Available online at www.garmian.edu.krd

Passer Journal

Passer 4 (Issue 2) (2022) 150-160

http://passer.garmian.edu.krd/

End-to-End Kurdish Speech Synthesis Based on Transfer Learning

Hadi Veisi[1*], Sabat Muhamad[2]

[1]*Faculty of New Sciences and Technologies, University of Tehran, Tehran, Iran*

[2]*Computer Science Department, Faculty of Science, Soran University, Kurdistan Region, Iraq*

Received 21 July 2022; revised 21 August 2022;
accepted 23 August 2022; available online 26 August 2022

doi:10.24271/psr.53

ABSTRACT

A text-to-speech (TTS) system converts the texts into speech in a specific language. Several TTS systems generate natural-like speech signals in numerous languages, such as English. On the other hand, the Kurdish language has just been examined. Existing preliminary research on Kurdish speech synthesis has utilized old methods and has generated low-quality speech. They also lack important aspects of speech, including intonation, emphasis, and rhythm. Some approaches were presented to address these challenges, including the use of concatenative systems. For example, the unit selection or statistical parametric methods. On the other hand, they need a great deal of time, effort, and domain knowledge. An additional factor for Kurdish speech synthesizers' low performance is the absence of publicly available speech corpora, unlike English, which has many freely-available corpora and audiobooks. The motivation of this paper is to create a Central Kurdish speech corpus and generate a human-like speech from the Kurdish text. This paper explains how to utilize Tacotron 2, an end-to-end neural network architecture and HiFi-GAN vocoder, to produce a high-quality, realistic, and human-like Kurdish voice. This work utilizes "text, audio" pairings, which contain 10 hours of recorded audio samples and texts collected from the Internet and textbooks. It shows how to use English character embedding as the pre-trained knowledge with Kurdish characters as input and how to preprocess these audio examples to get a great outcome. Our evaluations for various types of texts show a mean opinion score of 4.1, comparable with state-of-the-art synthesizers in other languages.

Keywords: Central Kurdish Speech Synthesis, Deep Learning Text-to-Speech, Tacotron 2, HiFi-GAN, Transfer-Learning, End-to-End.

1. Introduction

Speech synthesis is defined as the procedure of a machine for automatically producing spoken language. Text-to-speech (TTS) connectivity is another name for it. A text that is normally spoken is converted into a voice in this process[1]. Speech synthesis aims to create a system with a natural-sounding voice that can communicate with humans[2]. In the last twenty years, top-quality speech synthesized from electronic text has been focused on by researchers, leading to an expanding range of applications. Commercial telephone answering services, normal language programming interfaces, reading devices for the blinds and even further handicapped assistance, language acquisition systems, digital apps, audiobooks, and talking toys, and so on are some examples[3]. Different methods have contributed to the field over the years, also or several years, concatenative synthesis through unit-selections, the technique of linking together minor units of pre-recorded waveforms, was state-of the art[4, 5]. Then, statistical parametric speech synthesis[6, 7] was proposed, producing smooth trajectories of the speech characteristics to be synthesized directly by the vocoder, excluding several of the boundary artifacts concatenative synthesis has. Nevertheless, when compared to the human voice, the audio produced by the mentioned models is regularly muted, and they are unnatural. TTS methods regarding end-to-end neural-network architecture have controlled the market in recent years[8, 9]. WaveNet[10], the generative system of time-frequency wave-forms, generates audio performance that approaches actual human speech and is previously utilized in some TTS applications[11]. Tacotron, a sequence-to-sequence architecture[12] for generating magnitude spectrograms from a series of types, streamlines the classical speech synthesis pipeline by swapping the output of these linguistic and acoustic characteristics with an individual neural network learned solely on input. End-to-end voice synthesis is improved by Tacotron 2[8], an enhanced version of the Tacotron, and another neural network-based technology that synthesizes speech directly from the text. Convolutional neural networks (CNN) and recurrent neural networks (RNN) are used to create this architecture. Text-to-speech systems utilizing the Tacotron2 deep-neural-network architecture have been effectively developed and applied for

* Corresponding author
E-mail address: h.veisi@ut.ac.ir (Instructor).
Peer-reviewed under the responsibility of the University of Garmian.

various languages like English[8], Chinese[13], Arabic [14], and Persian[15]. On the other hand, Kurdish, for example, has gotten significantly less attention than other languages. So far, no research has been done on Kurdish TTS using deep learning, and this work will be the first work in this field using the presented method. In our scenario, we do not have a sufficient dataset to train a Kurdish end-to-end speech synthesizer. As a result, we gathered an overall ten hours of Central Kurdish speech data together with the associated text. We also recommend using transfer learning approaches from the previously released pre-trained Tacoron2 English model to train the proposed model for faster model convergence and more accurate pronunciation modeling. This research demonstrates how to produce Mel-spectrograms from Central Kurdish text as an in-between feature illustration, then use a HiFi-GAN architecture for a vocoder to generate a high-quality Kurdish voice using a deep architecture from Tacotron2.

2. Background of speech synthesis

The most common method used for communication among humans is through speech[16]. The process of transforming a text into a voice is known as synthesis. The text is converted to simulated speech that is as similar to human speech as possible while adhering to special language pronunciation norms[17]. For decades, voice synthesis, or the automated processing of speech waveforms, has been a work in progress. However, the standard of current models has improved to the point that they are suitable for a variety of uses, including digital and telecommunications. Speech technology is a branch of natural language processing (NLP) that covers related applications such as speech synthesis, speech recognition, and dialog systems[18]. In a TTS system, the speech is formed by passing the input text through all of the steps presented in Figure 1.

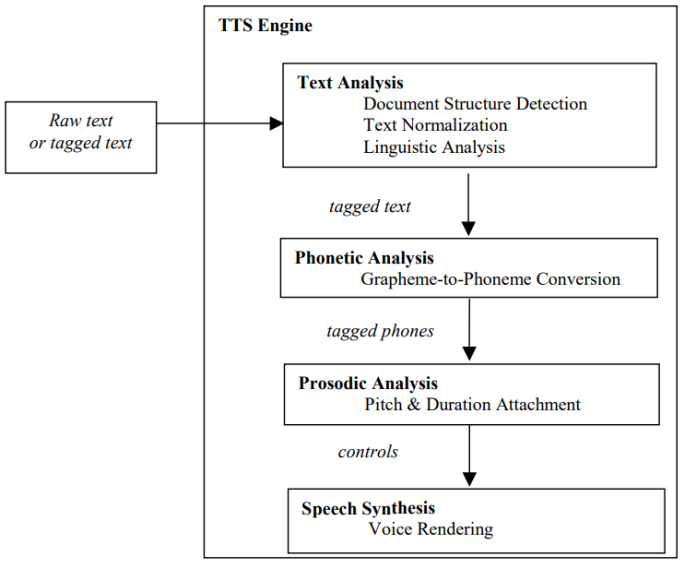

Figure 1: A summary of a TTS system architecture[19].

The following are the most popular forms of synthesis systems:

1. ***Formant***: This is the earliest approach for speech synthesis, and for a long period of time, it controlled synthesis implementations. Formant-synthesis is a rule-based method of describing the vocal tract's resonant frequencies. This approach employs a language output source-filter model. This technique creates an artificial speech waveform by combining parameters including fundamental pitch, voicing, and noise levels over a period of time[1].

