Proceedings of 10th IOE Graduate Conference
Peer Reviewed
ISSN: 2350-8914 (Online), 2350-8906 (Print)
Year: 2021 Month: October Volume: 10

Deep Learning Based Voice Conversion Network

Ashok Basnet [a], Basanta Joshi [b], Suman Sharma [c]

[a, b, c] *Department of Electronics and Computer Engineering, Pulchowk Campus, IOE, Tribhuvan University, Nepal*
Corresponding Email: [a] 075mscsk003.ashok@pcampus.edu.np, [b] basanta@ioe.edu.np, [c] suman.sharma@ioe.edu.np

Abstract

Timbre, Content, Rhythm and Prosody are four crucial aspects of speech. These aspects controls how person speaks. When two speaker utters same content, then other three aspects controls the differences in their speech. Converting any person voice to the targeted speaker voice by control over these three parameters is the the main work presented in this paper. Voice synthesized by text-to-speech system if trained on lesser amount of data produces robotic and foggy sound. Producing a new dataset for adding a new speaker on the system is costly. The proposed method is an end-to-end system based on multi-domain google's Wavenet auto-encoder with disentangled latent space and shared encoder trained on Nepali speech dataset. The use of an auto-encoder can remove noise from the audio. The encoder part of the autoencoder transforms audio into the latent space representation whereas decoder side decodes latent space representation back to the voice of the targeted speaker. The rhythm, prosody and timbre of targeted speaker's voice is modified artificially. The network is trained in unsupervised way to recover these modified aspects back to the original speech. Attention mechanism network is supposed to recover the timing of the speaker in order to match the prosody of the targeted speaker. The model is trained on 480 datasets from 17 speakers followed by training on 1500 datasets of single speakers extracted from youtube for 5000 epochs. The correlation of the synthesized audio with the recorded speech of targeted speaker is found to be 0.78. Also the evaluation of quality by mean opinion score results the score of 2.78. Increase in the size of dataset, clarity of recorded audio sample and increase in number of training epochs can further increase the naturalness of the converted speech.

Keywords

Timbre, Content, Rhythm, Prosody, Google's WaveNet, Autoencoder, Latent Space, Attention Mechanism

1. Introduction

Since the evolution of Artificial Intelligence, scientists are constantly looking for a way to make text to speech systems more natural sounding. Release of Google's Wavenet, Deep Neural Network for generating raw audio waveforms in 2016 reported a Mean Opinion Score of 4.21 in TTS for US English which is very close to that of a Human speaker having MOS 4.55.[1] It's a massive reduction in the gap between the state-of-art and human level of performance. As the sound is getting more natural it is being used for the promotion of digital content and gadgets by conveying pleasant voices with a variety of speaker's sound embedded inside. Here, attention mechanism is implemented on WaveNet autoencoder for universal speech conversion. This work aims for converting any speaker voice similar to the voice of the target speaker. WaveNets employs dilated CNN encoders to transform audio into latent semantic space and by the use of powerful audio decoders the latent representations are mapped back to high quality target audio. Universal conversion is achieved by training audio from multiple domains at once and adding domain adaptation terms. Hence, there can be input from any music/audio domain and the obtained output is one among the training domains. Attention components can learn the timing of individual speakers. This component is trained by generating randomly manipulated synthetic signals in the time axis. Hence, the voice encoder, decoder and the attention network are the three major components for universal voice conversion.[2]

The Facebook AI Research team has obtained Music translation from different musical domains into targeted instruments tone and the result obtained was the same level as played by professional musician.[3]

Such conversion plays an important role in the music generation process and can be used in music and other audio processing industries for removing noise and adding different audio effects. It is often hard to produce the voice that exactly matches the voice of the target speaker. Also maintaining the long term dependency in speech signal during voice conversion is a very complicated task. Flite TTS engine has been implemented for voice conversion tasks which is based on the work done by Carnegie Mellon University .[4] The core of this engine is mostly written in C and C++, hence it is fast to run on small computing power. But during the process of voice to voice conversion for generating the voice of different target speakers, it appears the output voice to be foggy and distorted. So, it is our goal to implement Google's Wavenet Based autoencoder for voice conversion. In this work we have experimented with attention mechanisms to obtain universal voice conversion.

