UKRAINIAN CATHOLIC UNIVERSITY

MASTER THESIS

Parameterizing Human Speech Generation

Author:
Nazariy PEREPICHKA

Supervisor:
Diego SAEZ-TRUMPER

*A thesis submitted in fulfillment of the requirements
for the degree of Master of Science*

in the

Department of Computer Sciences
Faculty of Applied Sciences

УКРАЇНСЬКИЙ КАТОЛИЦЬКИЙ УНІВЕРСИТЕТ

APPLIED
SCIENCES
FACULTY

Lviv 2020

Declaration of Authorship

I, Nazariy PEREPICHKA, declare that this thesis titled, "Parameterizing Human Speech Generation" and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed: ________________________________

Date: ________________________________

UKRAINIAN CATHOLIC UNIVERSITY

Abstract

Faculty of Applied Sciences

Master of Science

Parameterizing Human Speech Generation

by Nazariy PEREPICHKA

In modern days synthesis of human images and videos is arguably one of the most popular topics in the Data Science community. The synthesis of human speech is less trendy but deeply bonded to the mentioned topic. Since the publication of WaveNet paper by Google researchers in 2016, the state-of-the-art approach transferred from parametric and concatenative systems to deep learning models.

Most of the work on the area focuses on improving the intelligibility and naturalness of the speech. However, almost every significant study also mentions ways to generate speech with the voices of different speakers. Usually, such an enhancement requires the model's re-training in case of generating audio with the voice of a speaker that was not present in the training set.

Additionally, studies focused on highly modular speech generation are rare. Therefore there is a room left for research on ways to add new parameters for other aspects of the speech, like sentiment, prosody, and melody.

In this work, we aimed to implement a competitive text-to-speech solution with the ability to specify the speaker without model re-training and explore possibilities for adding emotions to the generated speech.

Our approach generates good quality speech with the mean opinion score of 3,78 (out of 5) points and the ability to mimic speaker voice in real-time, which is a significant improvement over the baseline that merely obtains 2,08. On top of that, we researched sentiment representation possibilities. We built an emotion classifier that performs on the level of the current state of the art solutions by giving an accuracy of more than eighty percent.

Acknowledgements

I want to thank Diego Saez Trumper for invaluable pieces of advice and guidance during the development of this work.

Also, I want to thank Dmitry Spodarets from VITech for giving computational resources, which highly benefited the experimental results of this thesis.

Furthermore, my last thanks will go to my classmates Volodymyr Lut and Roman Riazantsev, for the fellowship and mutual support during our time spent in the Ukrainian Catholic University.

Contents

List of Figures

List of Tables

List of Abbreviations

TTS	**T**ext **T**o **S**peech
SOTA	**S**tate **O**f **T**he **A**rt
WAV	**WAV**e Audio File Format
STFT	**S**hort-**T**erm **F**ourier **T**ransform
NN	**N**eural **N**etwork
CNN	**C**onvolutional **N**eural **N**etwork
RNN	**R**ecurrent **N**eural **N**etwork
LSTM	**L**ong **S**hort-**T**erm **M**emory
GRU	**G**ated **R**ecurrent **U**nit
GEEL	**G**eneralized **E**nd-to-**E**nd **L**oss
MOS	**M**ean **O**pinion **S**core

To my dearest Yuliia

Chapter 1

Introduction

1.1 Motivation

The speech synthesis problem has a long research history. The desire to generate human speech from a written text is easy to understand, given that the potential areas of applications for such systems are enormous: generation of audio-books, voice acting of films, making computer systems socially accessible. Most of the possible applications require not only high quality of the speech but also the ability to specify other speech parameters, like tone and accent, speaker voice and emotion, melodic and rhythmic components.

In the last five years, the researchers have made significant progress in this field. The big breakthrough happened with the applying of deep learning techniques to the task (A. v. d. Oord et al., 2016). Every year, the solutions diminish the difference between program-generated and human speech samples(Arik et al., 2017, Wang et al., 2017, Shen et al., 2018) and optimize training time (A. v. d. Oord et al., 2017, Kalchbrenner et al., 2018).

Though the quality of generated speech increased, the replication of speech parameters is still a pretty young research area(Jia et al., 2019, Y. Wan et al., 2018). The ability to generate natural-sounding, emotional speech can become the next big breakthrough in the speech synthesis domain.

1.2 Goals and Contributions

The main goal of this work is to implement the end-to-end Text To Speech (TTS) system, which generates reasonable speech output with limited computational resources. Also, we studied and tested the possibilities for speech parameterization by criteria, like voice and sentiment.

Developing such a TTS model requires a setup of a framework for the evaluation of results and profound research in the field of speech synthesis.

The main contributions of this thesis are:

- Implementation of a TTS system with limited computational resources.
- Creating a framework for human evaluation of the experiments.
- Building of representation emotions representation from speech audio.

1.3 Thesis structure

In Chapter 2, we review the theoretical background needed for the research. It contains two parts: Signal Processing and Deep Learning theory. Signal processing is required to understand audio representations and preprocessing techniques for them. The Deep Learning section contains an explanation of the theory and algorithms mentioned in the thesis.

Next, in Chapter 3, we provide an overview of the related work in the field.

Chapter 4 explains our proposed method for the solution and go over architectural decisions made during the research.

In Chapter 5, we describe the experiments, evaluation framework, and results of the work.

Finally, in Chapter 6, we summarize the thesis: provide key results and discuss possible directions of further work.

Chapter 2

Background

In this chapter, we describe the background knowledge needed to develop this work. The chapter is divided into two sections. The first one provides a background for the understanding of the most relevant components of audio generation and processing. The second section overviews deep learning algorithms, as nowadays, they are the primary approach for speech synthesis.

2.1 Signal Processing

The most basic TTS task definition is a mapping of written words to the audio file. It requires experience in working with sounds in digital formats. The domain of expertise, which covers manipulations with sound representation, is called signal processing.

The need to process (store, compress, modify) the sound urged with the appearance of the first "ear-oriented" electronic devices(radio, telephone, phonograph) (Spanias, Painter, and Venkatraman, 2006). Therefore, the field of audio signal processing has a history of more than one hundred years, and in the following section, we present core definitions and algorithms related to the research.

2.1.1 Audio formats

The physics definition of sound is a vibration that propagates in a waveform through transmission matter as gas, liquid, or solid.

Two main characteristics of a sound are amplitude and frequency. Amplitude is the size of the vibration, and it defines the loudness of the sound. Frequency is the speed of the vibration, and it determines the pitch of the sound.

The most basic application of the audio signal processing field is storing of sounds. Two main techniques for storing a sound is through analog and digital signals. Analog signals are continuous, and usually, they present sound waves in the air by corresponding electrical voltage. Digital signals are discrete and approximate the sound by a sequence of binary numbers.

Most of the datasets suitable for developing of a TTS system contain audio in Waveform Audio File Format(WAV). WAV files do not compress the audio and represent the sound directly by capturing the raw value of sound waves in a particular period.