2. ***Articulatory-Synthesis***: Articulatory synthesis produces speech directly by displaying the actions of human articulators, making it the most satisfying approach for producing high-quality speech in theory. Lip aperture, lip protrusion, tongue tip position, tongue tip-height, tongue position, and tongue height are all articulatory function parameters[20]. In articulatory synthesis, there are two challenges. The initial challenge is collecting a dataset for the articulatory model. The second challenge is striking the balance between the highly precise system and an easy-to-design and-control model[21].

3. ***Concatenated***: Concatenative synthesis follows the data-driven method. By linking natural, pre-recorded-speech units, concatenative-synthesis generates voice. Words, syllables, demisyllables, phonemes, diphones, and triphones are examples of these units. The duration of the device affects the consistency of the synthesized voice. Longer units have more genuineness, need fewer concatenation points, but require additional space, and the several units contained within the database grow rapidly. Shorter units need less space, but sample collection and marking methods have become more complicated[22].

4. ***Statistical Parametric (Hidden Markov Model)***: Another choice is for inferring specification-to-parametric mapping from the dataset using statistical-parametric synthesis methods. These methods have two advantages: first, storing the parameter of a model requires less memory than storing the dataset itself. Second, further combinations were possible, such as converting the original voice into a different voice. HMM synthesis is considered one of the most popular utilized statistical-parametric-synthetic methods. The most often used categories of features are the Mel frequency cepstral coefficient (MFCC) and the main and second derivatives[22].

5. ***Deep Learning based Speech Synthesis***: Unlike the HMM-based approach, the deep learning (DL) based technique uses deep neural networks to explicitly plot linguistic features to auditory features. Deep neural networks have proved to be extremely effective at studying inherent data features. Many models have been suggested in previous research that uses a DL-based approach for speech synthesis[23].

5.1 ***End-to-End Speech Synthesis***: The text analysis front end, an acoustic model, and a speech synthesizer have been used as the most common components of a TTS framework. While these mechanisms have been trained separately and depended on time-consuming domain knowledge, errors from every element can compound. Furthermore, for solving these issues, end-to-end speech synthesis approaches, which integrate certain components into a

single structure, are becoming popular in the speech synthesis area[23].

The following parts cover a short overview of the end-to-end speech synthesis approaches.

5.1.1 ***Wave-Net***: Wave-Net is a sophisticated generative system of raw-audio waveforms that emerged from the Pixel-CNN or Pixel-RNN model used in picture production. DeepMind suggested it in 2016, and this allows end-to-end voice synthesis. This can generate somewhat realistic sounding human-like sounds by straight modeling wave-forms utilizing the DNN model trained on real-world speech samples[24].

5.1.2 ***Tacotron***: Tacotron is the complete speech synthesis system. It can train the speech synthesis model from texts and audio sets, obviating the necessity for time-consuming feature engineering. Furthermore, meanwhile, it is character-based; it could be used in nearly any language[23].

The Tacotron-model, like WaveNet, is a generative model. Unlike WaveNet, Tacotron maps text to a spectrogram, which is a strong approximation of voice, using a seq2seq model with an attention function. Since a spectrogram lacks phase information, the method reconstructs the audio using the Griffin–Lim algorithm[25], Iteratively extracts phase parameters from the spectrogram. Tacotron 2 is an extended version of Tacotron proposed by[8].

2.1 Tacotron

Tacotron2 is fully end-to-end speech synthesis. Furthermore, Tacotron2 TTS acoustic model is used to build the TTS system. Also, a HiFi-GAN[26] implementation is a slightly modified version of the model offered in[8]. As a result, the model in Figure 2 is divided into two parts:

1. A sequence-to-sequence architecture spectrogram prediction network uses attention for predicting the corresponding Mel-spectrogram from an input text (i.e., Central Kurdish text).

2. HiFi-GAN, a program that accepts Mel-spectrograms as input and generates a time domain wave-form of the texts.

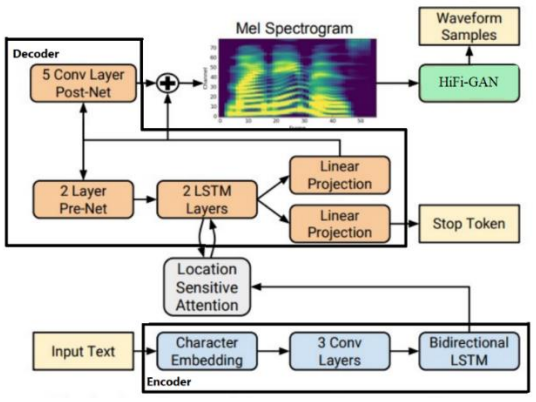

Figure 2: Block-diagram of the spectrogram-prediction-network with HiFi-GAN, which accepts letters as inputs and outputs an audio wave-form[8].

2.2.1 Spectrogram Prediction Network

The Mel-spectrogram prediction network has the sequence to-sequence architectures, as illustrated in Figure 2. It has an encoder, that generates the internal representation of the input signals, which is then given to the decoder, which generates the expected Mel-spectrogram. Character embedding, 3 convolution layers, and bidirectional long short-term memory LSTM are the three components of the encoder. It generates a hidden feature vector representation from a character sequence as input. A two-layer LSTM network, two layer pre-net, five Conv-layer PostNet, and linear progression make up the decoder. It takes the encoder's hidden feature vector representation and creates Mel-spectrograms for the provided input letters.

The embedding layer (512-dimensional-vector) is the next component in the design, and it numerically represents each character symbol. Because any word could be written with the letters, each word vector may be produced even if it's an out-of-vocabulary in character embedding, but word implanting handles observed words better. Character-embedding is also the greatest option for managing misspelled words, uncommon words, emoticons, new terms, and even jargon. Character embeddings of sizes 512 are used in this model to handle text. The embedding layer's output is routed to 3 convolutional layers, each with five hundred twelve-dimension filters of 5 X 1 to simulate long-term contexts (N-gram) and span 5 characters. Batch-normalization[27] and ReLU-activation[28], are applied to each convolutional layer. The last convolutional layer's output is passed into a 512-unit bi-directional LSTM (two hundred fifty-six in any direction). The forward and backward findings are combined to create encoded features for the decoder to use. A hybrid-attention model proposed in[29] is used by the Spectrogram Prediction Network. Because it is difficult for encoders-decoders architecture without concentration for memorizing extended input sequences, an attention mechanism is required for the spectrogram prediction network to solve long sequence difficulties (long character sequences). As a result, over long sequences, the architecture's performance without an attention mechanism would decline. Long sequences are solved by the attention mechanism by paying to a segment of the sequences (using attention weight) in the same way as humans do while attempting to understand lengthy sequences[14].

The output of the decoder layer is sent into the pre-net, which is made up of 2 completely linked layers with two hundred fifty-six hidden ReLU-units each, followed by two unidirectional LSTM with 1024 units each. To estimate Mel-spectrogram, the concatenation of LSTM output and contextual vector is anticipated to the linear transformations, which are then sent to a 5-layer PostNet. By anticipating the concatenation of the context-vector with the decoder LSTM out-put and running them through the sigmoid-activation, a stop token is computed in parallel to anticipate when to completely stop generating speech at inference time. Mel-spectrogram is generated utilizing a Fifty m-s frame hub and the Han-window operator[8].