2. Related Works

Voice signals can be decomposed into pitch, timbre, content and rhythm. Rhythm distinguishes speakers on the basis of timing of the syllable i.e. how fast he/she utters each syllable. In a spectrogram, the length of the horizontal axis indicates rhythm. Long span indicates the fast speaker whereas short span of axis indicates fast speaker. Pitch is simply the rise and fall of each syllable. Pitch range reflects the identity whether the speaker is male or female. Timber indicates the formant frequencies i.e. the resonant frequency of the vocal tract. Bright voice is indicated by high formant frequency whereas low formant frequencies indicate the deep voice. Content has the smallest unit called phone and each phone has its specific formant pattern. This disentangled representation can be used in speech generation, conversion and analysis tasks. The disentanglement can be carried out using multiple intricately designed information bottlenecks [5].

In an **exemplar-based** voice synthesis network, the labeled speech corpus are stored in a database such that relevant parts can be searched during the inference phase. Here, the query is performed on a large database of stored speech and certain portions of the speech are selected on the basis of how well the specification is matched. Here the speech is directly derived from the recorded voice samples. Using this system, one can only derive the speech which are stored in the database.The quality of output derives directly from the quality of the recordings and it appears that the larger the database, the better the coverage. However, these techniques limit the output speech to the same style as that in the original recordings. In addition, recording large databases with variations is very difficult and costly [6].

Speech synthesis via **Statistical Parameter** obtained attention in the AI community. This model uses parameters such as mean, variance of PDF for capturing the parameters of the training data. It has multilingual support. The quality of speech produced by the initial statistical parametric systems was significantly lower than this of unit-selection systems. Consideration of mentioned three factors are supposed to improve the quality i.e. vocoder, over smoothing during speech inference and the modelling accuracy [7].

Although the Deep Neural Networks can solve difficult learning tasks, it cannot map sequence to sequences. But use of LSTM can map input sequence to a vector of a fixed dimensionality [8]. The main disadvantage of the sequence to sequence model is that it produces a fixed length context vector.

Bahdanau successfully implemented an **attention mechanism** to overcome the problem of needing to represent input sequence by fixed length vector representation in encoder-decoder architecture. The attention mechanisms are frequently used in transduction models, sequence modeling and so on. It allows to model dependencies without regard to the input and output sequence.This context vector is incapable of remembering the sequence once it has processed the sequence. So to overcome this inefficiency a neural network is introduced in between encoder and decoder which serves the attention mechanism. Attention allows the model to focus on the relevant parts of the input sequences as needed for current time-step output. To focus on certain parts of input it looks at a set of hidden states to find the most associated word. Each hidden state is given a score and finally each hidden state is multiplied by its soft maxed score. The reason for using softmax function is to amplify the high score of the hidden state and drown out the low score state. This frees a neural translation model from having to squash all the information of a source sentence, regardless of its length, into a fixed length vector. The model based on attention achieves 28.4 BLEU on the WMT 2014 Englishto-German translation task, improving over the ensembles, by over 2 BLEU [9].

Gaussian Mixture Models are a probabilistic model that represents normally distributed subpopulations within an overall population. It assumes all the data points are generated from a mixture of a finite number of Gaussian distributions with certain parameters. Expectation Maximization algorithm is used to fit GMM to the dataset. It learns the representation of a multimodal data distribution as a combination of unimodal distributions. GMM assumes the data in a specific cluster are generated by a specific Gaussian distribution/component. The GMM based algorithm along with dynamic frequency warping can avoid the over smoothing during speech conversion [10]. Two attention mechanism are popular for TTS system i.e. Dot product based attention mechanism and Gaussian Mixture Model (GMM) based attention mechanism. GMM based monotonic attention mechanisms are stated to be more stable.

WaveNet. WaveNet[1] is an auto regressive system for predicting the probability distributions of data. It can generate raw speech signals with subjective naturalness never before reported in the field of text-to-speech (TTS), as assessed by human raters. The model for raw audio waveform generation needs to deal with long range temporal dependencies and wavenet uses dilated causal convolutions to deal with such dependencies. Also, dilated causal convolutions exhibit very large receptive fields. WaveNet can be used for multi speaker speech generation, text-to-speech as well as on music audio modelling [11]. WaveNet produced speech with MOS of 4.21 for North American English, which is an substantial increase than previous LSTM-RNN parametric method with MOS of 3.67[12] and HMM-driven concatenative approach with MOS of 3.86 [13] in Text-To-Speech application.

