An Evaluation of Postfiltering for Deep Learning Based Speech Synthesis with Limited Data

Beáta Lőrincz,[1,2] Maria Nuţu,[1,2] Adriana Stan,[1] Mircea Giurgiu[1]
[1] Communications Department, Technical University of Cluj-Napoca, Romania
[2] Department of Computer Science, "Babeş-Bolyai" University, Cluj-Napoca, Romania
Email: {beata.lorincz, maria.nutu, adriana.stan, mircea.giurgiu}@com.utcluj.ro

***Abstract*—Recently, deep neural network (DNN) based speech synthesis achieved close to human speech quality and became the state-of-the art in the field of text-to-speech (TTS) synthesis systems. However, a major part of its efficiency comes from the use of large quantity of high-quality speech recordings. When this data is not available, other approaches are still preferred.**

This paper evaluates the DNN-based postfiltering of the synthesised speech as a means to increase the quality of DNN-based TTS systems trained on very limited speech resources. 20 different systems are compared objectively using the Mel Cepstral Distortion measure. The systems differ in terms of: training data, network architecture, and training method. Out of the 20 initial systems, 7 are evaluated subjectively in listening tests performed for two different speakers. Results show that even when starting from as little as 5 minutes of speech recordings, the postfiltering process improves the quality of the synthetic speech output. So it can, therefore, be used as a training strategy for TTS systems where sufficient high-quality data is not available.

***Keywords*—statistical-parametric synthesis; limited data; deep neural networks; postfiltering; text-to-speech synthesis; Romanian**

I. Introduction

Recently, Tacotron 2 [1] a text-to-speech (TTS) synthesis system based on deep neural networks (DNN), obtained a Mean Opinion Score (MOS) rating equal to 4.53. This rating is very similar to the MOS score for natural speech (4.58 as reported by the authors of [1]). This result, in conjunction with multiple other studies of DNN-based speech synthesis [2, 3, 4, 5, 6, 7, 8, 9], made this approach the new state-of-the-art paradigm for TTS systems. However, all these systems require large amounts of high quality speech recordings for training–over 20 hours of data from a single speaker for most of the previously cited works. So there is still the issue of obtaining good TTS systems for languages or speakers where data is limited. In this case, there are several approaches, such as that of Lee et al. [10] which grades and filters the available data to maximize the quality of the output. Another interesting study for this scenario is that of Sone et al. [11], which uses a deep relational model to estimate a neural network's parameters from the joint distribution of acoustic and linguistic features.

Yet the most common approach is to fine-tune or adapt a pre-trained model's parameters using data from the target speaker or language [12, 13, 14]. Or to append speaker/language embeddings to the linguistic/acoustic features, so that the model can jointly learn common and discriminative features from the training set [6, 13, 15, 16, 17].

Although not aimed at solving the data limitation problem, the postfilter presented in [18] could be an alternative solution. This postfilter is trained to map the synthetic speech generated by a Hidden Markov Model (HMM) based system into natural samples by using two DNNs, one operating in the Mel cepstral domain, and the other in the spectral domain. Other studies related to this topic are those of Coto-Jimenez and Close [19] and Muthukumar and Black [20]. Coto-Jimenez and Close append a deep neural network with long-short term memory cells as a postfiltering step for HMM-based speech synthesis. Muthukumar and Black also use a recurrent neural network to enhance the output of the Clustergen statistical-parametric synthesiser. [21] presents a speaker-adaptive postfiltering method for statistical parametric speech synthesis using pre-trained models adapted with limited data to new speakers. A postfilter implemented with Generative Adversarial Network (GAN) is proposed by [22] that is used to learn how to discriminate between synthesised and natural speech. If multi-speaker pre-trained models are available, with few shot methods good quality speech can be obtained for the newly added speaker [23, 24]. To the best of our knowledge, there are no methods which postfilter the DNN-based speech synthesis output without adapting existing models to newly added speakers.

Starting from this overview, we address the problem of developing DNN-based speech synthesis systems with limited speech data by employing a post-synthesis neural network trained to learn the mapping between the synthesised acoustic features and the natural speech features. The method builds upon previously published studies, and focuses on an extensive evaluation of several training strategies and network architectures. 20 different systems are trained and analysed objectively. Out of the 20 systems, 7 were selected for a subjective listening test incorporating two different voices. Both the objective and subjective results illustrate that the postfiltering method can be successfully applied for building TTS systems when large quantities of data are not readily available.