Dropout[30] is used to control all convolutional layers, whereas zoneout[31] is used to control LSTM layers.

2.2.2 HiFi-GAN vocoder

HiFi-GAN-based vocoder is used instead of a WaveGlow-based vocoder to improve vocoding quality and efficiency. The network design resembles that of configuration V1[26] The generator takes a Mel spectrogram as input and upsamples it using transposed convolutions until the length of the output sequence matches the original waveform's temporal resolution[32].

3. Related works

In this section, we first shortly review the related deep learning-based TTS systems, and then the related TTS researches in the Kurdish language are presented.

Wave-Net, a deep neural network for creating raw audio waveforms, was presented in[10]. They utilized a North American English dataset with 24.6 hours dataset and a Mandarin-Chinese dataset with 34.8 hours of speech data. Subjective paired comparison test and mean-opinion score (MOS) evaluation were used to evaluate WaveNets' performances. WaveNet exceeded both baseline statistical-parametric and concatenative-voice-synthesizers in both languages. In[33], ClariNet is presented which is a new parallel wave generating approach built on the Gaussian inverse autoregressive flow (IAF). Utilized an internal English speech dataset with around twenty hours of audio. It outperformed the prior pipeline, which linked a text-to-spectrograms model to a WaveNet that had been individually trained. They also succeed in distilling a parallel-waveform synthesizer conditioned on the hidden representation.

Tacotron, an end-to-end Text To Speech model, was presented in[34]; they used 24.6 hours of speech data from an existing North-American-English dataset to train Tacotron. For the evaluations, 100 unseen phrases were used, with each phrase receiving eight points. As compared to previous schemes, Tacotron reaches a MOS of 3.82, outperforming the parametric system. In[8] defines Tacotron2, the completely neural TTS schemes that combined the seq-to-seq recurrent network with care for predicting Mel spectrograms through an adapted Wave-Net-vocoder. All of their systems were trained on the normalized texts and were based on 24.6 hours of an inner-USA-English dataset. Their model gets a MOS of (4.53), which is equivalent to the MOS of (4.58) for a professional reported voice.

In (2019), the authors in[35] presented a method for incorporating prosodic annotation into the Tacotron model to produce rhythmed and natural Chinese expressions. They trained the system on the BZSYP dataset for 10.38 hours. To extract the acoustic parameter from audio, the Librosa Python package was used. The test of their proposed approach on native speakers reveals that it outperforms the baseline system educated without prosodic annotation. To enhance the prosodic phrasing of the Tacotron-based TTSmodel[36], suggested the novel two-task learning scheme. For Chinese, they used TH-CoSS (TsingHua Corpus of Speech Synthesis). They used the 03FR00 subset of TH-CoSS, which includes about nine hours of speech dataset. In sum, Mongolian speech data comprised about 17hours of voice. The listening exercises included twenty Chinese and fifteen Mongolian utterers. Their proposed system reliably outperforms all contrastive structures.

Tacotron 2 and Wavenet-vocoder were used in[13] work on Chinese end-2-end speech synthesis. The dataset consists of 31 hours of transcribed Chinese-female voices. To enhance prosodic phrasing, the three suggested contexts (Full-Sen vs. P-Word, P-Word vs. N-gram, N-gram vs. Baseline) have been used. The FullSen approach was the most powerful of the bunch. In[37], they present the teacher-student-training system. For a quick turnaround, they employed the Griffin-Lim algorithm for waveform creation in all of their studies. They ran several tests on both Chinese and English to determine the natural-ness and robustness of the language. For both languages, the suggested Tacotron-2-KD (knowledge-distillation) framework reliably outperforms the baseline systems.

DOPTacotron, the fast end-to-end Chinese Text to speech model a focus on the local area, has been suggested in[38]. All of the experiments in this study were conducted on the 12 hours of biaobei speech corpus. They chose 50 sentences at random as the evaluation set. The MOS of DOP Tacotron is 3.683, which is higher than that of Tacotron[39] suggested the system for end-to-end normalized TTS wave-form-synthesis. Two single speaker datasets were used: a proprietary dataset including around 39-hour of speech utters, and the public LJ-speech dataset. Experiments reveal that the suggested model creates voice with a quality that is equivalent to the state-of-the-art neural TTS model but at a substantially faster rate.

Tacotron 2 was utilized in[40] where they trained their model on 5 hours of Myanmar corpus. Their result was MOS=3.89. In[14] used Tacotron 2 for generating high-quality and human-like Arabic speech. They trained the model on 2.41 hours of Nawar Halabis Arabic dataset by utilizing a pre-trained English model. They achieved a MOS of 4.21. A unique TTS system based on Tacotron 2 was proposed in[15] for Persian. They created 21 hours of Persian speech dataset to train their model. They obtained different values when evaluating their model from MOS 3.01 to MOS 3.97.

3.1 Kurdish TTS Works

Until now, Kurdish Text-to-Speech is in the early stages, with less study conducted in comparison to other languages. Kurdish is an Iranian language that belongs to the Indian European language family[41]. There are 29 consonants and 8 vowels in Kurdish. There are two scripts in this language: a modified Arabic alphabet, and a modified Latin alphabet[42].

Several synthesis models based on allophones, syllables, and diphones for the Kurdish language were developed in[43], The allophone-based model had the lowest quality, and in actuality, it was the most difficult to use. The syllable-based method had a good overall quality and high intelligibility. And between all three systems, the diphone-based TTS system had the best quality. In the same year in[44], a comparison was made between the three systems of allophone, syllable, and diphone for the Kurdish TTS system through the use of concatenation. The diphone-based TTS system had the best quality and best Diagnostic-Rhyme-Test DRT (97%). Also, in the same year in[45], they used the allophone unit in their concatenative method. Their result showed that the produced speech has a great score of intelligibility.

In[46] to have more natural speech they used the concatenative synthesis method. To improve the transition between phonemes, they employ diphone units in their Concatenative technique. And their result showed that the produced speech has a good score of intelligibility. A summary of some related works is outlined and is shown their main points in Table 1.

Table 1: A summary of the related works for TTS.

No.	References	Year	Method	Dataset and languages	result
1	[10]	2016	WaveNet	North-American English (24.6 hrs). Mandarin-Chinese dataset (34.8 hrs).	MOS 4.0
2	[33]	2019	Clarinet: Parallel WaveNet	An internal English speech dataset (20 hrs)	MOS 4.15
3	[34]	2017	Tacotron End-to-End	North American English dataset	MOS 3.82
4	[8]	2018	Tacotron-2	US English dataset (24.6 hrs)	MOS 4.526
5	[35]	2019	Tacotron	BZSYP-Chinese database.	84%
6	[36]	2020	Multi-task-learning (MTL) Tacotron	Chinese-TH-CoSS (TsingHua-Corpus) (9 hrs) Mongolian speech-data (17 hrs)	MOS-Chinese 3.91 MOS-Mongolian 3.83
7	[13]	2019	Tacotron-2 and Wavenet vocoder.	Chinese dataset (31 hrs)	Significant (p = 0.001)
8	[37]	2020	Tacotron-2-KD	English and Chines dataset	MOS-English 3.93 MOS-Chinese 3.94
9	[38]	2020	DOP-Tacotron	Biaobei speech corpus-Mandarin (12 hrs)	MOS 3.683
10	[39]	2021	Wav-Tacotron	(39 hrs) of speech and the public LJ-speech dataset.	MOS-char 4.07 MOS-phone 4.23
11	[40]	2020	Tacotron2	Myanmar corpus (5 hrs)	MOS 3.89
12	[14]	2020	Tacotron 2-Transfer Learning	Nawar Halabi's Arabic Dataset (3 hrs)	MOS 4.21
13	[15]	2022	Tacotron 2	Persian dataset (21 hrs)	MOS 3.01 – 3.97
14	[43]	2009	Concatenative (Allophone, Syllable and Diphone)	Kurdish Language	Allophone MOS 2.45 Syllable MOS 3.02 Diphone MOS 3.51
15	[44]	2009	Concatenative (Allophone, Syllable and Diphone)	Kurdish Language	Best quality score 3.5 Best DRT 97%
16	[45]	2009	Concatenative (Allophone)	Kurdish Language (2100 words)	Best quality score 2.4
17	[46]	2011	Concatenative (Diphone)	Kurdish Language (2100 words)	Best quality score 55%

