Multi-Speaker Neural Vocoder

Oriol Barbany[1,2]*, Antonio Bonafonte*[1] *and Santiago Pascual*[1]

[1]Universitat Politècnica de Catalunya, Barcelona, Spain
[2]École Polytechnique Fédérale de Lausanne, Lausanne, Switzerland
oriol.barbanymayor@epfl.ch, antonio.bonafonte@upc.edu, santi.pascual@upc.edu

Abstract

Statistical Parametric Speech Synthesis (SPSS) offers more flexibility than unit-selection based speech synthesis, which was the dominant commercial technology during the 2000s decade. Although offers more flexibility and doesn't need the whole speech database in deployment, classical systems have lower naturalness than unit-selection methods. Deep learning have outperformed SPSS thanks to recurrent architectures by offering high quality speech while preserving the desired flexibility in choosing the parameters such as the speaker, the intonation, etc. This paper exposes two proposals conceived to improve Deep Learning-based Text-to-Speech systems. The baseline model, obtained by adapting SampleRNN, was able to generate voice from different speakers with one single model and after implementing a combination of the two approaches that will be discussed in this paper, state-of-the-art results were obtained. The first proposal differs from typical feature normalizations that don't consider the origin of such features, which could have intrinsic differences like in the case of modeling various speakers with the same network. This is aimed to obtain acoustic features with similar values across the speakers, e.g. men have lower pitch than woman but the variation of it in a given utterance, i.e. the intonation, is not so different in both cases. Thanks to this, the network can more easily model the patterns of these features in all the speakers with a single model. The second proposal, named as look ahead, consists in feeding information of future frames to the network with the aim of better modeling the speech signal and avoid possible discontinuities. Human listeners prefer the system that combines both techniques, which reaches a rate of 4 in the Mean Opinion Score scale (MOS) with the balanced dataset and outperforms the other models.

Index Terms: Deep Learning, Speech synthesis, Recurrent Neural Networks, Text-to-Speech, SampleRNN, Time Series

1. Introduction

Deep learning has revolutionized almost every engineering branch over the past decades and have also been successfully applied to Text-to-Speech (TTS), where it yields state-of-the-art performance and overcomes classical approaches. The time series problem have been completely leveraged by Recurrent Neural Networks (RNNs) and their variants, which make them lead to interesting results in the speech synthesis field. Moreover, deep generative models can generate speech sample by sample as first proposed in Wavenet [1], which achieved very fine-grained waveform amplitudes and outperformed previous Statistical Parametric Speech Synthesis (SPSS) models. This paper exposes two of the proposals presented in the bachelor's thesis of the main author [2] that were applied to better model the speech generated with a Multi-Speaker Deep Learning-based model.

Deep learning have outperformed classical systems by offering high quality speech while preserving the flexibility of SPSS systems. To achieve a state-of-the-art TTS system, SampleRNN [3] was adapted to generate coherent speech in Spanish attributable to different speakers. The main motivation of this work was to generalize the system proposed in [4] for many speakers as a shared Deep Neural Network (DNN) structure because it achieves better results in generating quality speech than learning the parameters of a single isolated speaker [5]. In this case, the learning of the base shared structure can be transferred for a new speaker to achieve speaker adaptation like in [6] with limited training data, achieving good results in naturalness and similarity to the original speaker. The proposals were initially conceived to improve the speech obtained with a deep generative network able to model multiple speakers with the same structure. Nevertheless, the speaker-dependent normalization could be used as a new preprocessing technique in a variety of problems and the look ahead approach can be generalized to time series modeling. Current state-of-the art TTS models like WaveNet [1], Tacotron [7] or VQ-VAE [8] already model several speakers with a unique model, but doesn't apply speaker-dependent normalizations, which is shown to deteriorate results. The look ahead proposal is not either mentioned, but it outperformed our baseline model and could be applied to other time series modeling system.

The first proposal was aimed to perform voice conversion, which is a technique to modify a speech waveform which freely converts non-para-linguistic information while preserving linguistic information [9]. This means that intonation, pauses and spoken text is exactly the same but the speaker changes. The acoustic features fed to the network and used to condition the generated speech, such as pitch or Mel-Frequency Cepstral Coefficients (MFCC), depend on the speaker, which means that the input to indicate the speaker identity is redundant. This redundancy is identified by the network, which assigns low weights to the speaker identity and make it irrelevant. Nevertheless, this input is needed for voice conversion to choose the desired speaker. The speaker-dependent normalization aims to give importance to the speaker identity by isolating the features from the speaker and thus forcing the network to use the speaker identity to generate natural speech for each user.