II. Postflltering setup

The scope of our study is to determine a DNN-based postfiltering method for the DNN-based synthesis, such that the final speech output of the system is enhanced even when

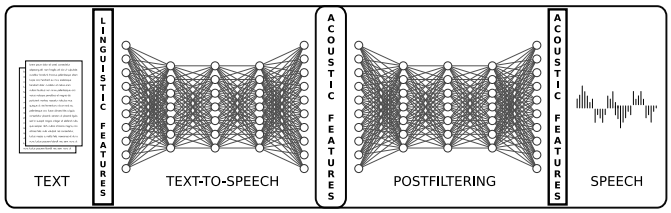

Fig. 1. The postfiltering process.

only limited training data is available. Thus, we employed a two-step procedure: first, a DNN-based TTS system is trained with various amounts of data; and second, the output of the synthesis system is used as input for a postfiltering neural network. An overview of the process is shown in Figure 1.

In DNN-based TTS systems the general trend, nowadays, is to use end-to-end architectures which learn to map raw text-sequences into acoustic representations or waveforms [9]. However, although this training scheme yields very high quality speech output, it is not well suited for the case of limited training data, or real-time synthesis. Therefore, in this study, the synthetic voices are built using the statistical parametric approach. The text is first converted into a set of discrete lexical features, including phonetic transcription, lexical stress assignment, syllabification and part-of-speech tagging, as well as a set of contextual features, such as left and right phonemes, number of syllables and words in a sentence, etc. The complete list of lexical features is based on the common HTS label format [25]. The lexical features are then paired at frame-level to an acoustic parametrization. Phone-level time alignments between text and speech are required, and can be obtained with forced alignment procedures [26].

For the postfiltering step, the entire training dataset prompts are synthesised with the respective TTS system, and the output features are retained. Dynamic Time Warping (DTW) [27] is then applied to align the synthetic and natural feature vectors. The resulting aligned pairs of acoustic features represent the input data for the training of the postfiltering network.

III. Evaluation

A. Data

The training data consists of the RND1 subset of the SWARA Romanian speech corpus [28].[1] Out of the 17 speakers, we chose 8 female ones: *BAS, CAU, EME, DCS, DDM, HTM, PMM* and *SAM*. As the corpus data does not contain purposely built test sets, two other female speakers: *BEA* and *MAR* were additionally recorded and added to the training set. The prompts were the same as for SWARA speakers, and the recordings took place in similar studio conditions. None of the speakers in the combined dataset are professional speakers.

The data is sampled at 48kHz with 16bps, and it was manually segmented at utterance-level. Phoneme state-level alignments were obtained from an iteratively trained HMM-based forced aligner, similar to the first step from the ALISA tool [26]. The aligner used 100 utterances from each speaker. No evaluation of the alignment accuracy was carried out.

[1]Available online: `speech.utcluj.ro/swarasc/`

B. Synthesis systems

The DNN-based TTS systems followed the Blizzard Challenge 2017 Merlin baseline system setup [29, 30]. Linguistic features were derived with an updated version of the Romanian TTS front-end described in [31].[2] Acoustic features were extracted with the WORLD vocoder [32], and comprised 59 plus the 0^{th} Mel generalised coefficients, 5 band aperiodicity coefficients and a fundamental frequency (F_0). The acoustic features were augmented with their delta and delta-delta values. The network architecture consists of 6 layers with 1024 nodes each. The system is trained using the $tanh$ activation function and the stochastic gradient descent optimizer. A separate network with similar architecture is trained to predict the duration of the phoneme states. The postfiltering uses the same set of acoustic features extracted with WORLD, and a baseline network architecture as the one described in [33].

For the evaluation to provide a correct overview over the effectiveness of the postfiltering, we trained 20 different synthesis systems using the *BEA* data. The systems use different training strategies, quantities of training data, and types of postfiltering network architectures. Their details are presented next.

The *training strategy* analyses: simple DNN-based TTS systems trained on linguistic-to-acoustic pairs of features (ID:**M**);[3] TTS system plus DNN postfiltering trained on synthesised-to-natural acoustic feature pairs (ID:**M*_PF**);[4] and DNN speaker adaptation, where an initial eigen voice is trained from the data of all the speakers, and then the network weights are fine tuned for a target speaker (ID:**SPKA**).