3. Model Architecture

The attention based auto-encoder comprises encoder and decoder. Say the number of speakers is k, then the network needs to be trained on k pathways. The network is built in a manner that all paths share a single encoder with k decoder. All domain specific decoder outputs are passed to the softmax reconstruction loss individually during training. The shared encoder if not trained adversarially then there is the possibility of speaker specific encoding resulting in the audio that is not distinguished from one speaker to another. Attention Network cannot be trained until the encoder is stabilized. So, the training is performed in two phases. The first phase trains the autoencoder without applying an attention mechanism. In the second phase an attention network is added and is trained jointly with decoders keeping the encoder fixed and again the network is trained to predict the output.

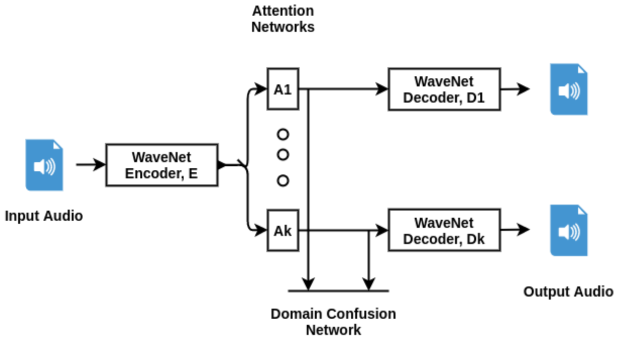

Figure 1: The architecture for the voice conversion using autoencoder along with domain confusion network and attention mechanism.

3.1 Encoder-Decoder

The Encoder-Decoder network is for encoding the variable length input sequence to the fixed length vector representation followed by decoder for decoding this fixed length vector into output sequence of variable length. Thus the model is able to learn the conditional distribution between two variable length sequences.

3.2 WaveNet Autoencoder

WaveNet is an autoregressive model which has the capability to predict the probability distribution of the next sample, given the previous samples and an input conditioning signal. It produces a whole sequence of samples by feeding a model with previously generated samples. Choice of softmax as probability distribution in WaveNet makes both training and synthesis tasks computationally tractable. Conditioning of WaveNet on acoustic features convey prosodic and verbal information. These variables are upsampled using the desired frequency and those are fed to the network of WaveNet via the conditioning network. WaveNet is made up of two modules, first is a convolution stack and second is post processing module. The convolution stack is composed of dilated convolution residual units. This unit performs multi scale feature extraction. Post-processing unit simply combines the output information from these residual blocks to infer the next sample.

In the architecture below Fig.2, the encoder will be fully convolutional hence it can be applied to any length of sequence. The network will have three blocks of ten residual layers each. Each residual layer will have ReLU nonlinearity, Non causal dilated convolution unit, a second ReLU, 1x1 convolution layer and finally residual summation of the activations before the first ReLU. There is a fixed width of 128 channels. After the temporal encoder an additional 1 x 1 layer is applied and output of this layer passed through another layer which performs average pooling. Polling is done by a kernel size of 50 ms (i.e. 800 samples). This polling downsample the encoding by a factor of 12.5. The encoding is upsampled temporally to the original audio rate using nearest neighbor. Interpolation is used to condition a WaveNet decoder. The conditioning signal is passed through a 1 × 1 layer that is different for each WaveNet layer.

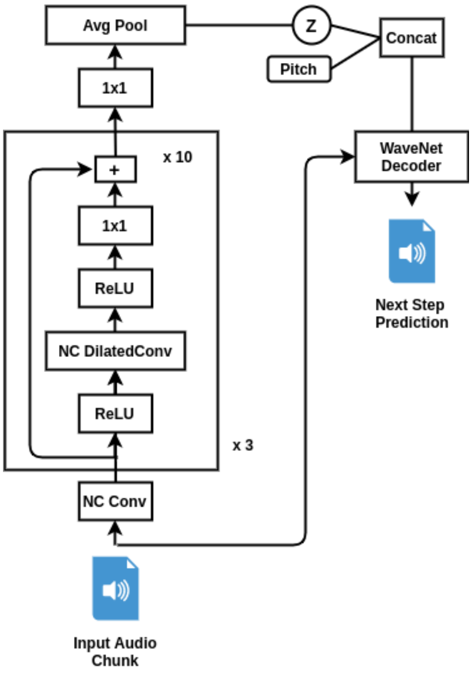

Figure 2: The architecture for WaveNet autoencoder.