The look ahead approach questions the causality of time series modeling, which is not needed unless input features are extracted at real time and therefore not known beforehand. In the case of TTS systems, the text which will be uttered is known before hand and therefore, the acoustic features of all the signal are known. Moreover, in natural speech, the phonemes sound different depending on the context and thus can change depending on future phonemes (co-articulation). By giving information of the future behavior of the predicted sequence, there are no discontinuities and artifacts are reduced. This is translated into better quality speech as rated by human listeners.

These two proposals derive in four different configurations

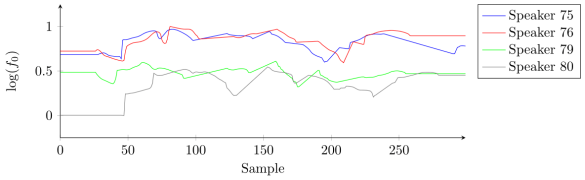

Figure 1: *Classical speaker-independent normalization*

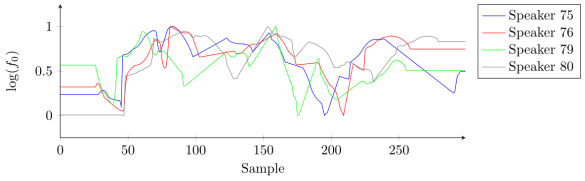

Figure 2: *Proposed speaker-dependent normalization*

that will be further explored and compared in this work. As rated by the users, speaker-dependent normalization achieve substantially better results in naturalness for the generation of speech modeled with balanced datasets when combined with the look ahead approach. Regarding the second proposal, it outperforms the previously achieved scores and reaches state-of-the-art results when combined with the speaker-dependent normalization.

2. Proposals

2.1. Speaker-dependent feature normalization

Features fed to a neural network are often previously normalized to control the magnitude of both the activations and gradients in training. With the hypothesis of having speaker-dependent features, an independent normalization for each of the speakers was proposed to isolate the speech features from the source. Maximum and minimum values for each of the parameters were found within the training partition so it could happen that some features of the train or validation partitions overpass the bounds. The chosen normalization function was a simple feature scaling that follow equation (1), which bound each of the features from 0 to 1. This approach could be also applied with other normalization functions like the $z-$score, i.e. statistical normalization. This last option was not tested before the writing of this paper due to the low improvement in results of this only modification (see table 1). Nevertheless, as it can be seen in the same table, this approach outperforms the other models when combined with the look ahead approach (explained in section 2.2). Therefore, a statistical normalization could also be tested in future work.

$$\hat{\boldsymbol{x}} = \frac{\boldsymbol{x} - \boldsymbol{x}_{\min}}{\boldsymbol{x}_{\max} - \boldsymbol{x}_{\min}} \tag{1}$$

This proposal aims to give importance to the speaker identity to ideally allow voice conversion without the need of a complex mapping of features. Inspiration came from the behavior of the pitch for every speaker, which is depicted for both speaker-independent and speaker-dependent normalizations in figures 1 and 2 respectively. These plots illustrate the evolution of the logarithmic fundamental frequency for four different speakers including two males and two females that read the exact same text and thus are very similar once normalized following a speaker-dependent approach. Note that there is some time shifting due to different duration of phonemes and pauses but the signal is yet very similar.

After the classical speaker-independent normalization (figure 1), it is very easy to distinguish between females (75, 76) and males (79, 80). This means that it would be impossible to perform voice conversion because the network doesn't need the speaker identifier for being this information implicit in the features. This is why this redundancy is translated into the futility of this input observed when trying to change the speaker identity at will. The behavior of the pitch once normalized by speaker is very similar if the intonation is comparable. Nevertheless, the other features that are fed to the network (see next section) resulted in very similar normalizations for both speaker-independent and speaker-dependent approaches.

2.2. Look ahead

In the modeling of non-real-time sequences such as the generation of speech in a TTS system, the features that will be fed to the network are known beforehand. This means that, in contrast with a possible phone call where both ends are talking at real time, the features that will condition the sequence at future time steps are always known and thus can be used to better model the generated signal.

With this idea in mind, the causality that speech synthesizers inherited from the vocoders used in decoding is questioned and both the current and future windows of features are fed to the network. This results in a larger model because the number of features is duplicated at each time step but also achieves better quality without the need of more features.