The amount of *training data* for the TTS system was set to: 50 utterances (approx. 5 minutes), 100 utterances (approx. 10 minutes), and 500 utterances (approx. 50 minutes). In the postfiltering step we also selected 50, 100 or 500 utterances. The postfiltering utterances were the same as those used to train the correspondent TTS system. The utterances are random newspaper sentences, and they are not phonetically or acoustically balanced or filtered. To overcome the lack of data, we also used an artificial data enhancement method, in which the original speech samples were added twice to the training set, thus doubling the training data (ID:**Db**). This method was applied either for just the postfiltering network, or for both the TTS system and the postfiltering (ID:**M*Db_P*Db**).

In this study, for the postfiltering, only a feed-forward *network architecture* was considered. However, the number of layers (4, 5 and 6), the activation function ($tanh$ and $ReLU$), and the number of neurons per layer (256, 1024, and layer halving or bottleneck: 1025-512-256-512-1024) were examined.

For *speaker adaptation*, different volumes of data from each speaker were used to train the eigen voice (ID:**SPKA*_E***): 100 utterances which translates into 1000 total utterances for training, and 500 utterances from each speaker, 5000 in total.

[2]Online demo: `www.romaniantts.com`

[3]The ID refers to the system ID used in Table I.

[4]The asterisk ('*') marks a variable value.

The network's adaptation was then performed with either 100 or 500 utterances from the target speaker. For postfiltering a network trained on the same 100 utterances from each of the 10 speakers (ID:***MSPK**) was evaluated. This system is similar to the speaker adaptation strategy, in the sense that the network is trained on multi-speaker data. However, no aposteriori weight tuning was performed, and no speaker embeddings were used as features.

Table I summarises the systems' description and the IDs selected for the objective and the subjective evaluations.[5] Audio samples from all the systems are available at: `speech.utcluj.ro/pf_is2020/`.

C. Listening test setup

Although many studies have been conducted on the objective analysis of the synthesised speech quality [34], there are still no measures which truly correlate to the perceptual evaluation of the synthesised speech. Hence, subjective listening tests are required. In this evaluation, as the number of initial systems is quite large for a listening test, the 7 most relevant systems were selected and tested with two different voices. The systems and their listening test identifiers are shown in Table I.

The lower bound of our setup is M050 (A)–the TTS system trained on 50 utterances (approx. 5 mins). The upper bounds are M500 (G) trained on 500 utterances (approx. 50 mins) and the natural (H) samples. System M100 (B) is our baseline for the postfiltering process. Out of the various postfilter network architectures, the 6 $tanh$ layers of 1024 nodes each (M100_PF100_6TANH1024) exhibited the best objective score for both speakers (see Section IV), and they were included for the evaluation of the postfiltering effect alone (C). Artificially doubling the data in both voice training and postfiltering also showed an increase of the objective score, so that system (M100Db_PF100Db) was selected, as well. As the multi-speaker network could be viewed as an eigen postfilter, systems M100_PF_MSPK and SPKA100_E100 were included for the multi-speaker setup comparison.

The listening test comprised 4 sections: a) *Naturalness*–evaluated using a Mean Opinion Score (MOS) scale consisting of 5 points [1-Unnatural, 5-Natural]; b) *Speaker similarity*–evaluated on a 5-point MOS scale [1-Not similar at all, 5-Very similar]; c) *Intelligibility*–evaluated using a Word Error Rate (WER) measure; and d) *ABX naturalness*–each system was randomly paired with all other systems and listeners had to mark which sample sounds more natural.

IV. Results

A. Objective measures

The systems' performance is objectively measured with Mel Cepstral Distortion (MCD) [35]. Because the accuracy of the state-level alignment is unknown, the MCD value was obtained over the best path in DTW, and it does not take into account the 0^{th} coefficient. 50 utterances not contained in the training dataset were synthesised and used to compute the average MCD for speakers *BEA* and *MAR*. *MAR* speaker's distortion included only the listening test systems. Figure 2 shows these results.