3.3 Attention Mechanism

The ability to focus on a specific subset of inputs in a neural network can be added using attention. The main disadvantage of the sequence to sequence model is that it produces a fixed length context vector. An attention mechanism is used to overcome the problem of necessity to represent input sequence by fixed length vector representation in encoder-decoder architecture. Here, the purpose of this module is to modify the samples temporally in order to capture the patterns of the target speaker. Say the encoder E time-stretched version of the signal s^j. The signal is partitioned into segments of 0.3 to 0.5 second and stretched by 50 to 150 percent. Then speaker specific attention network A^j is trained to get back signal s^j from $T(s^j)$. Before training the attention component, the encoder must be stabilized. Hence, the training of the network is performed in two steps. The first phase trains the network without an attention mechanism. Once, the encoder is stabilized then the network is trained adding the attention component. The attention network is trained to minimize the loss given:

$$loss\Psi_j = \sum_{s^j}(A^j(E(T(s^j))) - E(s^j))^2 \tag{1}$$

3.4 Losses Used

Encoder needs to be stabilized before training the attention network. Hence training is performed in two steps so the loss is also used in two stage. Consider s^j as the input audio sample of j^{th} speaker. We have previously assumed E as encoder and D^j as a decoder of j^{th} speaker. Let C be the speaker classification which is made up of three 1D convolution layers with ELU nonlinearity [14] as an activation. There will be k output vector followed by softmax. During the first phase of training C reduces the cross entropy classification loss using :

$$\Omega = \sum_j \sum_{s^j} L(C(E(s^j)), j) \tag{2}$$

Hence, the encoder-decoder is trained with the loss:

$$-\lambda\Omega + \sum_j \sum_{s^j} L(D^j(E(s^j)), j) \tag{3}$$

Here $\lambda = 10^{-2}$ and $L(o,y)$ is the cross entropy loss applied. Here the decoder D^j is an autoregressive model and by the condition obtained on the output of Encoder E. Once the encoder is stabilized, the second phase of training starts where the network is trained jointly with the decoders. The loss is given as:

$$\delta \sum_j \Psi_j + \sum_j \sum_{s^j} L(D^j(A^j(E(s^j))), s^j) \tag{4}$$

During the synthesis phase we use the attenuated autoencoder in order to obtain a conversion using $D^j(A^j(E(s^j))$.

4. Experiment and Results

4.0.1 Data Collection and Preprocessing

Building a good model for voice conversion requires large datasets of high quality audio from targeted speakers. The model was first trained on 480 audio datasets from 17 male speakers followed by training on 1500 audio datasets of single male speakers collected from youtube. Data Preprocessing is done in three stages:

- **Data Parsing:** In this step, data from individual speakers is arranged in individual folder structure. Only the .wav format file was kept inside the folder. Unnecessary audio clips and data were dropped and all folders were placed inside a single folder. There was data of very small size and time duration (i.e. in range of microsecond). Those data were also removed during parsing.
- **Data Split:** Here we will separate data into train, test and validation sets. 10% of data were kept for the validation set and another 10% for the test set. The remaining 80% of the data were used for training the model.
- **Data Preprocessing:** Now we want .wav format data in .h5 format. Because the .h5 format is also known as Hierarchical Data Format 5. It allows us to store a huge amount of numerical data and easily work on the data using Numpy arrays. h5py python library was used for this conversion.

4.0.2 Training Setup

The training procedure involves the training of the WaveNet autoencoder without an attention network followed by the attention network. The code was written on the Pytorch framework. The training setup was done on the GeForce GTX 1050 GPU of Nvidia. The device consisted of Ubuntu 18 os and had 16 GB ram. The network was trained for 5000 epochs. It took 6.5 days to complete training. ADAM optimization algorithm is used and a learning rate of 0.0001 was used and Batch size of 4 was taken. Because of resource constraints, it was not possible to use larger batch sizes. Sampling rate of 16000 was taken as suggested by nyquist criterion. One second of the audio clip contains 44.1K samples. Model attained the training loss of 1.18 and test loss of 1.27.