3. Experimental setup

The speech dataset used to train the model was formed by six Spanish voices from the TC-STAR project [10], where half of them are males and the other half are females. The database was unbalanced with one of the female speakers barely having a quarter of speech recording time compared to the others. Despite in some works like [11], it is recommended to balance the data per user so all of them have approximately the same amount of samples to train, it was chosen to use all the available data to avoid restricting all the speakers to only 14 minutes of speech instead of an hour. The total duration of the whole dataset including the six speakers was of 5.25 hours, which was divided into 80% for training, 10% for validation and 10% for test.

3.1. SampleRNN

The baseline model was SampleRNN, an unconditional end-to-end neural audio generation model [3] that consists of two recurrent modules running at different clock rates that aim to model the short and long term dependencies of speech signals, and one module with auto-regressive Multi-Layer Perceptrons (MLPs) that process speech sample by sample. The recurrent architecture used for this model is the Gated Recurrent Unit (GRU) [12], which differ from the implementation with Long Short-Term Memory (LSTM) modules proposed by the authors of SampleRNN. The three tier architecture provides flexibility in allocating the amount of computational resources for modeling different levels of abstraction and results very efficient in memory during training. The final output of SampleRNN model is the probability of the current sample value conditioned on all the previous values of the sequence that can be expressed following the chain rule as stated in equation (2). The output

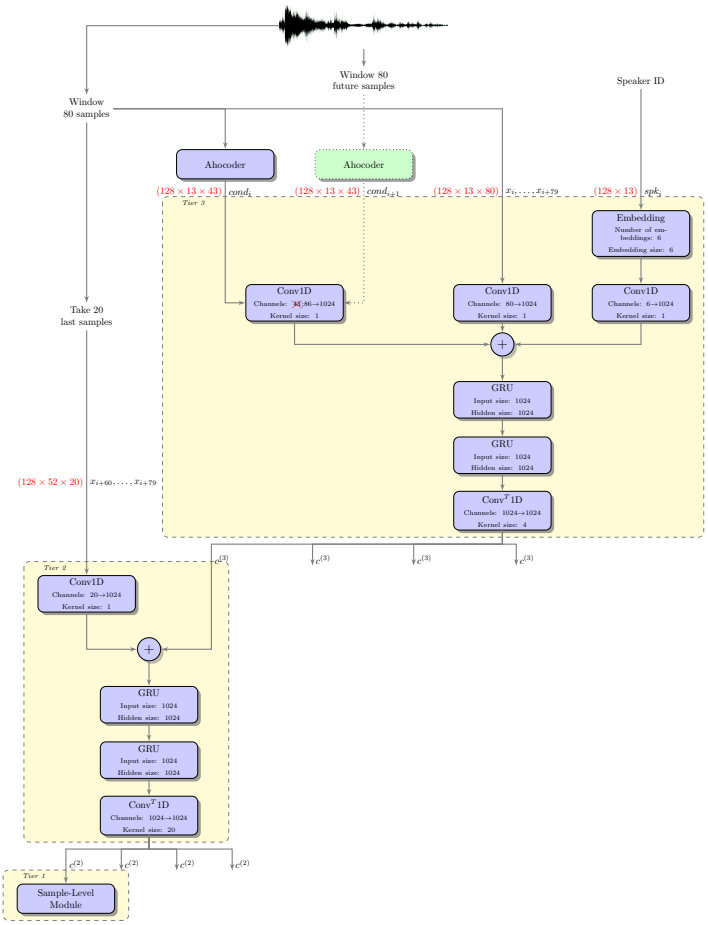

Figure 3: *SampleRNN multi-speaker adaptation*

follows a Multinoulli distribution, which could be unintuitive due to the naturalness of speech signals, which are real-valued, but achieves better results as it doesn't assume any distribution of the data and thus can more easily model arbitrary distributions. In this work, speech samples are quantized with 8 bits, having therefore 256 possible values. Differing from the linear quantization proposed in SampleRNN, this work applies a $\mu-$law companding transformation [13] before classifying into the 256 possible classes to flatten the Laplacian-like distribution of speech signals.

$$P(\boldsymbol{X}) = \prod_{t=1}^{T} P(x_t | xi_1, \ldots, x_{t-1}) \tag{2}$$

In order to generate coherent speech, the model was conditioned like in [4] with acoustic features obtained with Ahocoder [14], a high-quality harmonics-plus-noise vocoder that predicts a set of features that can characterize speech signals. The adapted model is depicted in figure 3.