4. Research methodology

In this section, we discuss the processes in our suggested technique in detail, such as the collected data for the Central Kurdish corpus. In other steps, another aspect has been the reconstruction of end-to-end speech synthesis model. The general steps of the proposed model are illustrated in Figure 3.

4.1 Creation of Speech Corpus

A speech corpus is the main data set source for creating the text-to-speech system. This study provides the first Central Kurdish voice corpus for TTS systems. First, we create a text and audio pairings dataset with one female speaker over 30 years of age with a bachelor's degree. For the text corpus, we collected 4652 sentences from a set of texts in 14 various categories, including news, sport, linguistics and psychology, poem, health, question and exclamation sentences, science, everything, general information, interview, politics, education and literature, story, and tourism, to create the train sentences. These sentences are compiled from several web sources and then improved. Table 2 shows the subjects of the chosen sentences as well as the number of sentences for each subject. A sample of the sentence is shown in Table 3.

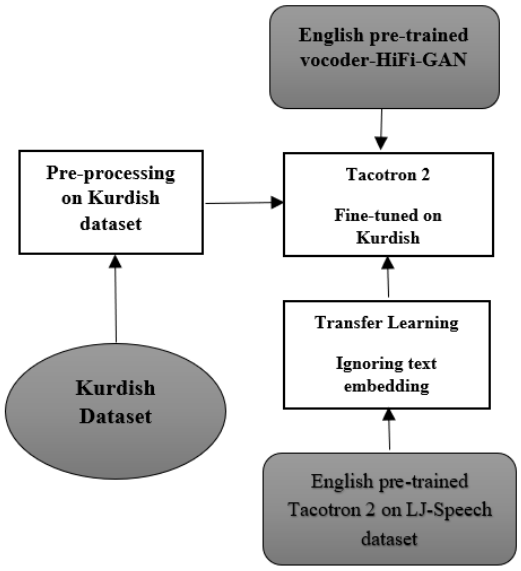

Figure 3: Block diagram of our approach.

Table 2: The train sentences of the speech corpus.

Topics	No. of Sentences
News	306
Sport	452
Linguistics	1136
Poem	215
Health	348
Questions	129
Interview	241
Science	186
Everything	391
General information	102
Politics	250
Education and literature	519
Story	246
Tourism	131
Total	**4652**

Table 3: Examples of the designed train sentences.

Topic	Kurdish	English
News	زانکۆکه له ساڵی دوو هەزارو حەفدەوه دەستیکردوه به تاوتوێکردنی سکاڵاکان.	The university has been discussing complaints since two thousand seven.
Sport	چافی هێرناندێز وەک راهێنەری نوێی یانەی بارسێلۆنا ناسێندرا.	Chevy Hernandez was introduced as Barcelona's new coach
Politics	زۆر له نازادی دەترسین.	We are so frightened of freedom.

To create the test set, we gathered 110 sentences from a set of documents from 17 different areas. These sentences are compiled from a variety of web sources and then polished. Table 4 lists the themes of the chosen sentences as well as the number of sentences for each subject.

Table 4: The distribution of test sentences.

Topics	No. of Sentences
News	10
Sport	9
Linguistics	5
Psychology	6
Poem	8
Health	6
Questions	7
Exclamation	4
Science	6
Everything	6
General information	6
Interview	5
Politics	5
Education and literature	5
Story	6
Tourism	6
Formal letter (SMS)	10
Total	**110**

We recorded audios in a voice recording studio at 44100 Hz, and all audio files are down-sampled to 22050 Hz in our modeling process. The audio ranges from one to twelve seconds in length. We generate the speech corpus in this method, and the last speech has about 4652 texts and audio pairings, which takes around 10 hours. Figure 4 displays the distribution of long sentences in the dataset.

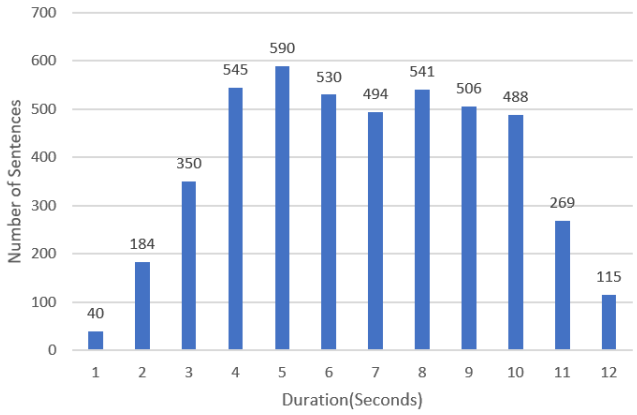

Figure 4: Distribution of length of sentences of the dataset.

Altogether audio files are saved in wave format, and the texts are saved in text files in the corresponding sub-folders. Furthermore, for model training, all of the audio files are gathered in a single folder. Each line in the transcript files is formatted as wavs| audio file's name.wav| transcript. The audio file's name includes the extensions, and the transcript was the speech's text. The transcript file is divided into two *.txt files for both training and testing the system. In the training file, there were 4,500 transcript lines and 152 in the validation file.

4.2 Pre-Processing

Kurdish orthography has yet to be entirely standardized, despite various attempts. As a result, many Central Kurdish terms have various spellings. So, unified textual forms must be determined as the TTS engine's input. Text normalization is a process to unify the TTS input and acts as an essential step in improving the quality of TTS models. The normalization is much more important when it comes to the Kurdish Language since Kurdish writers and publishers utilize a variety of encoding schemes and orthographic standards[47]. To develop a text-to-speech system, we do text normalization, and the details of normalization completed on the text corpus are presented in[48,49] in our pre-processing stage. Some Kurdish writing includes a variety of numerical forms, such as date, time, and amounts. In this instance, text normalization is required; for example, ("۲۰۲۱-۱۱-۱") is changed to (" یەکی یانزەی دوو هەزارو بیست و یەک"). This is an essential aspect of the Kurdish text-to-speech system's pre-processing step, where the numbers in the text dataset must be turned into spoken syllables. Table 5 shows a few examples of normalized Kurdish text.

Table 5: Examples of Kurdish Text Normalization.