4.0.3 Evaluation

Evaluation of audio signals only by Quantitative means gives lousy results. Qualitative Evaluation of the synthesized audio is done by taking the Mean Opinion Score. It is an average of scores given by subjects to represent the quality of the system or quality of experience. It is commonly used in the area of video and audio quality evaluation. The score has five levels of ratings from 1 to 5 for labels of bad, poor, fair, good and excellent. Generally, audio samples in a test set are rendered from the model by input text. Then the original output and rendered output are rated by eight human listeners with scores between 1 to 5. Score steps are taken at 0.5 and raters will independently rate the output. Finally the averaged output of ground truth and rendered sample is evaluated. Similarity of synthesized audio to the targeted person's voice can quantitatively be measured using correlation of synthesized signal with the ground truth.

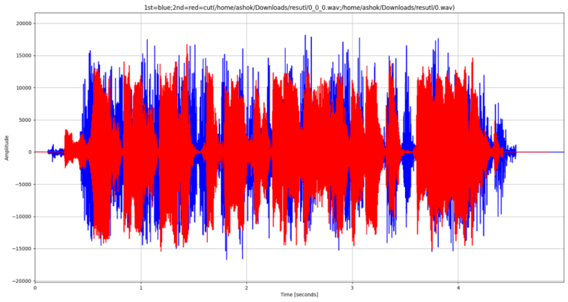

Figure 3: Amplitude vs Time in second plot, Comparison of recorded audio waveform with the synthesized audio waveform.

The plot of a recorded signal waveform on top of synthesized audio waveform can be plotted by synchronization of time using Syncstart (i.e. a tool developed by MIT for signal comparison). This provides the visual comparison of how aligned the two waveforms are. Both the waves utter the same words ”Bahul Jaati Bhasa Dharma Sanskriti Chan Bishala”. Following mel-spectrogram representation can be observed and visually analysed the patterns of amplitude in frequency vs time plot. Besides some noise in the plot both the spectrogram appears to be similar.

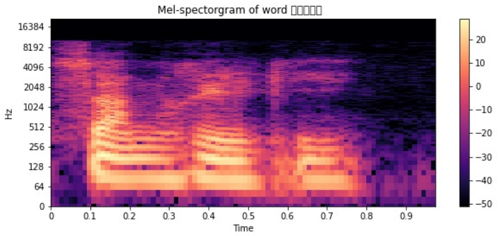

Figure 4: Mel-spectrogram plot of target word 'Samridhi'.

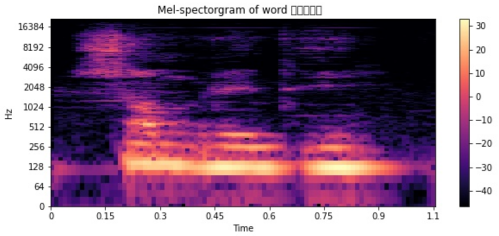

Figure 5: Mel-spectrogram plot of converted word 'Samridhi' by the model.

For more evaluation 25 audio samples were synthesized. The evaluation was performed after the noise removal of the sample using audacity. The input samples were from a different speaker voice than that of the targeted speaker. The MOS of the audio samples was found to be 2.78 as rated by 8 listeners.

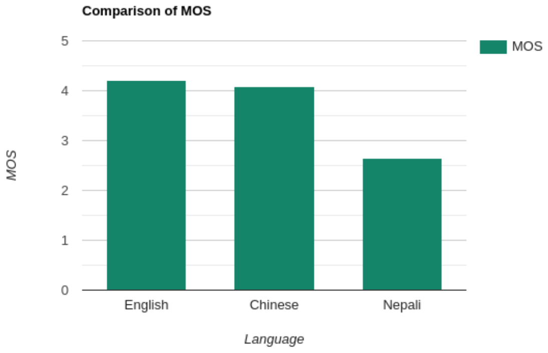

Figure 6: Comparing MOS of our model (i.e. for Synthesizing audio in Nepali) with the other language model trained on higher volume of data.

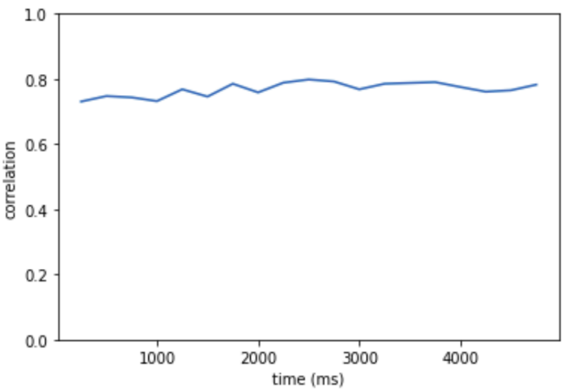

Figure 7: Correlation between recorded audio signal from targeted speaker and synthesized output audio sample from model taken at the frame of 200ms.