WAV format has two main characteristics:

- sampling rate - the number of samples recorded per second; commonly measured in Hz;
- bitrate - number of bits needed to store one second of audio;

2.1.2 Fourier transform

Fourier transform is the core algorithm for audio signal processing, based on the Euler's identity formula(2.1), which defines the fundamental relation between the trigonometric functions and complex exponential function (Rahman, 2011).

$$e^{j\omega} = cos(\theta) + jsin(\theta) \tag{2.1}$$

Fourier transform allows us to decompose a function of time into its constituent frequencies. For a continuous function, the formula is defined as in (2.2).

$$X(\omega) = \int_{-\infty}^{\infty} x(t)e^{-jwt}\,dt,\ \omega \in (-\infty, \infty) \tag{2.2}$$

For the processing of digital signals more useful is the discrete variation of the formula (2.3).

$$X(\omega_k) = \sum_{n=0}^{N-1} x(t_n)e^{-jw_k t_n},\ k = 0, 1, 2, .., N-1 \tag{2.3}$$

The result of the Fourier transform is often referred to as a spectrum.

Another Fourier transform modification used for audio processing is the short-term Fourier transform(STFT). STFT divides longer time signals into shorter components and computes Fourier transform on each of them. This approach allows to tractate change of a spectrum as a function of time (Sejdića, Djurović, and Jiang, 2009).

2.1.3 Spectrogram

The result of the Fourier transform can be visualized with a spectrogram. A spectrogram is a graph of frequencies through time. The intensity of frequencies(amplitudes) is usually defined through the corresponding color scale.

2.1.4 MEL-spectrogram

Spectrograms give the ability to understand the sound in a more intuitive way. The main problem with this approach is that the human ear is not a perfect audio receiver, and we perceive sounds in the range from 20Hz to 20kHz(Olson, 1967). The human perception of the sound has the associative nature: we hear different sound frequencies with a specific step as the same. The whole classical music theory is built on this fact.

To incorporate the subjectivity of human perception, the group of Harvard researchers introduced the scale (Stevens et al., 1937), which maps Hz to mels. Mel is a defined unit of frequency, which represents frequency in the human hearing range. There is no unified formula for the mapping, but the most commonly used one is (O'Shaughnessy, 1987).

$$m = 2595 \log_{10}(1 + f/700) \tag{2.4}$$

After converting frequency values to this psycho-acoustic scale, we can visualize signals using spectrograms. The result is called Mel-spectrogram.

The Mel scale emphasizes frequencies, which are perceived by a human. Figure 2.1 shows the difference between regular and Mel spectrograms.

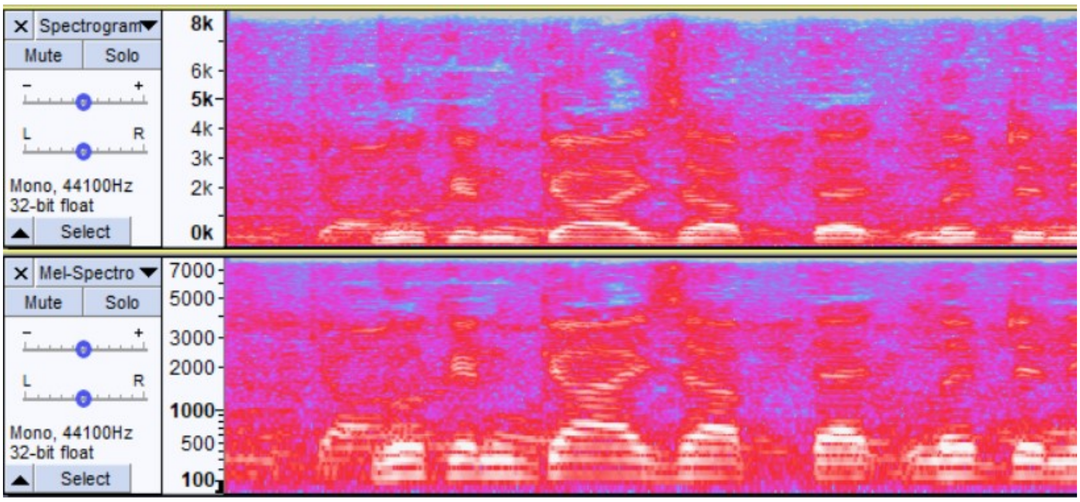


FIGURE 2.1: Visualization of audio spectrograms in Linear and Mel scales accordingly

2.2 Deep Learning

Current SOTA approaches for TTS problems are based on Deep Learning techniques. The proposed solution explicitly relies on the Neural Network theory.

In this section, I provide the theoretical background needed to understand the methods mentioned in the thesis.

2.2.1 Neural Networks

A neural network(NN) is a computing system inspired by the structure and computational behavior of the human brain. NNs provide a way for solving complex nonlinear tasks, which cannot be solved with classical algorithms.

NNs are composed of layers, which are composed of nodes called neurons. Each node has an activation function and weight coefficient, which reflects the importance of the node output in the layer. Based on the output of the NN, we calculate the function, which reflects the quality of the output, - loss function - and based on the chain rule, we modify the weights to optimize the loss function.

2.2.2 Convolutional Neural Networks

Convolutional Neural Network(CNN) is a neural network, which uses convolution operation (Goodfellow, Bengio, and Courville, 2016).

CNNs are dominant in the field of computer vision. They recognize visual patterns in the images by applying convolutional operation and extract more complex patterns with every following layer (Lecun et al., 1998).

2.2.3 Recurrent Neural Network

Recurrent Neural Networks(RNN) are a subset of artificial neural networks. They are mostly used for sequence data. By using an internal state, models process sequences of inputs. RNNs are crucial for the TTS systems, as human speech is sequential by its nature.

RNNs have different types of architecture. We describe only two of them, which are the most important for our research.

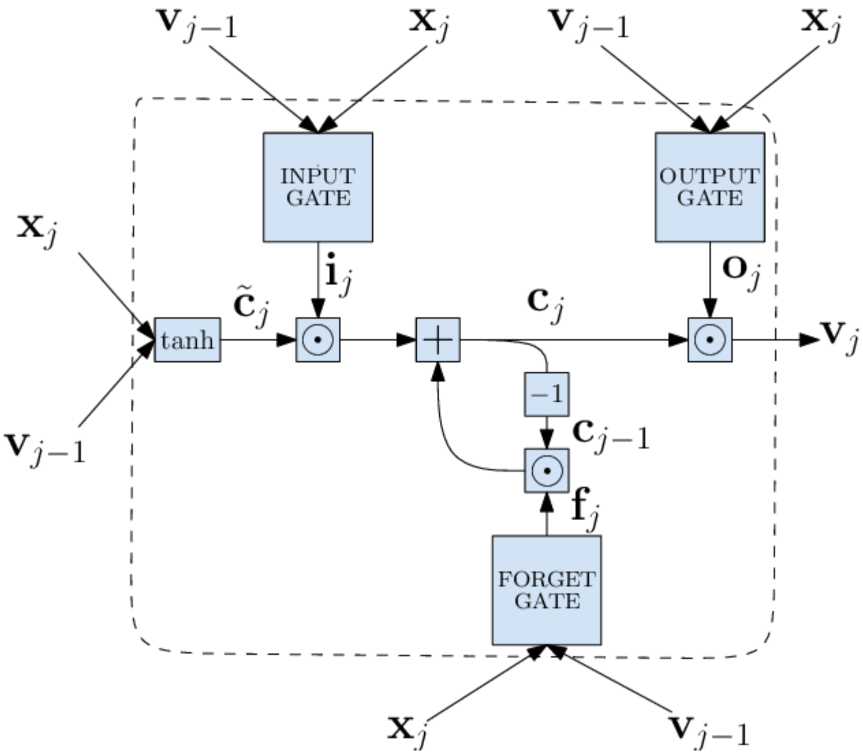


FIGURE 2.2: Visualization of LSTM cell (Bolaños et al., 2017)

Long Short-Term Memory

Long Short-Term Memory(LSTM) is an RNN architecture introduced in 1997 (Hochreiter and Schmidhuber, 1997).