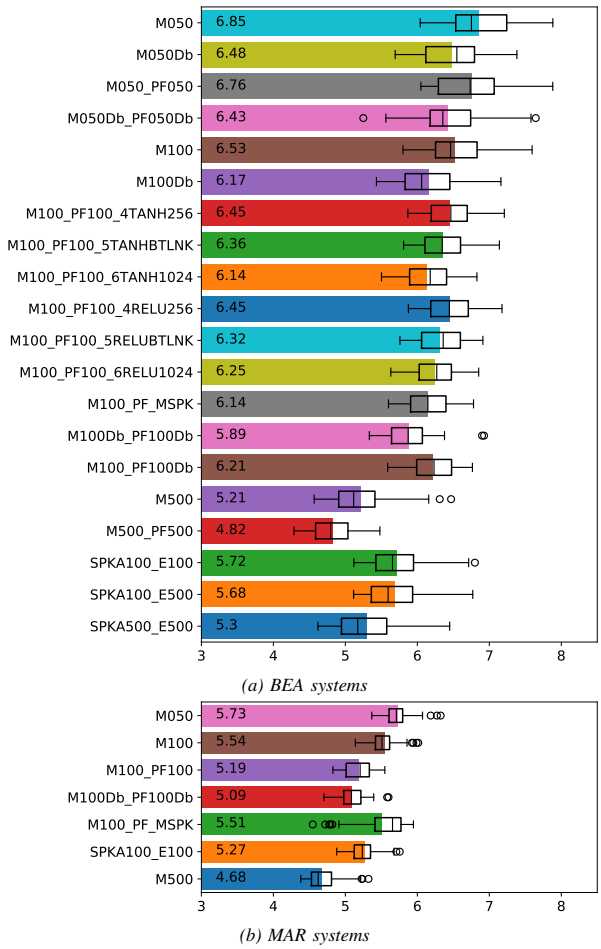

Fig. 2. Average Mel Cepstral Distortion for the (a) **BEA** and (b) **MAR** systems. Horizontal bars represent the mean MCD values, and are overlapped with boxplots.

As expected, out of the baseline TTS systems, M500 performed the best and M050 the worst. M100's scores are quite low, but artificially doubling the data increases the quality of the synthesis (M050Db, M100Db). The postfiltering also decreases the cepstral distortion relative to the correspondent TTS (M050_PF050, M100_PF100, M100Db_PF100Db, M500_PF500). The average decrease in MCD is 5%, with a maximum of 7.5% for M500_PF500. Postfiltering plus data doubling has the most effect (M050Db_PF050Db, M100Db_PF100Db), with a 10% decrease in MCD for M100Db_PF100Db. Doubling the data for the postfiltering alone (M100_PF100Db) only marginally decreases the MCD. With respect to the postfilter network architecture, the 6 $tanh$ layers with 1024 nodes per layer (M100_PF100_6TANH1024) had the best performance. All other network architectures have higher MCD scores, yet not significantly higher. When

[5]Different IDs are used in the objective evaluation as it is easier to follow the multiple setups.

TABLE I
SYNTHESIS SYSTEMS' DESCRIPTION

No.	System ID	Listening test ID	No. utts voice training	No. utts postfiltering	Postfiltering architecture
1	**NAT**	H	Natural	N/A	N/A
2	**M050**	A	50	N/A	N/A
3	**M050Db**	-	50x2	N/A	N/A
4	**M100**	B	100	N/A	N/A
5	**M100Db**	-	100x2	N/A	N/A
6	**M500**	G	500	N/A	N/A
7	**M050_PF050**	-	50	50	6 TANH x 1024
8	**M050Db_PF050Db**	-	50x2	50x2	6 TANH x 1024
9	**M100_PF100_4TANH256**	-	100	100	4 TANH x 256
10	**M100_PF100_5TANHBTLNK**	-	100	100	5 TANH (1024-512-256-512-1024)
11	**M100_PF100_6TANH1024**	C	100	100	6 TANH x 1024
12	**M100_PF100_4RELU256**	-	100	100	4 RELU x 256
13	**M100_PF100_5RELUBTLNK**	-	100	100	5 RELU (1024-512-256-512-1024)
14	**M100_PF100_6RELU1024**	-	100	100	6 RELU x 1024
15	**M100_PF_MSPK**	E	100	10x100 Multi-speaker	6 TANH x 1024
16	**M100Db_PF100Db**	D	100x2	100x2	6 TANH x 1024
17	**M100_PF100Db**	-	100	100x2	6 TANH x 1024
18	**M500_PF500**	-	500	500	6 TANH x 1024
			No. utts for eigen voice		**No. utts for target speaker**
19	**SPKA100_E100**	F	10x100		100
20	**SPKA100_E500**	-	10x500		100
21	**SPKA500_E500**	-	10x500		500

multiple speakers are available, speaker adaptation is indeed a solution: systems SPKA100_E100, SPKA100_E100, SPKA_E500 have some of the lowest MCD scores. However, the multi-speaker postfilter (M100_PF_MSPK) is comparable only to the speaker-dependent filter.