Note that the look ahead approach modifies the architecture because the upper right 1D-Convolution block doubles its input size (the original value of 43 is crossed out in the figure and replaced by 86 to accept both the features of the current and future frames). Differing from the original SampleRNN model [3] and apart from the previously mentioned addition of the acoustic conditioners that allow to synthesize coherent speech, the authors also incorporated the blocks in the left of the figure. These are aimed to differentiate between all the speakers of the database and thus compute an embedding from the identifier which is also used to condition the model along the aforementioned features extracted with Ahocoder.

The acoustic parameters are extracted in frames of 15ms shifted every 5ms and include:

- 40 Mel-cepstral coefficients
- Maximum Voiced Frequency (fv)
- Logarithmic F0 value
- Voiced/Unvoiced flag (uv)

The pitch contour signal behaves very different for voiced and unvoiced signals. In the first case, it is a continuous signal and in the second case, it does not exist and Ahocoder represents it as -10^{10}. To tackle this discontinuity in statistics that could degrade the model by these outliers, the output of Ahocoder is post-processed and the pitch contour is interpolated to be log-linear for the unvoiced segments.

These features are then scaled following either the proposed speaker-dependent or the classical speaker-independent normalization. The normalized features are then rearranged to match the speech samples used in training and a speaker identifier is added as an independent input (see figure 3) to the system. This is aimed to both better model each of the different voices and perform voice conversion.

The conditioned model outputs a distribution that not only depends on the previous samples but also on the features obtained with Ahocoder and on the speaker identity. Therefore, the expression of the adapted model follows equation (3), where $\boldsymbol{l}_t$ stands for a 49-dimensional vector that results of an embedding of dimension 6 representing the identity of the speaker and the 43-dimensional acoustic vector corresponding to the analysis window of the current sample x_t. Note that in the case of following a look ahead approach, $\boldsymbol{l}_t$ would be a 92-dimensional vector because of the doubling of the acoustic vector size.

$$P(\boldsymbol{X}|\boldsymbol{L}) = \prod_{t=1}^{T} P(x_t|xi_1, \ldots, x_{t-1}, \boldsymbol{l}_t) \tag{3}$$

The learning strategy was to train each of the models derived from the previous proposals with mini-batch Stochastic Gradient Descent (SGD) using a mini-batch size of 128 and minimizing the Negative Log-Likelihood. The chosen optimizer was Adaptive Moment Estimation (ADAM) [15], an SGD algorithm with adaptive learning rate, and an initial value of 10^{-4}, that was enhanced with an external rate controller known scheduler. This had two milestones at epochs 15 and 35. In each of that milestones, the current learning rate decreases by a factor 0.1, which counterattacks the sudden changes in the loss curve that showed up at first epochs. Weight normalization [16] was also used on the 1-D Convolutional layers to speed-up the training.

3.2. Subjective evaluation

A Mean Opinion Score (MOS) test was conducted to get a more in-depth comparison between the four experiments derived of combining the two proposals. MOS rates the naturalness in a scale of natural integers from 1 to 5, meaning these bounds bad and excellent quality respectively. The volunteers that participated in the test, could listen the different recordings as many times as required to compare the systems and rate them. For each sentence, the transcription of the audio was provided to ease the listening, and audios of each of the different systems synthesizing the same sentence, were disposed side by side to compare.

4. Results

Results exposed in table 1 were obtained by averaging the evaluation of 25 volunteers for every system. Some of the comments written by the volunteers who took the test, highlighted the difference in quality between males and females. This is why the following table does a separation between genders that shows that best results are indeed obtained with man voices. This was attributed to the unbalanced speech database, which contained one female speaker with only 25% of recordings compared to all other speakers. Seemingly, this affected in the modeling of female voices, which were more noisy.

The samples belonging to the best system combining both proposals can be heard in the GitHub of the main author[1].

5. Conclusions

Speaker-dependent normalization was not enough for voice conversion purposes, so more complex architectures were proposed in [2]. Nevertheless, human listeners preferred the speech modeled with speaker-dependent normalization and, given the similarity of the features normalized for each speaker, a better quantization could be applied for coding applications or for deployment of neural networks with limited resources.

