison of declarative and interrogative intonation in chinese," in *Speech Prosody 2002, International Conference*, 2002.

[6] Wu Yanhong, Tao Jianhua, and Lu Jilun, "The design of corpus for interrogative speech synthesis," 2006.

[7] Masatsune Tamura, Takashi Masuko, Keiichi Tokuda, and Takao Kobayashi, "Adaptation of pitch and spectrum for hmm-based speech synthesis using mllr," in *Proc. ICASSP*. IEEE, 2001, vol. 2, pp. 805–808.

[8] Junichi Yamagishi, Makoto Tachibana, Takashi Masuko, and Takao Kobayashi, "Speaking style adaptation using context clustering decision tree for hmm-based speech synthesis," in *Proc. ICASSP*. IEEE, 2004, vol. 1, pp. I–5.

[9] George Saon, Hagen Soltau, David Nahamoo, and Michael Picheny, "Speaker adaptation of neural network acoustic models using i-vectors.," in *ASRU*, 2013, pp. 55–59.

[10] Chunyang Wu and Mark JF Gales, "Multi-basis adaptive neural network for rapid adaptation in speech recognition," in *Proc. ICASSP*. IEEE, 2015, pp. 4315–4319.

[11] Tian Tan, Yanmin Qian, Maofan Yin, Yimeng Zhuang, and Kai Yu, "Cluster adaptive training for deep neural network," in *Proc. ICASSP*. IEEE, 2015, pp. 4325–4329.

[12] C Zhang and Philip C Woodland, "Dnn speaker adaptation using parameterised sigmoid and relu hidden activation functions," in *Proc. ICASSP*. IEEE, 2016, pp. 5300–5304.

[13] Xurong Xie, Xunying Liu, Tan Lee, and Lan Wang, "Rnn-lda clustering for feature based dnn adaptation," *Proc. Interspeech*, pp. 2396–2400, 2017.

[14] Yuchen Fan, Yao Qian, Frank K Soong, and Lei He, "Multi-speaker modeling and speaker adaptation for dnn-based tts synthesis," in *Proc. ICASSP*. IEEE, 2015, pp. 4475–4479.

[15] Zhizheng Wu, Pawel Swietojanski, Christophe Veaux, Steve Renals, and Simon King, "A study of speaker adaptation for dnn-based speech synthesis.," in *Prof. Interspeech*, 2015, pp. 879–883.

[16] Hieu-Thi Luong, Shinji Takaki, Gustav Eje Henter, and Junichi Yamagishi, "Adapting and controlling dnn-based speech synthesis using input codes," in *Proc. ICASSP*. IEEE, 2017, pp. 4905–4909.

[17] Rama Doddipatla, Norbert Braunschweiler, and Ranniery Maia, "Speaker adaptation in dnn-based speech synthesis using d-vectors," *Proc. Interspeech*, pp. 3404–3408, 2017.

[18] Pawel Swietojanski and Steve Renals, "Learning hidden unit contributions for unsupervised speaker adaptation of neural network acoustic models," in *Spoken Language Technology Workshop (SLT), 2014 IEEE*. IEEE, 2014, pp. 171–176.

[19] Tomoki Toda, Alan W Black, and Keiichi Tokuda, "Voice conversion based on maximum-likelihood estimation of spectral parameter trajectory," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 8, pp. 2222–2235, 2007.

[20] Yibin Zheng, Ya Li, Zhengqi Wen, Bin Liu, and Jianhua Tao, "Investigating deep neural network adaptation for generating exclamatory and interrogative speech in mandarin," *Journal of Signal Processing Systems*, 2017.

[21] Yuchen Fan, Yao Qian, Frank K Soong, and Lei He, "Unsupervised speaker adaptation for dnn-based tts synthesis," in *Proc. ICASSP*. IEEE, 2016, pp. 5135–5139.

[22] Dong Yu and Michael L Seltzer, "Improved bottleneck features using pretrained deep neural networks," in *Twelfth Annual Conference of the International Speech Communication Association*, 2011.

[23] Yulan Liu, Pengyuan Zhang, and Thomas Hain, "Using neural network front-ends on far field multiple microphones based speech recognition," in *Proc. ICASSP*. IEEE, 2014, pp. 5542–5546.

[24] Rama Doddipatla, Madina Hasan, and Thomas Hain, "Speaker dependent bottleneck layer training for speaker adaptation in automatic speech recognition," in *Fifteenth Annual Conference of the International Speech Communication Association*, 2014.

[25] Frantisek Grézl, Martin Karafiát, Stanislav Kontár, and Jan Cernocky, "Probabilistic and bottle-neck features for lvcsr of meetings," in *Proc. ICASSP*. IEEE, 2007, vol. 4, pp. IV–757.

[26] Zhizheng Wu, Cassia Valentini-Botinhao, Oliver Watts, and Simon King, "Deep neural networks employing multi-task learning and stacked bottleneck features for speech synthesis," in *Proc. ICASSP*. IEEE, 2015, pp. 4460–4464.

[27] Hideki Kawahara, Ikuyo Masuda-Katsuse, and Alain De Cheveigne, "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 communication*, vol. 27, no. 3, pp. 187–207, 1999.