Input -text	Normalized-text	English
۱۱	یانزە	Eleven
۱-۱۱-۲۰۲۱	یەکی یانزەی دوو هەزارو بیست و یەک	First of November two thousand twenty-one
٥.٦٣	پێنج پۆینت شەست و سێ	Five point sixty-three
محمد ئیمسأل قۆناغی چوارە	محەممەد ئیمسأل قۆناغی چوارە	Muhammed is in the fourth grade this year

Audio files are going to be converted to Mel-scale spectrogram after text data has been properly normalized. As an example, Table 6 depicts the contents of training data.

Table 6: Example of training collected data.

Kurdish	English
مام حاجیەک هەموو رەمەزانێک سی دەنک مێوژ دەکاتە گیرفانی و هەموو رۆژێکی رەمەزان تێپەڕبێت دەنکێکی دەخوات بۆ ئەوەی بزانێت چەند ڕۆژی ماوە بۆ جەژن؟	The elderly man fills his pockets with thirty raisin seeds and eats each as each day passes so that he can know when the eid is?
دوای چەند ڕۆژێک، جەماعەتێ لێی دەپرسن مام حاجی چەند ڕۆژی ماوە بۆ جەژن؟	After a few days, a group will ask the elderly man how many days are left for the feast/eid?
ئێمە دەرۆزەکارین و ئەمەش ڕاستە!-مارتن لوسەر	We are beggars, and this is true!-Martin Luther

4.3 End-to-End Speech Synthesis

Tacotron2 and HiFi-GAN appeared to be among the most appropriate deep learning-based end-to-end models for creating natural-sounding speech. We utilize the Pytorch implementations. Separate training is required for these two modules. Section 4.3.1 discusses the Tacotron2 model's training and the transfer learning approach. There is no intentional training of the vocoder in this work. The model is a pre-trained HiFi-GAN model that was trained on the equivalent substantial English dataset as the Tacotron-2. The HiFi-GAN vocoder is generally stable over unknown languages and speakers. Using a pre-trained model also minimizes the computational load. In the following subsection, transfer learning from the English pre-trained model is discussed in detail.

4.3.1 Transfer Learning from an English Model

The use of transfer learning is described in this section, as well as in what way it is used to fine-tune the pre-trained model on the Kurdish dataset. The pre-trained model was trained using the openly accessible LJ Speech dataset, which includes 24 hours of single-speaker female speech and transcripts. Transfer learning from a pre-trained model entails fine-tuning using specific components of the pre-trained model. This decision is made based on the work at hand. The optimizer details and text embedding weights were not included in this study, and the rest of the pre-trained model is utilized to fine-tune the dataset. These embedding weights capture textual information, which is a dataset reliant on and independent of speaker or speaking style. Because the pre-trained model was trained in English, it's clear to disregard these and fine-tune solely the pre-trained model's speech features. In comparison to random initialization of the model parameter, transfer learning has a considerable influence on model convergence in low-resource environments.

The pre-trained HiFi-GAN model is utilized in the vocoder, as mentioned in paragraph 2.1.2, because it is only employed throughout inference and is resilient to alteration of genders and languages; thus, employing it straight appears to be a justifiable decision. This minimizes both the computational load and the total training time.

5. Results and discussion

5.1 Training Set

We use a Central Kurdish dataset to train all of our models, which comprises ten hours of speech from a single female speaker. The collection is made up of text and audio pairings. The input texts are Central Kurdish characters, with a 16-bit 22050 Hz of sampling rate output with a bit rate of 352 kbps. The audio ranges from 1 to 12 seconds in length.

The dataset location, as well as the training and validation files, were also supplied. It ought to be mentioned that the Tacotron 2 models were initially learned entirely on Google-Colaboratory, the free-TensorFlow-compatible platforms, and later on a place in a high-performance computing environment with a GPU Nvidia GeForce RTX 3080. All texts are normalized, meaning

that any number and non-Kurdish characters are written in Kurdish. After preprocessing, sets of numerical sequences in NumPy arrays and Mel-Spectrograms recorded in NumPy files (.npy) are what we receive.

By (a) converting Central Kurdish words into English characters, we used transfer learning from English. (b) completely training the attention mechanism utilizing the pre-trained English model[1] with the learned English characters integration. The audio training samples were down-sampled to 22050-Hz in order to use the similar audio parameter as an open-source implementation[2] (trained on the LJS-speech dataset), such as hop length and filter length.

Based on the related literature in[50, 51, 13], it is sufficient to train up to 100K iterations to produce high-quality speech. We trained our model three times using a fixed batch size of 8 and three different numbers of epochs. The first model is trained with 100 epochs (50,000 iterations), the second one with 300 epochs (150,000 iterations), and the last one used 500 epochs (250,000 iterations). The number of epochs is an important parameter in neural network training that controls the trade-off between model quality and network overfitting.

Although there are other methods, such as using a validation set to optimize this parameter and perform early stopping, considering that this parameter is determined in a try and error manner in Tacotron2, we have tested different values in this research to find the better one. For this aim, three values of 100, 300, and 500 have been evaluated and the selection of these values is based on similar experiences in similar tasks in other languages[50, 51, 13]. The right batch size parameters that the previous researchers used in their works were 32 and above. For our training, selecting the right batch size is crucial. The quality of the outcome suffers when the mini-batch size is reduced from 32 to 8. As a result, if we want to get high-quality findings, we would create the min batch size as large as feasible. The precision with which the input letters and the output waveform structures are aligned is referred to as the alignment graph. A diagonal alignment map shows that the models can produce understandable speech because they have learned how to solve the seq2seq problem between the input text (encoder stages) and output spectrogram (decoder stages). Throughout the training procedure, it is critical to keep an eye on the alignment plots; if they do not appear to be linear, the training should be redone. Some alignment graphs derived from our suggested model are shown in Figure 5.

A training period took roughly about 20 minutes on average for each epoch, whereas generating a waveform took just about 2 seconds. Other training parameters are presented in Table 7. The values of the dropout and learning rate are obtained in a tray an error manner, and the values of other parameters are taken from previous similar works[8].

In the second stage of the experiment, the output of the synthesized speech is assessed for correctness and naturalness using a mean opinion score (MOS).

Table 7: Final values of the training model hyperparameters.

Hyperparameter	Value
Epochs	500
Batch-size	8
Attention dropout	0.1
Decoder dropout	0.1
Decay start	15000
Learning rate	1e-5
Weight decay	1e-6