LSTM captures patterns in the long sequences by storing intermediate results in the memory cells; therefore, they are well-suited for time series data with more extended periods between important events. LSTM also solves the problem of vanishing and exploding gradients.

The visualization of LSTM cell is presented in 2.2 figure. The mathematical definitions can be found in 2.5.

$$\begin{aligned}
f_t &= \sigma_g(W_f x_t + U_f h_{t-1} + b_f) \\
i_t &= \sigma_g(W_i x_t + U_i h_{t-1} + b_i) \\
o_t &= \sigma_g(W_o x_t + U_o h_{t-1} + b_o) \\
c_t &= f_t \circ c_{t-1} + i_t \circ \sigma_c(W_c x_t + U_c h_{t-1} + b_c) \\
h_t &= o_t \circ \sigma_h(c_t)
\end{aligned} \tag{2.5}$$

Since the introduction of the architecture, it achieved excellent results for the tasks related to modeling text and audio.

Gated Recurrent Unit

Gated Recurrent Unit (GRU) was introduced in 2014 (Cho et al., 2014) to simplify the LSTM structure. GRUs perform similar to LSTM cells but are more computationally efficient due to less learnable parameters.

The 2.3 shows the internal structure of the GRU cell. The mathematical definition presented by formulas in 2.6.

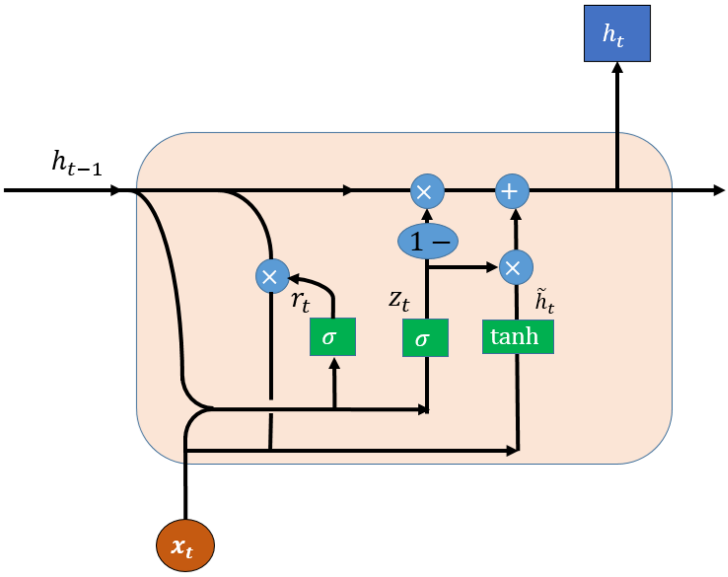


FIGURE 2.3: Visualization of GRU cell (Huang et al., 2019)

$$\begin{aligned} z_t &= \sigma_g(W_z x_t + U_z h_{t-1} + b_z) \\ r_t &= \sigma_g(W_r x_t + U_r h_{t-1} + b_r) \\ h_t &= z_t \circ h_{t-1} + (1 - z_t) \circ \phi_h(W_h x_t + U_h(r_t \circ h_{t-1}) + b_h) \end{aligned} \tag{2.6}$$

Chapter 3

Related Work

In this chapter, we describe the transition of TTS systems from more algorithmic approaches to current solutions. We focus on the Deep Learning systems, as they are more relevant to our proposed method.

3.1 Concatenative systems

Concatenative systems are the implementation of the intuitive idea to compose the speech audio out of small pre-recorded samples. This system builds output by concatenating recording units(words, phonemes).

Such an approach satisfies the intelligibility requirement, but it has multiple drawbacks: large, hard to collect unit database; unnatural "robotic" sound; hard-coded rule-based programming (Zhang, 2004).

3.2 Parametric systems

Parametric systems are a statistical approach for speech generation. Such systems synthesize speech based on acoustic and linguistic features. Such models use Hidden Markov Models, and speech waves are generated based on the maximum likelihood criterion (Tokuda et al., 2013).

The general pipeline for developing parametric TTS can be found in Figure 3.1.

Parametric TTS requires feature engineering by hand, and it is the main drawback of approach. Hypothetically, with proper features selection, such systems should work on the same level as deep learning models, but practically such systems perform much poorly.

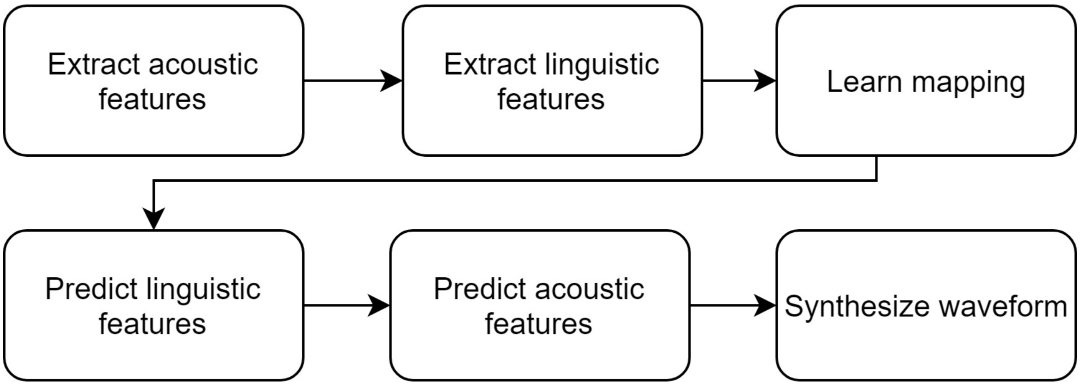

Extract acoustic features
Extract linguistic features
Learn mapping
Predict linguistic features
Predict acoustic features
Synthesize waveform

FIGURE 3.1: General schema of parametric TTS systems

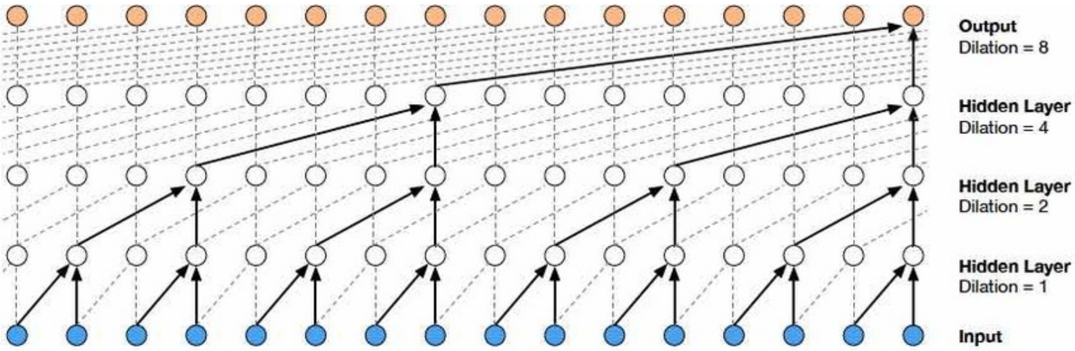


FIGURE 3.2: Visual representation of WaveNet convolutional layers (A. v. d. Oord et al., 2016)

3.3 Griffin-Lim Algorithm

The Griffin-Lim Algorithm was introduced in 1984 (Griffin and Lim, 1983). GLA estimates the signal from its modified short-time Fourier transform.