B. Listening tests

The 7 selected systems, along with natural speech samples were included in two separate listening tests: one for speaker *BEA*, and one for speaker *MAR*. Each voice was evaluated by 20 native Romanian listeners. A couple of listeners misread the MOS scale, and their results were discarded.

Figure 3 shows the results. The best performing system (G) is considered the baseline synthesis system as it uses the most amount of data (approx. 50 minutes). The other systems analysed are of higher interest in the evaluation as they use approximately 10 minutes or less data. It can be observed that the naturalness and the speaker similarity are improved by the postfiltering (C) for both speakers. Artificially doubling the data (D) enhances the output speech's naturalness, but not the speaker similarity. However, the intelligibility is affected by the postfiltering in all setups, and slightly improved by the data doubling. The multi-speaker postfiltering network (E) has similar effects as the speaker dependent postfiltering in terms of naturalness. But it is interesting to notice that when it comes to the speaker similarity section, the network trained with multi-speaker data performs better than the speaker dependent one. In the ABX section, the preference over each systems is incremental, with a minor exception for *MAR*'s system D.

C. Discussions

Both the objective and the subjective results showed that postfiltering and artificial data doubling have beneficial effects over the quality of the synthesised output, and can be jointly used in scenarios where the training speech data is insufficient. The effect of the postfilter can be interpreted as a result of the fact that as opposed to the TTS network, it only needs to learn a mapping of vectors which are sampled from similar feature spaces. So it actually learns where the TTS system failed with respect to the natural sample, and not to the lexical input. Artificially doubling the data is useful especially in the DNN setup. Here, the training is done at batch-level, and a global overview of the entire dataset is not available to the learning mechanism at each step. As the batches are not selected sequentially, having more samples to learn from can improve the output. Similar to the data doubling, the high sampling frequency (48kHz) also provides more data points. This was also useful for HMM-based synthesis [31]. The fact that the eigen-postfilter was rated higher in the speaker similarity test, could be a result of a better modelling of the speech characteristics in general, and not of the target speaker in particular.

By listening to the samples, there are some interesting observations to be made. Many of the voiced/unvoiced decision errors of the TTS system were corrected by the postfilter. Also, the buzziness of the TTS speech is noticeably reduced. However, the postfilter makes the speech more metallic-sounding, and it could translate into the drop in intelligibility. The decrease of intelligibility is not at all desired, especially in the case of limited training data, and it is already the focus of our next studies.

V. CONCLUSIONS

This paper described an evaluation of a DNN-based postfiltering method for DNN generated speech using limited resources. The postfilter is trained on pairs of synthetic-to-natural acoustic features, and used to enhance the output of DNN-based TTS system trained on the same data. Starting from as little as 10 minutes of speech from one speaker, this processing chain improves the output synthetic speech as