Whilst the speaker-dependent normalization itself doesn't seem to improve the results obtained with the classical speaker-independent feature scaling, when combined with the look ahead approach, it achieves a 4 score with the male balanced dataset. To sum up, with the combination of the two proposals, a state-of-the-art MOS score have been achieved for a multi-speaker speech synthesis system. Both of that approaches were novelties introduced in this thesis and results show that they could be beneficial to other TTS systems as well as for a bunch of other applications involving features from different sources and modeling of no-real-time sequences.

Normalization:	Spk-D	Spk-Ind	Spk-D	Spk-Ind
Look ahead:	No	No	Yes	Yes
Female:	3.3	3.3	3.8	3.6
Male:	3.6	3.6	4.0	3.8
Total:	3.5	3.5	3.9	3.8

Table 1: *Table with subjective results comparing proposed methods*

[1] https://github.com/Barbany

6. References

[1] 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," *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, pp. 1–15, 2016. [Online]. Available: http://arxiv.org/abs/1609.03499

[2] O. Barbany Mayor, "Multi-Speaker Neural Vocoder," Bachelor's thesis, Universitat Politècnica de Catalunya, 2018.

[3] S. Mehri, K. Kumar, I. Gulrajani, R. Kumar, S. Jain, J. Sotelo, A. Courville, and Y. Bengio, "SampleRNN: An Unconditional End-to-End Neural Audio Generation Model," *ICLR*, pp. 1–11, 2017. [Online]. Available: http://arxiv.org/abs/1612.07837

[4] A. Bonafonte, S. Pascual, and G. Dorca, "Spanish Statistical Parametric Speech Synthesis using a Neural Vocoder," *InterSpeech*, 2018. [Online]. Available: https://www.isca-speech.org/archive/Interspeech_2018/pdfs/2417.pdf

[5] Y. Fan, Y. Qian, F. K. Soong, and L. He, "Unsupervised speaker adaptation for DNN-based TTS synthesis," *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, pp. 4475–4479, 2015.

[6] A. W. Black, H. Zen, and K. Tokuda, "Statistical Parametric Speech Synthesis," *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, pp. 1229–1232, 2007.

[7] Y. Wang, R. J. Skerry-Ryan, D. Stanton, Y. Wu, R. J. Weiss, N. Jaitly, Z. Yang, Y. Xiao, Z. Chen, S. Bengio, Q. V. Le, Y. Agiomyrgiannakis, R. Clark, and R. A. Saurous, "Tacotron: A fully end-to-end text-to-speech synthesis model," *CoRR*, vol. abs/1703.10135, 2017. [Online]. Available: http://arxiv.org/abs/1703.10135

[8] A. van den Oord, O. Vinyals, and K. Kavukcuoglu, "Neural Discrete Representation Learning," in *NIPS*, 2017. [Online]. Available: http://arxiv.org/abs/1711.00937

[9] T. Toda, L. H. Chen, D. Saito, F. Villavicencio, M. Wester, Z. Wu, and J. Yamagishi, "The voice conversion challenge 2016," *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, vol. 08-12-Sept, pp. 1632–1636, 2016.

[10] A. Bonafonte, H. Höge, I. Kiss, A. Moreno, U. Ziegenhain, H. V. D. Heuvel, H. Hain, X. S. Wang, and M. N. Garcia, "TC-STAR : Specifications of Language Resources and Evaluation for Speech Synthesis," *Proceedings of the Language Resources and Evaluation Conference LREC06*, pp. 311–314, 2006.

[11] S. Pascual de la Puente, "Deep learning applied to speech synthesis," Master's thesis, Universitat Politècnica de Catalunya, 2016.

[12] K. Cho, B. van Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation," *CoRR*, 2014. [Online]. Available: http://arxiv.org/abs/1406.1078

[13] ITU-T. Recommendation G. 711, "Pulse Code Modulation (PCM) of voice frequencies," 1988.

[14] D. Erro, I. Sainz, E. Navas, and I. Hernaez, "Harmonics plus Noise Model based Vocoder for Statistical Parametric Speech Synthesis," *IEEE Journal on Selected Topics in Signal Processing*, vol. 8, no. 2, pp. 184–194, 2014.

[15] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *CoRR*, vol. abs/1412.6980, 2014. [Online]. Available: http://arxiv.org/abs/1412.6980

[16] T. Salimans and D. P. Kingma, "Weight normalization: A simple reparameterization to accelerate training of deep neural networks," *CoRR*, vol. abs/1602.07868, 2016. [Online]. Available: http://arxiv.org/abs/1602.07868