The GLA is iterative, and it minimizes the mean squared error between estimated and modified Short Time Fourier Transforms.

3.4 Wavenet

The big breakthrough in speech synthesis happened with the publication from Google Deepmind in 2016 (A. v. d. Oord et al., 2016). The researchers presented a new architecture called WaveNet, which operates directly on the raw audio waveform and functions as a vocoder(model for audio generation). The joint probability of a waveform is factorized as a product of conditional probabilities, as follows:

$$p(x) = \prod_{t=1}^{T} p(x_t|x_1, ..., x_{t-1}) \tag{3.1}$$

The authors modeled the conditional probability distribution with a stack of convolutional layers. On the output layer, they receive conditional probability distribution for x_t given by softmax function.

For modeling, authors used gated activation units of the following form inspired by PixelCNN (Aaron van den Oord, 2016) architecture.

In the paper, authors mention the ability to parameterize output audio by incorporating additional parameters h in the joint probability, as follows:

$$p(x|h) = \prod_{t=1}^{T} p(x_t|x_1, ..., x_{t-1}, h) \tag{3.2}$$

For their research, they parameterized the narrator - by training on the dataset with multiple speakers - and text - by passing linguistic features of the text.

The TTS solution significantly outperformed all of the previous benchmarks.

3.5 Tacotron 2

In 2018 Google researchers presented the paper (Shen et al., 2018), which describes Tacotron 2 architecture for speech synthesis.

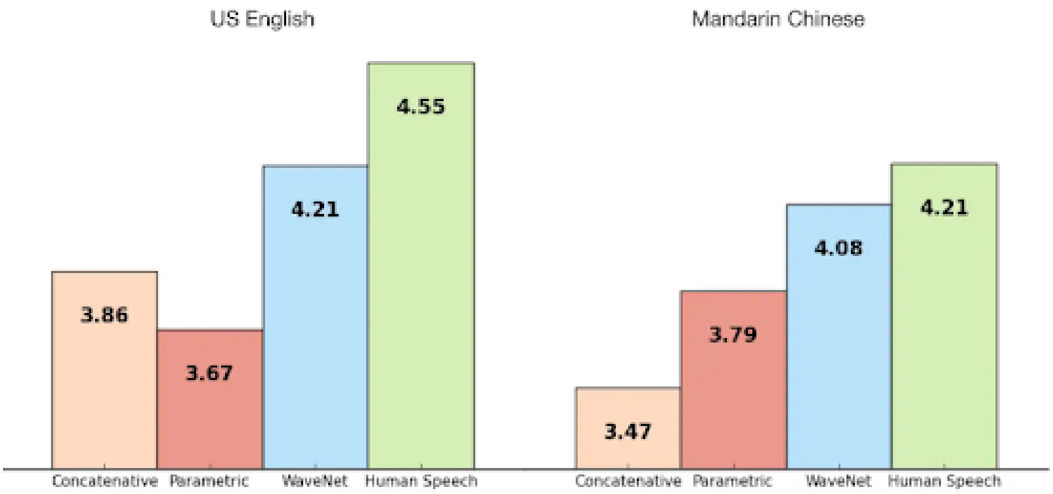


FIGURE 3.3: Histogram of MOS scores for WaveNet and other TTS approaches (A. v. d. Oord et al., 2016)

The model projects text to MEL-scale spectrograms and then uses a modified Wavenet vocoder to audio output itself. This architecture combines WaveNet and Tacotron 1 models.

Tacotron 2 achieved a MOS score of 4.53, in comparison to 4.58 by the professionally recorded human speech.

3.6 Generalized End-To-End Loss For Speaker Classification

In 2019 Google researchers published a paper (L. Wan et al., 2018), in which they introduced new loss function(GEEL) for more efficient training of speaker recognition models.

For k speakers with M utterances, the model finds k centroids, which represent the voice in the best way. Centroids are defined through the following formula:

$$c_k = \sum_{m=1}^{M} e_{km}/M \tag{3.3}$$

Using LSTM-based architecture, they build speaker embedding, which is an L2-normalized output of the model.

On each step, the similarity matrix $S_{ji,k}$ is built. It contains cosine similarities values between each embedding vector e_{ji} and each centroid c_k.

$$S_{ji,k} = \omega * \cos e_{ji}, c_k + b \tag{3.4}$$

The final classification layer is softmax. The authors define two losses: standard and contrast. Standard loss has the following form:

$$L(e_{ji}) = -S_{ji,j} + \log \sum_{k=1}^{N} exp(S_{ji,k}) \tag{3.5}$$

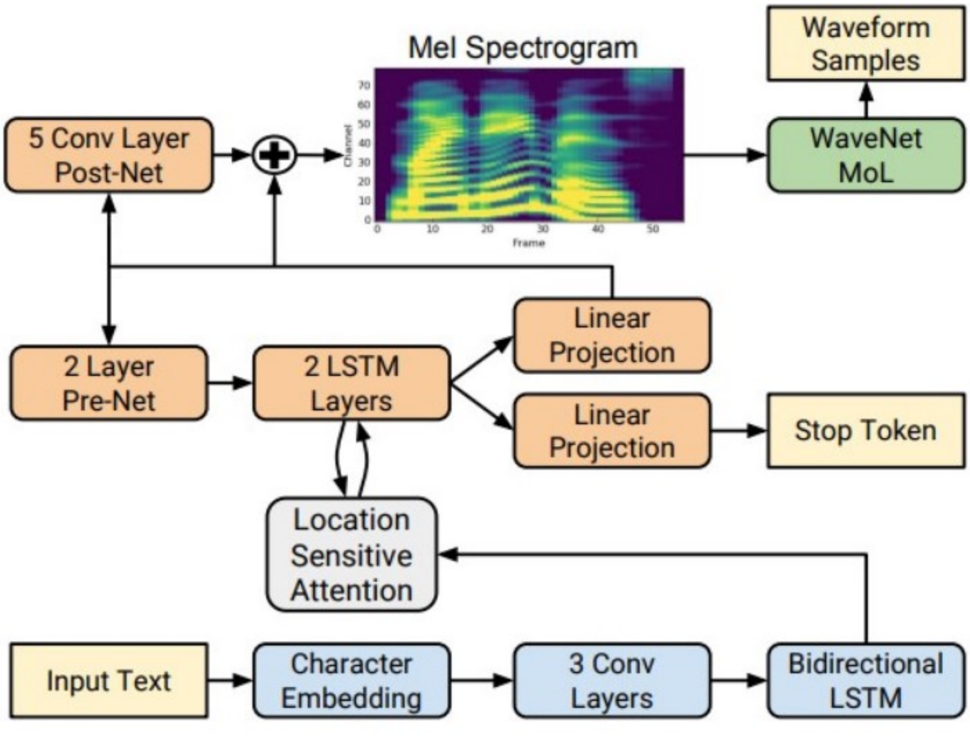


FIGURE 3.4: Tacotron 2 architecture (Shen et al., 2018)

For the positive pairs and most aggressive negative pairs authors introduce contrast loss:

$$L(e_{ji}) = 1 - \sigma(S_{ji,j}) + \max \sigma(S_{ji,k}) \tag{3.6}$$

The final loss is a sum of previously defined:

$$L_G(x; w) = \sum_{j,i} L(e_{ji}) \tag{3.7}$$

In their paper, researchers showed theoretical and experimental results, which proved the superiority of the GEEL loss over its predecessors.