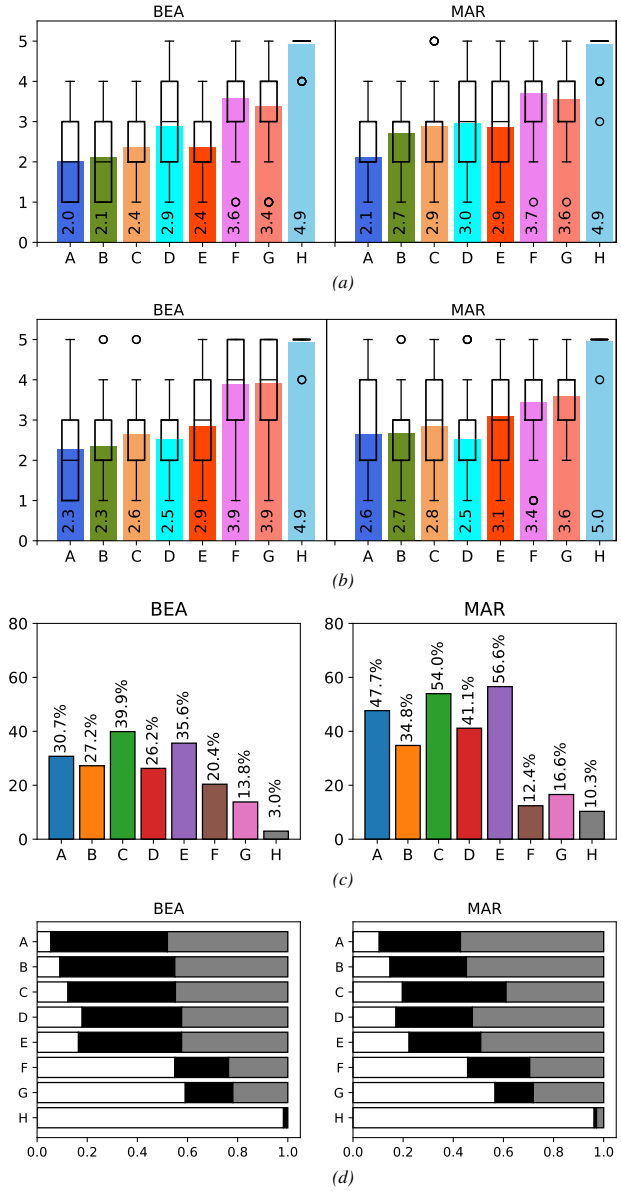

Fig. 3. Listening test results for speakers **BEA** and **MAR**: (a) Naturalness MOS scores, (b) Speaker similarity MOS scores, (c) Intelligibility WER, and (d) ABX preference. In (a) and (b) is fedbars represent the mean value with boxplots overlapped. In (c) bars represent the average WER. In (d) the horizontal bars represent the preference for one system against all others, no preference, or preference for any of the other systems.

evaluated objectively with MCD, and subjectively in listening tests. A downside of this process at this point is the drop in intelligibility, which can be caused by the metallic speech characteristic introduced by the postfilter, and it needs to be investigated further.

For future work, we still need to study other network topologies, as well using other vocoders, or adding additional features to the postfilter, such as lexical or speaker embeddings. In the multi-speaker postfilter, we also need to analyse the weight tuning for the target speaker. Using the correct state-level alignments also needs to be considered. This is important for a direct mapping of synthetic-to-natural features. Also, male speaker voices were not evaluated, and might exhibit a different behaviour.

ACKNOWLEDGMENTS

This work was funded through a grant from the Romanian Ministry of Research and Innovation, PCCDI – UEFISCDI, project number PN-III-P1-1.2-PCCDI-2017-0818/73. We would also like to thank our listening test volunteers.

REFERENCES

[1] J. Shen, R. Pang, R. J. Weiss, M. Schuster, N. Jaitly, Z. Yang, Z. Chen, Y. Zhang, Y. Wang, R. J. Skerry-Ryan, R. A. Saurous, Y. Agiomyrgiannakis, and Y. Wu, "Natural TTS synthesis by conditioning wavenet on mel spectrogram predictions," *CoRR*, vol. abs/1712.05884, 2017. [Online]. Available: http://arxiv.org/abs/1712.05884

[2] A. van den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu, "WaveNet: A Generative Model for Raw Audio," in *arXiv*, 2016. [Online]. Available: https://arxiv.org/abs/1609.03499

[3] A. van den Oord, Y. Li, I. Babuschkin, K. Simonyan, O. Vinyals, K. Kavukcuoglu, G. van den Driessche, E. Lockhart, L. C. C. Rus, F. Stimberg, N. Casagrande, D. Grewe, S. Noury, S. Dieleman, E. Elsen, N. Kalchbrenner, H. Zen, A. Graves, H. King, T. Walters, D. Belov, and D. Hassabis, "Parallel WaveNet: Fast High-Fidelity Speech Synthesis," Google Deepmind, Tech. Rep., 2017. [Online]. Available: https://arxiv.org/abs/1711.10433

[4] Y. Wang, R. Skerry-Ryan, D. Stanton, Y. Wu, R. J. Weiss, N. Jaitly, Z. Yang, Y. Xiao, Z. Chen, S. Bengio, Q. Le, Y. Agiomyrgiannakis, R. Clark, and R. A. Saurous, "Tacotron: Towards End-to-End Speech Synthesis," in *Proc. Interspeech*, 2017. [Online]. Available: https://arxiv.org/abs/1703.10135