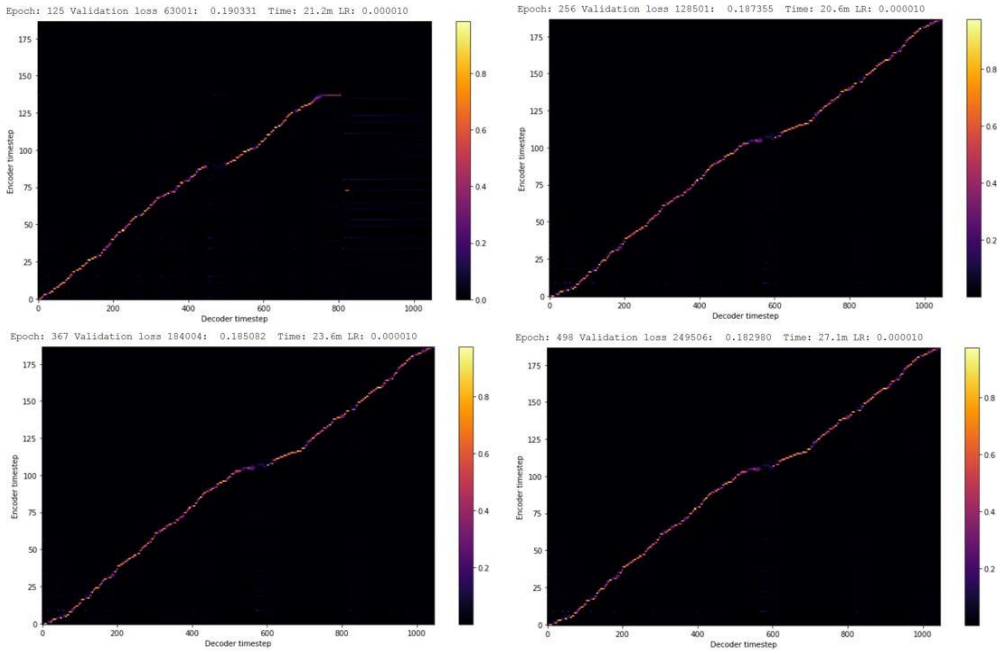

Figure 5: Examples of the alignments at different steps of the training model.

[1] https://drive.google.com/file/d/1bwL6Bz8Yohs_iCjWCk0JRPruBZUVsXH4/view?usp=sharing

[2] https://github.com/NVIDIA/tacotron2

5.2 Evaluation Results

The researchers performed mean-opinion-score (MOS) evaluations, in which participants were requested to rate the naturalness of the stimulus on the five-point Likert scale. Native speakers were enlisted to help with the MOS examinations. For the testing, 110 unseen sentences are utilized, with each phrase receiving five scores (5: very good, 4: good, 3: neutral, 2: bad, 1: very bad). When calculating MOS, we only consider ratings from those who used headphones. Equation (1) is used to determine the intelligibility and naturalness evaluation for MOS.

$$MOS = \frac{\sum_{j=1}^{s} \left[\frac{\sum_{i=1}^{N} R_{ij}}{N} \right]}{s} \quad (1)$$

Where S represents the overall number of speech output, N represents the overall number of assessors, R_{ij} represents the overall number of evaluation outcomes analyzed by *i* assessor, and j^{th} speech signal.

We inferred 110 random samples of spoken sentences from different categories, including News, Sports, Linguistics, psychology, Poem, Health, Questions, Exclamation, Science, Everything, General Information, interviews, politics, Education & Literature, Story, Tourism, and SMS. These samples had not been trained to the model. Implementing HiFi-GAN[3] model that had been pre-trained to generate the subjective MOS for audio naturalness, 12 raters (seven males and five females, their ages from 21 to 46) rated each sample on a scale of 1 to 5 with a step of 0.5. Figure 6 illustrates the findings from genuine voice, 50,000 (100 epochs), 150,000 (300 epochs) and 250,000 (500 epochs) iterations. Furthermore, we compare audio synthesized by our method to genuine voice in a side-by-side comparison. Raters were requested to give a score from one (very bad) to five (very good).

Upon the evaluation, the results of this research varied significantly depending on the type of sentences we chose to test our model. For example, the model scored very high for the sound generation of linguistic sentences. On the other hand, the model scored lower for SMS, political, and poem sentences. The reason for the low MOS score for the poem is related to the prosody modeling which is important for the evaluators in the poem, but because the trained model takes most of its training data from declarative sentences, it also reads the poems as a declarative text and there is no good intonation in the generated voices for the poems. The possible reason for the lower MOS score for SMS samples is the presence of informal words and non-standard sentence structure, which is different from the sentence structure learned by the model. Also, the lower score for the political texts is probably due to the existence of specific words and proper nouns in this type of text.

Table 8 shows the MOS results from all evaluations. As it can be seen, increasing the number of epochs results in a higher MOS rate.

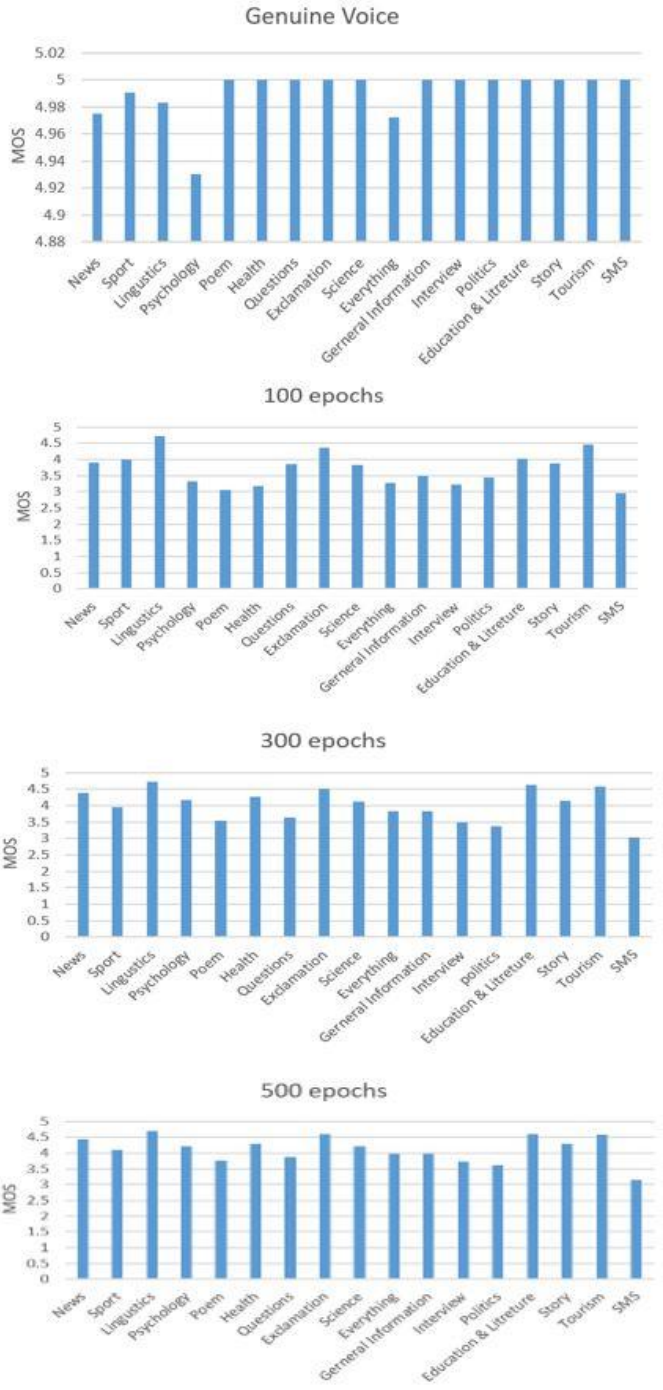

Figure 6: Results of the average of MOS in all different categories.

Table 8: The average MOS results of the proposed system.

	Result (MOS)
Genuine Voice	4.99
100 epochs	3.70
300 epochs	3.96
500 epochs	4.10

[3] https://colab.research.google.com/drive/1PZ4andZVFc8YALmhBbBB3AJghLN0r8zb#scrollTo=JEBBzewjaGWI

As there are no similar works in Kurdish to compare our results with them, in Table 9 our results are compared with the original Tacotron 2 result made by Google for English and the results of several other languages. Although the results cannot be directly compared together, however, they give an intuition about our result.