3.7 Voice Cloning

At the beginning of 2019, Google published the paper (Jia et al., 2019), which brought together ideas from Wavenet, Tacotron 2, and Generalized End-To-End Loss For Speaker Classification sections.

The paper described the TTS system, which allows performing real-time voice cloning.

The model consists of three parts: speaker encoder trained with GEEL loss, synthesizer with Tacotron 2 architecture, and Wavenet Vocoder. Each module of the final network trains separately and is replaceable by other architectures.

The authors did not share specifications for the model's architecture but the presented results, which showed the ability to generate high-quality human speech with the different voices and generalize to voices, which were not present in the training set.

3.8 WaveRNN

In 2018 researchers of DeepMind presented their approach for vocoding (Kalchbrenner et al., 2018). The paper introduces WaveRNN architecture, which aims to be more computationally efficient than the current SOTA solution - WaveNet.

Instead of a convolutional approach to audio sequence computation, WaveRNN uses a modification of GRU cells. Experiments showed that WaveRNN performs worse than WaveNet, but with the increasing of hidden units, the difference becomes insignificant, and computational benefits are valid.

The WaveRNN vocoder is suitable for training on personal computers and rather modest hardware, unlike the WaveNet one.

Chapter 4

Proposed method

In this work, we aimed to implement the working TTS system with an ability of parameterization by different factors.

We planned to build the model similar to the one described in the Voice Cloning section. However, there are several challenges to achieve that goal:

- The described paper (Jia et al., 2019) did not provide concrete details of implementation (preprocessing parameters, projection size of the embedding, hyperparameters),
- Training of the WaveNet vocoder was not feasible with the computational resources in our possession.

The first problem was solved by trial and error, and by experimenting with different approaches. We built our preprocessing engine and chose hyperparameters during the initial phases of the research. For the second problem, the only suitable solution was to replace original vocoder architecture with more suitable and computationally efficient.

On figure 4.1, the general architecture of the proposed solution is displayed. In the following subsections, we provide details on each part of the system.

4.1 Encoders

For encoding of voice, we implemented the model from Voice Cloning(Jia et al., 2019) with Generalized End-To-End Loss For Speaker Classification(L. Wan et al.,

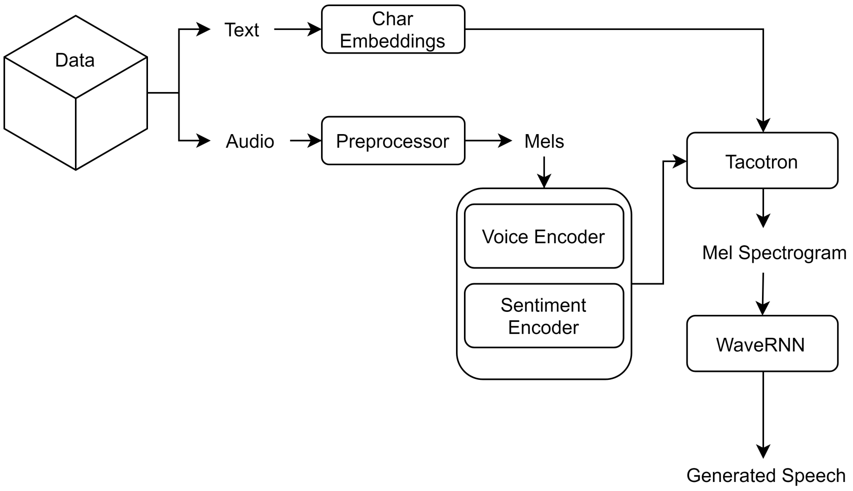


FIGURE 4.1: Generalized training flow for the proposed architecture

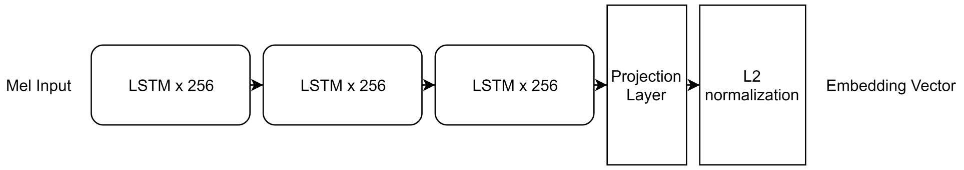


FIGURE 4.2: Encoder architecture used for general model

2018). We used two LSTM layers with 256 hidden units per each with the projection layer. We tried projection layers with different sizes, but in the end, the projection size of 256 worked the best.

For the learning of embeddings, we needed a dataset with a high-variety of speakers and emotions. There are multiple datasets for the speaker recognition task, which suit the objective. For the sentiment classification, there are fewer datasets available, and the majority of them are expensive.

For the training of the Mel synthesizer, there are more requirements for the dataset. Data should contain annotated text, speaker, and emotion. We did not find a database, which suits the description.

The initial idea for tackling of this problem was to classify emotions in the available voice+text datasets. The open-source implementations of sentiment classifiers did not have an accuracy needed; therefore, we used our implementation of the encoder for the classification problem but faced the issue of a high imbalance in the datasets suited for synthesizer training.

As a result, we parameterized the speech generation only by voice and built a sentiment classifier.

4.2 Mel synthesis

For Mel Synthesizer, we used Tacotron 2 architecture (Shen et al., 2018) parameterized by voice embedding.

We decided to use Mel preprocessing as it shows better results for speech generation then modeling on the raw waveform. During the research, we experimented with other ways to represent audio(Harmonic Model, Harmonic Product Spectrum.) but failed due to the complexity of such techniques and the absence of starting point for such an approach. Though we believe that such representation of audio can outperform models with Mel representation, we decided to use a time-proven solution.

4.3 Vocoder

For the final vocoder, we used WaveRNN (Kalchbrenner et al., 2018) architecture. Other two possible options were Griffin-Lim algorithm (Griffin and Lim, 1983) and Wavenet model(A. v. d. Oord et al., 2016).

Griffin-Lim algorithm was not included in the final model due to much worse performance than WaveRNN and WavNet, but due to its simplicity and computational efficiency, it was used for quick calculation of intermediate results of Mel synthesizer and debugging purposes.

WaveNet vocoder is the current SOTA solution, but the architecture size and amount of computation power needed to re-train it was not feasible for us.

Chapter 5

Experiments And Evaluation

5.1 Datasets Description

The building of a TTS system is a complex task, which requires many data. The table below provides the technical description of datasets used for model training.