[5] S. Ö. Arik, M. Chrzanowski, A. Coates, G. Diamos, A. Gibiansky, Y. Kang, X. Li, J. Miller, J. Raiman, S. Sengupta, and M. Shoeybi, "Deep Voice: Real-time Neural Text-to-Speech," *CoRR*, vol. abs/1702.07825, 2017. [Online]. Available: http://arxiv.org/abs/1702.07825

[6] S. Ö. Arik, G. F. Diamos, A. Gibiansky, J. Miller, K. Peng, W. Ping, J. Raiman, and Y. Zhou, "Deep Voice 2: Multi-Speaker Neural Text-to-Speech," *CoRR*, vol. abs/1705.08947, 2017. [Online]. Available: http://arxiv.org/abs/1705.08947

[7] W. Ping, K. Peng, A. Gibiansky, S. Ö. Arik, A. Kannan, S. Narang, J. Raiman, and J. Miller, "Deep Voice 3: 2000-Speaker Neural Text-to-Speech," *CoRR*, vol. abs/1710.07654, 2017. [Online]. Available: http://arxiv.org/abs/1710.07654

[8] S. Mehri, K. Kumar, I. Gulrajani, R. Kumar, S. Jain, J. Sotelo, A. C. Courville, and Y. Bengio, "SampleRNN: An Unconditional End-to-End Neural Audio Generation Model," *CoRR*, vol. abs/1612.07837, 2016. [Online]. Available: http://arxiv.org/abs/1612.07837

[9] J. Sotelo, S. Mehri, K. Kumar, J. F. Santos, K. Kastner, A. Courville, and Y. Bengio, "Char2Wav: End-to-end speech synthesis," in *International Conference on Learning Representations (Workshop Track)*, April 2017.

[10] K.-Z. Lee, E. Cooper, and J. Hirschberg, "A comparison of speaker-based and utterance-based data selection for text-to-speech synthesis," in *Proc. Interspeech 2018*, 2018, pp. 2873–2877. [Online]. Available: http://dx.doi.org/10.21437/Interspeech.2018-1313

[11] K. Sone and T. Nakashika, "DNN-based Speech Synthesis for Small Data Sets Considering Bidirectional Speech-Text Conversion," in *Proc. Interspeech 2018*, 2018, pp. 2519–2523. [Online]. Available: http://dx.doi.org/10.21437/Interspeech.2018-1460

[12] Y. Fan, Y. Qian, F. Soong, and L. He, "Speaker and language factorization in DNN-based TTS synthesis," in *Proc. of ICASSP*, 03 2016, pp. 5540–5544.

[13] S. Ö. Arik, J. Chen, K. Peng, W. Ping, and Y. Zhou, "Neural Voice Cloning with a Few Samples," *CoRR*, vol. abs/1802.06006, 2018. [Online]. Available: http://arxiv.org/abs/1802.06006

[14] Z. Huang, H. Lu, M. Lei, and Z. Yan, "Linear networks based speaker adaptation for speech synthesis," *arXiv e-prints*, p. arXiv:1803.02445, Mar 2018.

[15] I. Demirsahin, M. Jansche, and A. Gutkin, "A Unified Phonological Representation of South Asian Languages for Multilingual Text-to-Speech," in *Proc. The 6th Intl. Workshop on Spoken Language Technologies for Under-Resourced Languages*, 2018, pp. 80–84.

[16] B. Li and H. Zen, "Multi-Language Multi-Speaker Acoustic Modeling for LSTM-RNN based Statistical Parametric Speech Synthesis," in *Proc. of Interspeech*, 2016.