Table 9: A comparative MOS results.

Method and Language	Result (MOS)
Tacotron 2- English[8]	4.526
Tacotron 2- Arabic[14]	4.21
Tacotron 2- Persian[15]	3.97
Our proposed Tacotron 2- Kurdish	**4.10**

The results we found on the test set are very good in terms of quality, but there are problems in terms of understanding and pronunciation of some words and letters that we have presented in Table 10. The possible reasons for these problems are the lack of enough training data and the wrong mapping of using the transfer learning.

Table 10: A list of the words that the model generates wrong sounds.

Test set (sentences)	Word	Generated speech by the model
نەتەوە یەکگرتووەکان دەرفەتی کار دەداتە ئەنگێلا مێرکل.	ئەنگێلا	ئەنگێل
سەرمایەکی زۆر بەرێوەبە	سەرمایەکی	سەرمەیەکی
وەرە هەردووکمان نالّەمان یەکخەین بەڵکو بەو نیشتە نیشتمان سەرخەین	یەکخەین سەرخەین	یەککەین سەرکەین

4. Conclusion

In this research, we have utilized the Tacotron-2-based Central Kurdish TTS system. The Tacotron-2 model predicts the series of Mel-spectrogram frames from the input character sequences utilizing the pre-trained English model and overall of ten hours of recorded speeches, followed by HiFi-GAN-vocoder to synthesize high-quality Kurdish speech. Furthermore, the results have revealed that we have been able to attain satisfactory intelligibility and naturalness in the output speech. It also demonstrates that, despite the fact that both languages are significantly dissimilar in terms of character-level embedding and language phoneme, transfer learning from English TTS to Kurdish TTS can be performed effectively. It similarly explains how to pre-process audio speech training datasets in order to produce believable generated speech. On the other hand, using the English pre-trained model has some drawbacks for Kurdish letters, because both languages have different alphabet characters, sometimes the model will not pronounce some letters, including exchanging (k-ک) to (kh-خ), words like (دەکات) to (دەخات) as explained in Table 10. The researchers should employ a considerably bigger dataset to train the algorithm for future works, resulting in more convincing speech quality.

Conflict of interests

None

References

[1] S. Kayte, K. Waghmare, and B. Gawali, "Marathi Speech Synthesis: A review," *Int. J. Recent Innov. Trends Comput. Commun.*, vol. 3, no. 6, pp. 3708–3711, 2015.

[2] M. R. Mundada, B. Gawali, and S. Kayte, "Recognition and classification of speech and its related fluency disorders," *Int. J. Comput. Sci. Inf. Technol.*, vol. 5, no. 5, pp. 6764–6767, 2014.

[3] M. Elshafei, H. Al-Muhtaseb, and M. Al-Ghamdi, "Techniques for high quality Arabic speech synthesis," *Inf. Sci. (Ny).*, vol. 140, no. 3–4, pp. 255–267, 2002, doi: 10.1016/S0020-0255(01)00175-X.

[4] A. W. Black and P. Taylor, "Automatically Clustering Similar Units for Unit Selection Speech Synthesis," *Int. Speech Commun. Assoc.*, 1997.

[5] A. J. Hunt and A. W. Black, "Unit selection in a concatenative speech synthesis system using a large speech database," in *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, 1996, vol. 1, pp. 373–376, doi: 10.1109/icassp.1996.541110.

[6] H. Ze, A. Senior, and M. Schuster, "Statistical parametric speech synthesis using deep neural networks," in *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, 2013, pp. 7962–7966, doi: 10.1109/ICASSP.2013.6639215.

[7] H. Zen, K. Tokudaa, and A. W.Black, "Statistical parametric speech synthesis," *Speech Commun.*, vol. 51, no. 11, pp. 1039–1064, 2009, doi: https://doi.org/10.1016/j.specom.2009.04.004.

[8] Jonathan Shen, Ruoming Pang, Ron J. Weiss, Mike Schuster, Navdeep Jaitly, Zongheng Yang, Zhifeng Chen, Yu Zhang, Yuxuan Wang, RJ Skerry-Ryan, Rif A. Saurous, Yannis Agiomyrgiannakis, and Yonghui. Wu, "Natural TTS synthesis by conditioning WaveNet on mel spectrogram predictions," In Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), 2018.

[9] J. Sotelo, S. Mehri, K. Kumar, J. F. Santos, K. Kastner, A. Courville, and Y. Bengio. Char2wav: End-to-end speech synthesis. ICLR workshop, 2017.

[10] 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. arXiv preprint arXiv:1609.03499, 2016a.

[11] S. O. Arik, M. Chrzanowski, A. Coates, G. Diamos, A. Gib- ¨ iansky, Y. Kang, X. Li, J. Miller, J. Raiman, S. Sengupta, and M. Shoeybi, "Deep voice: Real-time neural text-to-speech," CoRR, vol. abs/1702.07825, 2017.

[12] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Advances in Neural Information Processing Systems*, 2014, vol. 4, no. January.

[13] Y. Lu, M. Dong, and Y. Chen, "Implementing Prosodic Phrasing in Chinese End-to-end Speech Synthesis," in *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, 2019, vol. 2019-May, pp. 7050–7054, doi: 10.1109/ICASSP.2019.8682368.

[14] F. K. Fahmy, M. I. Khalil, and H. M. Abbas, "A transfer learning end-to-end arabic text-to-speech (tts) deep architecture," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2020, vol. 12294 LNAI, doi: 10.1007/978-3-030-58309-5_22.

[15] N. Naderi, B. Nasersharif, and A. Nikoofard, "Persian speech synthesis using enhanced tacotron based on multi-resolution convolution layers and a convex optimization method," *Multimed. Tools Appl.*, vol. 81, no. 3, 2022, doi: 10.1007/s11042-021-11719-w.

[16] S. Lemmetty, "Review of speech synthesis technology," *Helsinki Univ. Technol.*, vol. 320, 1999.

[17] K. R. Aida-Zade, C. Ardil, and A. M. Sharifova, "The main principles of text-to-speech synthesis system," *World Acad. Sci. Eng. Technol.*, vol. 37, no. 1, pp. 13–19, 2010, doi: 10.5281/zenodo.1070639.

[18] N. K. Bakhsh and S. Alshomrani, "A Comparative Study of Arabic Text-to-Speech Synthesis Systems," *Int. J. Inf. Eng. Electron. Bus.*, vol. 6, no. 4, pp. 27–31, 2014, doi: 10.5815/ijieeb.2014.04.04.

[19] X. Huang, A. Acero, and H.-W. Hon, *Spoken Language Processing: A Guide to Theory, Algorithm & System Development*. 2001.

[20] B. J. Kröger, "Minimal rules for articulatory speech synthesis," *Theor. Appl. Amsterdam, Elsevier)*, pp. 331–334, 1992.

[21] D. H. Klatt, "Review of text-to-speech conversion for English," *J. Acoust. Soc. Am.*, vol. 82, no. 3, pp. 737–793, 1987, doi: 10.1121/1.395275.

[22] M. Z. Rashad, H. M. El-Bakry, I. R. Isma'il, and N. Mastorakis, "An overview of text-to-speech synthesis techniques," in *International Conference on Communications and Information Technology - Proceedings*, 2010, pp. 84–89.