Dataset Name	Size	Samples	File Format	Model Trained
LJ Speech Dataset	24 hours	13,100	.wav	Baseline, WaveRNN
VoxCeleb1	2000 hours	1 million	.wav	Voice Encoder
VoxCeleb2	2000 hours	1 million	.wav	Voice Encoder
LibriSpeech	1000 hours	500,000	.flac	Mel Synthesizer, Vocoder
Ravdess Speech	2 hours	1440	.wav	Sentiment Encoder

TABLE 5.1: Datasets description

Those four datasets were used for the training of the final models.

- **LJ Speech** (Ito, 2017)

  The database contains audio clips of a single speaker with transcriptions of the spoken time. The dataset is relatively light-weight and was mostly used for the testing and debugging purposes

- **VoxCeleb1** (Nagrani, Chung, and Zisserman, 2017)

  The dataset contains audio clips of more than 7,000 speakers without text transcriptions. It was used for the training of voice embeddings

- **VoxCeleb2** (Nagrani, Chung, and Zisserman, 2018)

  The database contains audio clips of more than 6,000 speakers without text transcriptions without overlap with VoxCeleb1. It was used for the training of voice embeddings.

- **LibriSpeech** (Panayotov et al., 2015)

  Dataset formed from audiobooks from the LibriVox project [1] with more than 1,000 speakers and annotation of spoken text. It was used for the training of voice embeddings and Mel synthesizer.

- **Ravdess** (Livingstone and Russo, 2018)

  Dataset of emotional speech with video and audio parts. It contains recordings of two sentences said by twelve actors with eight different emotions.

[1] https://librivox.org/

5.2 TTS model

5.2.1 Baseline model

For the starting point, we took an outsource implementation of Tacotron (Shen et al., 2018) with the Griffin-Lim Algorithm vocoder(Griffin and Lim, 1983. [2]

We re-trained the model on LJSpeech Dataset. The output of the model was not close to the SOTA results. However, computational powers utilized by Google researchers for such models are uncomparable to the one that we had, so we decided that the current model will suit as a sufficient benchmark.

5.2.2 TTS experiments

The voice encoder was trained on VoxCeleb1, VoxCeleb2, and LibriSpeech datasets.

During the implementation, we have experimented with different sizes of projection layers. Based on the early results of the training, we chose to construct an embedding of 256 units. The original model was trained on the much bigger dataset, constructed from a combination of open-source sources and exclusive to the authors' samples. Despite the difference in the training data, our model showed the ability to extract voice features. During the exploration of intermediate results, we also discovered that the model captured slight intonation, accent, and prosody features of the speaker.

The initial point for the Tacotron model was the baseline implementation, which we parameterized by voice embedding and trained the model on the LibriSpeech dataset. For quick iteration and intermediate results generation, we used Griffin-Lim Algorithm algorithm from the baseline model.

For vocoder, we used pre-trained open-source implementation of WaveRNN architecture [3], and trained it further on the LibriSpeech dataset.

5.2.3 Evaluation approach

The main two goals of the TTS system are intelligibility (capability of being understood) and naturalness (ability to mimic human speech). The human perception of the output defines both of the evaluation parameters. Therefore the evaluation of TTS systems requires subjective techniques for quality measurements.

The most popular metric is the mean opinion score (MOS) - average grade given to the audio sample by respondents. MOS is arithmetic mean over user-given rating and depends on two parameters: quantity of respondents and grading scale:

$$MOS = \frac{\sum_{n=1}^{N} R_n}{N} \tag{5.1}$$

5.2.4 Evaluation Framework

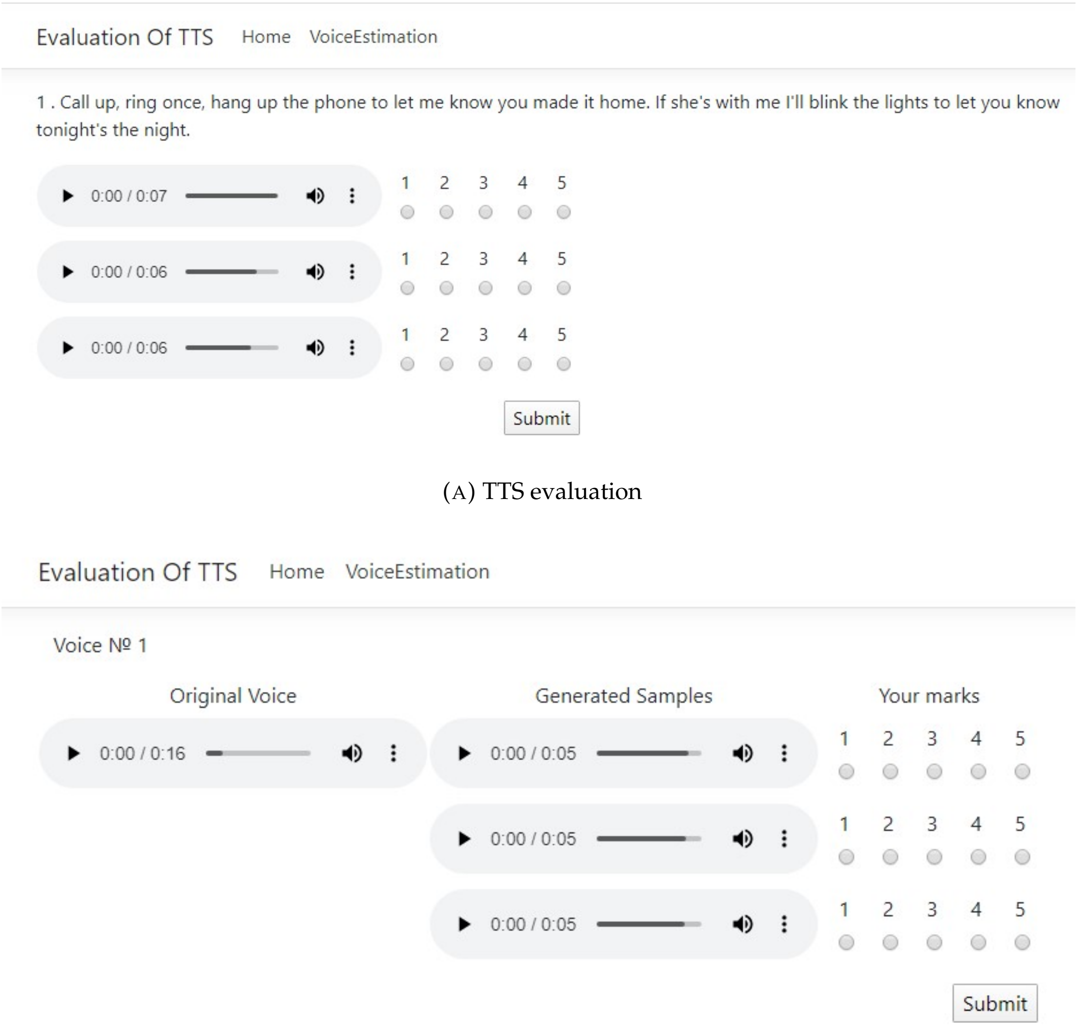
For the evaluation of generated speech, we developed a web tool [4] with the interface for evaluation of the TTS system and quality of voice copying.

For the TTS evaluation, we selected fifty sentences(see Evaluation sentences Appendix) and synthesized audio, using baseline and final models. For the final model,

[2] https://github.com/keithito/tacotron

[3] https://github.com/fatchord/WaveRNN

[4] https://tts-estimation.azurewebsites.net/



1 . Call up, ring once, hang up the phone to let me know you made it home. If she's with me I'll blink the lights to let you know tonight's the night.