[17] Y. Lee, T. Kim, and S. Lee, "Voice Imitating Text-to-Speech Neural Networks," *CoRR*, vol. abs/1806.00927, 2018. [Online]. Available: http://arxiv.org/abs/1806.00927

[18] L.-H. Chen, T. Raitio, C. Valentini-Botinhao, Z.-H. Ling, and J. Yamagishi, "A deep generative architecture for postfiltering in statistical parametric speech synthesis," *IEEE/ACM Trans. Audio, Speech and Lang. Proc.*, vol. 23, no. 11, pp. 2003–2014, Nov. 2015. [Online]. Available: http://dx.doi.org/10.1109/TASLP.2015.2461448

[19] M. Coto-Jiménez and J. G. Close, "LSTM deep neural networks postfiltering for improving the quality of synthetic voices," *CoRR*, vol. abs/1602.02656, 2016. [Online]. Available: http://arxiv.org/abs/1602.02656

[20] P. K. Muthukumar and A. W. Black, "Recurrent neural network postfilters for statistical parametric speech synthesis," *CoRR*, vol. abs/1601.07215, 2016. [Online]. Available: http://arxiv.org/abs/1601.07215

[21] M. G. Öztürk, O. Ulusoy, and C. Demiroglu, "Dnn-based speaker-adaptive postfiltering with limited adaptation data for statistical speech synthesis systems," in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2019, pp. 7030–7034.

[22] T. Kaneko, H. Kameoka, N. Hojo, Y. Ijima, K. Hiramatsu, and K. Kashino, "Generative adversarial network-based postfilter for statistical parametric speech synthesis," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017, pp. 4910–4914.

[23] S. Arik, J. Chen, K. Peng, W. Ping, and Y. Zhou, "Neural voice cloning with a few samples," in *Advances in Neural Information Processing Systems*, 2018, pp. 10 019–10 029.

[24] Y. Jia, Y. Zhang, R. Weiss, Q. Wang, J. Shen, F. Ren, P. Nguyen, R. Pang, I. L. Moreno, Y. Wu *et al.*, "Transfer learning from speaker verification to multispeaker text-to-speech synthesis," in *Advances in neural information processing systems*, 2018, pp. 4480–4490.

[25] K. Tokuda, H. Zen, and A. Black, "An HMM-Based Speech Synthesis System Applied To English," in *Proc. of SSW*, 10 2002, pp. 227 – 230.

[26] A. Stan, Y. Mamiya, J. Yamagishi, P. Bell, O. Watts, R. Clark, and S. King, "ALISA: An automatic lightly supervised speech segmentation and alignment tool," *Computer Speech and Language*, vol. 35, pp. 116–133, 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0885230815000650

[27] S. Salvador and P. Chan, "Toward accurate dynamic time warping in linear time and space," *Intell. Data Anal.*, vol. 11, no. 5, pp. 561–580, Oct. 2007.

[28] A. Stan, F. Dinescu, C. Tiple, S. Meza, B. Orza, M. Chirila, and M. Giurgiu, "The SWARA Speech Corpus: A Large Parallel Romanian Read Speech Dataset," in *Proceedings of the 9th Conference on Speech Technology and Human-Computer Dialogue (SpeD)*, Bucharest, Romania, July, 6-9 2017.

[29] S. King, L. Wihlborg, and W. Guo, "The Blizzard Challenge 2017," in *Proc. Blizzard 2017*, Stockholm, Sweden, September 2017.

[30] Z. Wu, O. Watts, and S. King, "Merlin: An Open Source Neural Network Speech Synthesis System," in *9th ISCA Speech Synthesis Workshop (2016)*, Sep. 2016, pp. 218–223.

[31] A. Stan, J. Yamagishi, S. King, and M. Aylett, "The Romanian speech synthesis (RSS) corpus: Building a high quality HMM-based speech synthesis system using a high sampling rate," *Speech Communication*, vol. 53, no. 3, pp. 442–450, 2011.

[32] M. Morise, F. Yokomori, and K. Ozawa, "WORLD: A Vocoder-Based High-Quality Speech Synthesis System for Real-Time Applications," *IEICE Transactions*, vol. 99-D, pp. 1877–1884, 2016.

[33] T. Toda, L.-H. Chen, D. Saito, F. Villavicencio, M. Wester, Z. Wu, and J. Yamagishi, "The Voice Conversion Challenge 2016," in *Interspeech 2016*, 2016, pp. 1632–1636.

[34] R. Ullmann, R. Rasipuram, M. Magimai.-Doss, and H. Bourlard, "Objective intelligibility assessment of text-to-speech systems through utterance verification," Idiap, Idiap-RR Idiap-RR-06-2015, 4 2015.

[35] R. Kubichek, "Mel-cepstral distance measure for objective speech quality assessment," in *Proceedings of IEEE Pacific Rim Conference on Communications Computers and Signal Processing*, vol. 1, May 1993, pp. 125–128 vol.1.