[23] Y. Ning, S. He, Z. Wu, C. Xing, and L. J. Zhang, "Review of deep learning based speech synthesis," *Applied Sciences (Switzerland)*, vol. 9, no. 19. p. 4050, 2019, doi: 10.3390/app9194050.

[24] 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. arXiv preprint arXiv:1609.03499, 2016a.

[25] D. W. Griffin and J. S. Lim, "Signal Estimation from Modified Short-Time Fourier Transform," *IEEE Trans. Acoust.*, vol. 32, no. 2, pp. 236–243, 1984, doi: 10.1109/TASSP.1984.1164317.

[26] J. Kong, J. Kim, and J. Bae, "HiFi-GAN: Generative adversarial networks for efficient and high fidelity speech synthesis," in *Advances in Neural Information Processing Systems*, 2020, vol. 2020-December.

[27] S. Ioffe, C. S.-I. conference on machine, and undefined 2015, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *proceedings.mlr.press*, Accessed: Mar. 01, 2022. [Online]. Available: http://proceedings.mlr.press/v37/ioffe15.html.

[28] Y. Li and Y. Yuan, "Convergence analysis of two-layer neural networks with RELU activation," in *Advances in Neural Information Processing Systems*, 2017, vol. 2017-December.

[29] J. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho, and Y. Bengio, "Attention-based models for speech recognition," in *Advances in Neural Information Processing Systems*, 2015, vol. 2015-January.

[30] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, 2014.

[31] D. Krueger, T. Maharaj, J. Kramár, … M. P. preprint arXiv, and undefined 2016, "Zoneout: Regularizing rnns by randomly preserving hidden activations," *arxiv.org*, Accessed: Apr. 25, 2022. [Online]. Available: https://arxiv.org/abs/1606.01305.

[32] M. Cuong Nguyen Smartcall JSC, K. Duy Trieu Smartcall JSC, B. Quyen Dam Smartcall JSC, T. Phuong Nguyen, and Q. Bao Nguyen, "Development of Smartcall Vietnamese Text-to-Speech for VLSP 2020," *aclanthology.org*, Accessed: Mar. 01, 2022. [Online]. Available: https://aclanthology.org/2020.vlsp-1.5.pdf.

[33] W. Ping, K. Peng, and J. Chen, "Clarinet: Parallel wave generation in end-to-end text-to-speech," 2019.

[34] Y. Wang, R. Skerry-Ryan, D. Stanton, Y. Wu, R. J. Weiss, N. Jaitly, Z. Yang, Y. Xiao, Z. Chen, S. Bengio, Q. Le, Y. Agiomyrgiannakis, R. Clark, and R. A. Saurous. Tacotron: Towards end-to-end speech synthesis. In Interspeech, 2017.

[35] C. Zhang, S. Zhang, and H. Zhong, "A prosodic mandarin text-to-speech system based on tacotron," in *2019 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference, APSIPA ASC 2019*, 2019, pp. 165–169, doi: 10.1109/APSIPAASC47483.2019.9023283.

[36] R. Liu, B. Sisman, F. Bao, G. Gao, and H. Li, "Modeling Prosodic Phrasing with Multi-Task Learning in Tacotron-Based TTS," *IEEE Signal Process. Lett.*, vol. 27, pp. 1470–1474, 2020, doi: 10.1109/LSP.2020.3016564.

[37] R. Liu, B. Sisman, J. Li, F. Bao, G. Gao, and H. Li, "Teacher-student training for robust tacotron-based TTS," in *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, 2020, vol. 2020-May, pp. 6274–6278, doi: 10.1109/ICASSP40776.2020.9054681.

[38] T. He, W. Zhao, and L. Xu, "DOP-Tacotron: A Fast Chinese TTS System with Local-based Attention," in *Proceedings of the 32nd Chinese Control and Decision Conference, CCDC 2020*, 2020, pp. 4345–4350, doi: 10.1109/CCDC49329.2020.9164203.

[39] R. J. Weiss, R. Skerry-Ryan, E. Battenberg, S. Mariooryad, and D. P. Kingma, "Wave-Tacotron: Spectrogram-Free End-to-End Text-to-Speech Synthesis," 2021, pp. 5679–5683, doi: 10.1109/icassp39728.2021.9413851.

[40] Y. Win and T. Masada, "Myanmar Text-to-Speech System based on Tacotron-2," in *International Conference on ICT Convergence*, 2020, vol. 2020-October, doi: 10.1109/ICTC49870.2020.9289599.

[41] W. M. Thackston, "Sorani Kurdish—A Reference Grammar with Selected Readings," *Harvard Univ.*, 2006.

[42] T. J. Sejnowski and C. R. Rosenberg, "Parallel networks that learn to pronounce English text," *Complex Syst.*, vol. 1, 1987.

[43] A. Bahrampour, W. Barkhoda, and B. Z. Azami, "Implementation of three text to speech systems for Kurdish language," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2009, vol. 5856 LNCS, doi: 10.1007/978-3-642-10268-4_38.

[44] W. Barkhoda, B. ZahirAzami, A. Bahrampour, and O. K. Shahryari, "A comparison between allophone, syllable, and diphone based TTS systems for Kurdish language," 2009, doi: 10.1109/ISSPIT.2009.5407540.

[45] F. Daneshfar, W. Barkhoda, and B. Z. Azami, "Implementation of a text-to-speech system for Kurdish language," 2009, doi: 10.1109/ICDT.2009.29.

[46] H. Hassani, … R. K.-W. on I. of, and undefined 2011, "Kurdish text to speech (KTTS)," *researchgate.net*, 2011, Accessed: Mar. 01, 2022. [Online]. Available: https://www.researchgate.net/profile/Hossein-Hassani-2/publication/295092948_Kurdish_Text_to_Speech_KTTS/links/59c78058458515548f37944d/Kurdish-Text-to-Speech-KTTS.pdf.

[47] H. Veisi, H. Hosseini, … M. M. preprint arXiv, and undefined 2021, "Jira: a Kurdish Speech Recognition System Designing and Building Speech Corpus and Pronunciation Lexicon," *arxiv.org*, Accessed: Jun. 13, 2022. [Online]. Available: https://arxiv.org/abs/2102.07412.

[48] A. Mahmudi and H. Veisi, "Automated grapheme-to-phoneme conversion for Central Kurdish based on optimality theory," *Comput. Speech Lang.*, vol. 70, 2021, doi: 10.1016/j.csl.2021.101222.

[49] H. Veisi, M. MohammadAmini, and H. Hosseini, "Toward Kurdish language processing: Experiments in collecting and processing the AsoSoft text corpus," *Digit. Scholarsh. Humanit.*, 2019, doi: 10.1093/llc/fqy074.

[50] A. Debnath, S. S. Patil, G. Nadiger, and R. A. Ganesan, "Low-Resource End-to-end Sanskrit TTS using Tacotron2, WaveGlow and Transfer Learning," 2020, doi: 10.1109/INDICON49873.2020.9342071.

[51] D. T.-B. of E. E. and Informatics and undefined 2021, "The first FOSD-tacotron-2-based text-to-speech application for Vietnamese," *beei.org*, vol. 10, no. 2, pp. 898–903, 2021, doi: 10.11591/eei.v10i2.2539.