(A) TTS evaluation

(B) Voice Evaluation

FIGURE 5.1: Screenshots of a user interface for model evaluation

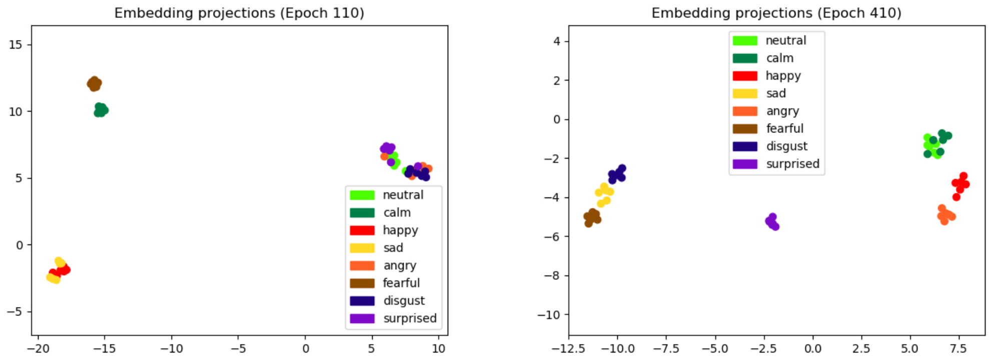


FIGURE 5.2: Visualization of learned sentiment embeddings

we used two voices: male and female, which were present in the training set and collected 5000 scores in total from 200 people.

Model	Mean Opinion Score
Baseline	2,08
Parameterized with female voice	3,78
Parameterized with male voice	3,75

TABLE 5.2: Evaluation results of TTS

For the quality of voice copying, we synthesized audio on one specific sentence and instructed respondents to take into account the only similarity between the original voice and generated one. The embedding vectors were formed for eight speakers from samples of different lengths: 3, 5, and 8 seconds. In total, we received 400 scores for the generated samples from 50 unique evaluators.

Sample Length	Mean Opinion Score
3 seconds	2,65
5 seconds	2,70
8 seconds	2,77

TABLE 5.3: Evaluation results for voice copying

5.3 Sentiment Classification

For the training, we used the Ravdess dataset and used the same architecture, as for voice encoder. As a benchmark, we took the open-source implementation of emotion recognizer, which had reported and accuracy of 76,6 percent on the test set. [5]

Due to the small size of the training set, the number of steps needed for training of the classifier was much smaller than for previously mentioned models. The model faced overfitting problems, which we resolved by adding a recurrent dropout to the initial architecture.

[5] https://github.com/maelfabien/Multimodal-Emotion-Recognition

5.3.1 Sentiment Classifier Evaluation

We trained the model on the Ravdess dataset. During the training, we faced a high variance problem, which was resolved by adding recurrent dropouts to the LSTM layers. We outperformed the baseline model by achieving an accuracy of 81,3 percent with an F1 macro score of 80,56.

The detailed evaluation of the classifier is presented in the table below:

class	precision	recall	f1-score
neutral	0,86	1,00	0,92
calm	1,00	1,00	1,00
happy	0,80	0,33	0,47
sad	0,60	0,75	0,67
angry	1,00	1,00	1,00
fearful	1,00	1,00	1,00
disgust	0,36	0,42	0,38
surprised	1,00	1,00	1,00
weighted average	0,83	0,81	0,81

TABLE 5.4: Sentiment classification results

The model outperformed the baseline result, but it is unlikely that it generalizes to the real-world scenarios, due to the size and variety of dataset.

Chapter 6

Conclusions

In our work, we researched modern approaches for high-quality human speech generation. The main achievements of this work are the following:

- We built the working TTS system with the ability to parameterize speech generation by voice in real-time.
- We developed an evaluation framework for the subjective estimation of systems for speech generation.
- We improved the quality of synthesized speech in comparison to the baseline solution by 1,6 points (out of five) in the opinion score.
- We did not achieve the same quality of voice parameterization as in Jia et al., 2019, but we observed the same tendencies. Worse quality can be explained by a much smaller dataset, and learning time.
- We showed that the method for the embedding of voice could be applied to the embedding of sentiment.
- We build the emotion classifier, which outperformed the baseline model and introduced the new method for emotion recognition of audio.Current classification results are close to SOTA solutions (Yoon, Byun, and Jung, 2018, Etienne et al., 2018) built on bigger datasets.

6.1 Future work

The main point of the future work is constructing the dataset suitable for training of the TTS system with the ability of voice parameterization.

Another direction of future work would be to focus on emotion recognition from audio, as the current result is promising, and we did not find mentions of the recurrent encoder with the GEEL loss function applied to the classification task.

The last thing would be to continue experiments with other audio representations and compare other acoustic models to the Mel-based preprocessing.

Appendix A

Evaluation sentences

1. Where is the Money, Lebowski? Where is the Money?!
2. Oak is strong and also gives shade.
3. Cats and dogs each hate the other.
4. The pipe began to rust while new.
5. Open the crate, but don't break the glass.
6. Add the sum to the product of these three.
7. Thieves who rob friends deserve jail.
8. The ripe taste of cheese improves with age.
9. Act on these orders with great speed.
10. The hog crawled under the high fence.
11. Move the vat over the hot fire.
12. But now you come to me, and you say: 'Don Corleone, give me Justice.' But you don't ask with respect; you don't offer friendship.
13. Get out of here, robber! I said get out of here!
14. London is the capital of the great Britain.
15. My mother thanks you. My father thanks you. My sister thanks you. And I thank you.
16. I am the lizard king. I can do anything. We came down. The rivers and highways. We came down from. Forests and falls
17. This kid's going to be the best kid in the world. This kid's going to be somebody better than anybody I ever knew.
18. I want to break free. I want to break free from your lies. You're so self satisfied I don't need you.
19. What are they doing in the Hyacinth House? To please the lions this day.
20. Yesterday all my troubles seemed so far away. Now it looks as though they're here to stay.
21. The time to hesitate is through. No time to wallow in the mire.

22. Hey, guys. It's John and welcome to another video. Let's jump right into it.

23. That it's a thriller, thriller night. Cause I can thrill you more than any ghost would ever dare try.

24. I've just seen a face I can't forget the time or place where we just met. Had it been another day I might have looked the other way.

25. You need cooling. Baby I'm not fooling. I'm gonna send ya back to schooling.

26. Scaramouche, Scaramouche, will you do the Fandango?

27. So you think you can stone me and spit in my eye? So you think you can love me and leave me to die?

28. I know a girl called Elsa. She's into Alka Seltzer. She sniffs it through a cane on a supersonic train.

29. As I walk through the valley of the shadow of death I take a look at my life and realize there's not much left.

30. Call up, ring once, hang up the phone to let me know you made it home. If she's with me I'll blink the lights to let you know tonight's the night.

31. Sun lights up the daytime, moon lights up the night. I light up when you call my name, and you know I'm gonna treat you right.

32. Georgia, Georgia. The whole day through. Just an old sweet song keeps Georgia on my mind.

33. Every evening, evening happens all of a sudden.

34. For though they may be parted, there is still a chance that they will see. There will be an answer, let it be.

35. Summertime and the livin' is easy. Fish are jumpin' and the cotton is high.

36. I have a dream that one day on the red hills of Georgia, the sons of former slaves and the sons of former slave owners will be able to sit down together at the table of brotherhood.