On training with 30 data, the input and reconstructed audio produced normalized cross-correlation was found to be 0.56 but after training on 480 data it was raised to 0.68. If the sample of data is increased and noise on data is reduced then the input and reconstructed audio produced shows better correlation. The correlation computed and averaged on 30 converted samples of audio signals with the ground truth i.e. voice of the targeted speaker was taken and found to be 0.78. The correlation was computed over a span of 200ms.

5. Conclusion

The work showed that it is possible to achieve voice conversion for Nepali Speech using the above architecture. The speech signal can be changed to latent space representation and can be decoded back to the original audio signal using decoder. The computational cost both for model training and for synthesizing a very small clip was very high. It is also found that on training with noisy data, the model won't stabilize and insufficient sample data won't produce satisfactory results. The quality of converted audio samples can further be improved by increasing the training set, increasing the number of epochs and decreasing the noise in the data. The architecture can be implemented as a vocoder on a text-to-speech system for synthesizing natural speech. This technology can also be implemented in music conversion and natural speech synthesis.

Acknowledgments

This work has been supported by the University Grants Commission, Nepal under a Faculty Research Grant (UGC Award No. FRG-76/77- Engg-1) for the research project ”Preparation of Nepali Speech Corpus: Step towards Efficient Nepali Speech Processing”.

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] Adam Polyak and Lior Wolf. Attention-based wavenet autoencoder for universal voice conversion. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6800–6804. IEEE, 2019.

[3] Jesse Engel, Cinjon Resnick, Adam Roberts, Sander Dieleman, Mohammad Norouzi, Douglas Eck, and Karen Simonyan. Neural audio synthesis of musical notes with wavenet autoencoders. In *International Conference on Machine Learning*, pages 1068–1077. PMLR, 2017.

[4] Alan W Black and Kevin A Lenzo. Flite: a small fast run-time synthesis engine. In *4th ISCA Tutorial and Research Workshop (ITRW) on Speech Synthesis*, 2001.

[5] Kaizhi Qian, Yang Zhang, Shiyu Chang, Mark Hasegawa-Johnson, and David Cox. Unsupervised speech decomposition via triple information bottleneck. In *International Conference on Machine Learning*, pages 7836–7846. PMLR, 2020.

[6] Oliver Watts, Cassia Valentini-Botinhao, Felipe Espic, and Simon King. Exemplar-based speech waveform generation. In *INTERSPEECH*, pages 2022–2026, 2018.

[7] Simon King. An introduction to statistical parametric speech synthesis. *Sadhana*, 36(5):837–852, 2011.

[8] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112, 2014.

[9] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

[10] Xie Chen, Wei-Qiang Zhang, Jia Liu, and Xiuguo Bao. An improved method for voice conversion based on gaussian mixture model. In *2010 International Conference on Computer Application and System Modeling (ICCASM 2010)*, volume 4, pages V4–404. IEEE, 2010.

[11] Akira Tamamori, Tomoki Hayashi, Kazuhiro Kobayashi, Kazuya Takeda, and Tomoki Toda. Speaker-dependent wavenet vocoder. In *Proc. Interspeech 2017*, pages 1118–1122, 2017.

[12] Heiga Zen, Yannis Agiomyrgiannakis, Niels Egberts, Fergus Henderson, and Przemysław Szczepaniak. Fast, compact, and high quality lstm-rnn based statistical parametric speech synthesizers for mobile devices. *arXiv preprint arXiv:1606.06061*, 2016.

[13] Xavi Gonzalvo, Siamak Tazari, Chun-an Chan, Markus Becker, Alexander Gutkin, and Hanna Silen. Recent advances in google real-time hmm-driven unit selection synthesizer. 2016.

[14] Djork-Arné Clevert, Thomas Unterthiner, and Sepp Hochreiter. Fast and accurate deep network learning by exponential linear units (elus). *arXiv preprint arXiv:1511.07289*, 2015.