37. Happiness is when what you think, what you say, and what you do are in harmony.

38. Nor, finally, are these remarks intended to examine the proper degree of privacy which the press should allow to any President and his family.

39. And I'm never gonna dance again - guilty feet have got no rhythm. Though it's easy to pretend - I know you're not a fool.

40. Up ahead in the distance, I saw a shimmering light. My head grew heavy and my sight grew dim. I had to stop for the night.

41. Take me to the place where you go, where nobody knows, if it's night or day. Please don't put your life in the hands, of a rock and roll band, who'll throw it all away.

42. In the pines, in the pines where the sun don't ever shine - I would shiver the whole night through.

43. Honey, the stars keep on calling my name. But don't worry, I've told you again and again. When I'm down, you're always the first one to know. Skipping town, I'll take you wherever I go.

44. Early in the morning. Just trying to let the sun in and open up my eyes.

45. Let her go, let her go, God bless her. Wherever she may be.She can search this whole wide world over - she won't ever find another man like me.

46. I just walked in to find you here with that sad look upon your face.

47. Take a jumbo across the water like to see America.

48. Goodbye stranger it's been nice. Hope you find your paradise. Tried to see your point of view, hope your dreams will all come true.

49. The attorney came up with several far-fetched arguments in a vain attempt to buttress his weak case.

50. Read rhymes with lead, and read rhymes with lead, but read and lead don't rhyme, and neither do read and lead.

Bibliography

Aaron van den Oord Nal Kalchbrenner, Koray Kavukcuoglu (2016). *Pixel Recurrent Neural Networks*. URL: https://arxiv.org/abs/1601.06759.

Arik, Sercan O. et al. (2017). *Deep Voice: Real-time Neural Text-to-Speech*. URL: https://arxiv.org/abs/1702.07825.

Bolaños, Marc et al. (2017). *Egocentric Video Description based on Temporally-Linked Sequences*. URL: https://www.researchgate.net/publication/315838647_Egocentric_Video_Description_based_on_Temporally-Linked_Sequences.

Cho, Kyunghyun et al. (2014). *Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translationc*. URL: https://arxiv.org/abs/1406.1078.

Etienne, Caroline et al. (2018). *CNN+LSTM Architecture for Speech Emotion Recognition with Data Augmentation*. URL: https://arxiv.org/abs/1802.05630v2.

Goodfellow, Ian, Yoshua Bengio, and Aaron Courville (2016). *Deep Learning*. MIT Press.

Griffin, Daniel W. and Jae S. Lim (1983). "Signal estimation from modified short-time Fourier transform". In: *ICASSP*.

Hochreiter, Sepp and Jürgen Schmidhuber (1997). *Long short-term memory*. Neural Computation. URL: https://www.mitpressjournals.org/doi/10.1162/neco.1997.9.8.1735.

Huang, Zhelin et al. (2019). *Convolutional gated recurrent unit -recurrent neural network for state-of-charge estimation of lithium-ion batteries*. URL: https://www.researchgate.net/publication/334385520_Convolutional_gated_recurrent_unit_-recurrent_neural_network_for_state-of-charge_estimation_of_lithium-ion_batteries.

Ito, Keith (2017). *The LJ Speech Dataset*. https://keithito.com/LJ-Speech-Dataset/.

Jia, Ye et al. (2019). *Transfer Learning from Speaker Verification to Multispeaker Text-To-Speech Synthesis*. URL: https://arxiv.org/abs/1806.04558.

Kalchbrenner, Nal et al. (2018). *Efficient Neural Audio Synthesis*. URL: https://arxiv.org/abs/1802.08435.

Lecun, Yann et al. (1998). "Gradient-based learning applied to document recognition". In: *Proceedings of the IEEE*, pp. 2278–2324.

Livingstone, Steven and Frank Russo (May 2018). "The Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS): A dynamic, multimodal set of facial and vocal expressions in North American English". In: *PLOS ONE* 13, e0196391. DOI: 10.1371/journal.pone.0196391.

Nagrani, A., J. S. Chung, and A. Zisserman (2017). "VoxCeleb: a large-scale speaker identification dataset". In: *INTERSPEECH*.

— (2018). "VoxCeleb2: Deep Speaker Recognition". In: *INTERSPEECH*.

O'Shaughnessy, Douglas (1987). *Speech communication: human and machine*. Addison-Wesley, p. 150. ISBN: 978-0-201-16520-3.

Olson, Harry Ferdinand (1967). *Music, Physics and Engineering*. Dover Publications, pp. 248–251. ISBN: 978-0-486-21769-7.

Oord, Aaron van den et al. (2016). *WaveNet: A Generative Model for Raw Audio*. URL: https://arxiv.org/abs/1609.03499.

Oord, Aaron van den et al. (2017). *Parallel WaveNet: Fast High-Fidelity Speech Synthesis*. URL: https://arxiv.org/abs/1711.10433.

Panayotov, Vassil et al. (Apr. 2015). "Librispeech: An ASR corpus based on public domain audio books". In: pp. 5206–5210. DOI: 10.1109/ICASSP.2015.7178964.

Rahman, Matiur (2011). *Applications of Fourier Transforms to Generalized Functions*. WIT Press. ISBN: 978-1-84564-564-9.

Sejdića, Ervin, Igor Djurović, and Jin Jiang (2009). *Time–frequency feature representation using energy concentration: An overview of recent advances*. URL: https://doi.org/10.1016/j.dsp.2007.12.004.

Shen, Jonathan et al. (2018). *Natural TTS Synthesis by Conditioning WaveNet on Mel Spectrogram Predictions*. URL: https://arxiv.org/abs/1712.05884.

Spanias, Andreas, Ted Painter, and Atti Venkatraman (2006). *Audio Signal Processing and Coding*. John Wiley & Sons. ISBN: 047004196X, 9780470041963.

Stevens, Stanley Smith et al. (1937). *A scale for the measurement of the psychological magnitude pitch*. URL: https://archive.is/20130414065947/http://asadl.org/jasa/resource/1/jasman/v8/i3/p185_s1.

Tokuda, Keiichi et al. (2013). *Speech Synthesis Based on Hidden Markov Models*. URL: https://ieeexplore.ieee.org/document/6495700.

Wan, Li et al. (2018). *Generalized End-to-End Loss for Speaker Verification*. URL: https://arxiv.org/abs/1710.10467.

Wan, Yuxuan et al. (2018). *Style Tokens: Unsupervised Style Modeling, Control and Transfer in End-to-End Speech Synthesis*. URL: https://arxiv.org/abs/1803.09017.

Wang, Yuxuan et al. (2017). *Tacotron: Towards End-to-End Speech Synthesis*. URL: https://arxiv.org/abs/1703.10135.

Yoon, Seunghyun, Seokhyun Byun, and Kyomin Jung (2018). *Multimodal Speech Emotion Recognition Using Audio and Text*. URL: https://arxiv.org/abs/1810.04635v1.

Zhang, Julia (2004). "Language Generation and Speech Synthesis in Dialogues for Language Learning". URL: http://groups.csail.mit.edu/sls/publications/2004/zhang_thesis.pdf.