Recent Adv Sci Eng, Vol. 2, No. 1, pp. 1-5, June, 2022

Recent Advances in Science and Engineering
Web page info: https://rase.yildiz.edu.tr
DOI: 10.14744/rase.2022.0001

Research Article

Is it possible to train a Turkish text-to-speech model with English data?

Engin ERGÜN[1,*], Tülay YILDIRIM[2]

[1]*TÜBİTAK BİLGEM, Kocaeli, Turkey*
[2]*Yıldız Technical University, Faculty of Electrical and Electronics Engineering, İstanbul, Turkey*

ARTICLE INFO

Article history
Received: 13 February 2022
Accepted: 04 April 2022

Key words:
Natural language processing, nlp, speech processing, speech synthesis, text-to-speech, tts, artificial intelligence

ABSTRACT

Most natural language processing (NLP) studies need language-specific data for that language. Some languages like Turkish have scarce data sources to train successful deep learning models. Studies like speech synthesis require dozens of hours of professionally recorded speech with its correct transcription. Creating or finding datasets for text-to-speech (TTS) studies can be quite costly for both time and financial perspectives. This study tries to observe whether English acoustic data can be used to train a Turkish text-to-speech model to eliminate the data problem.

Cite this article as: Ergün E, Yıldırım T. Is it possible to train a Turkish text-to-speech model with English data? Recent Adv Sci Eng 2022;2:1:1–5.

INTRODUCTION

Interaction with machines becomes more interactive day by day. In the past, graphical interfaces were often preferred for entering commands to machines, but today more modern interaction methods can be preferred like voice control.

To control a machine by voice, it is necessary to teach it to interpret the speech signal first. Then, we have to give the speaking ability to the machine to get a voice response from the machine. Researchers working in the field of natural language processing carry out these human-specific behaviors (listening, understanding, writing, speaking) to machines.

Speech synthesis, i.e. the process of making machines speak, provides the artificial generation of a speech signal against text input. While these studies were initially done by adding phones to each other as a rule-based[1], they were later modeled statistically[2]. Today, speech synthesis studies can be done as naturally as human speech with deep neural network-based models trained with sufficient data. In addition to speech naturalness, another important issue in speech synthesis is to ensure the production of the sound on the CPU is synthesized very close to real-time. Because in many products, performing speech synthesis on the device is an important factor to reduce server costs.

In models developed with deep learning methods, the amount and quality of data have great importance. As stated in many publications, the performance of models developed with deep learning methods increases with the amount of data[3]. However, in some cases, finding or producing data can be quite costly. Especially in speech synthesis studies, there is a need for tens of hours of sound recordings and transcriptions of these recordings, which are language-specific and recorded in a studio environment without background noise. In some languages with data scarcity, it can be very difficult to carry out speech synthesis studies with deep learning methods. Turkish is one of the languages that is very difficult to find open data sources in this field. There are datasets created by volunteers vocalizing non-copyrighted books in languages such as English[4] [5], German[6] and Russian[7]. Since there is no such dataset for Turkish, it will be investigated whether the need for data can be eliminated by expressing the sources in other languages phonetically in Turkish.

***Corresponding author.**
*E-mail address: eengin.eergun@gmail.com

Published by Yıldız Technical University Press, İstanbul, Turkey

In this study, it will be discussed how to construct a Turkish speech synthesis model using acoustic data from an English dataset. For this, the LJSpeech dataset, which is frequently used in English speech synthesis studies, was used. This set contains approximately 24 hours of recordings of a female speaker. In this dataset, there are 13100 speech segments from 1 to 10 seconds in length. The texts corresponding to these segments will be phonetically translated into Turkish. Then the speech synthesis model will be trained with the updated dataset. In the sub-headings of the methodology section, it will be explained how to do these operations in order.

LITERATURE REVIEW

In the speech synthesis process, the model tries to obtain a complex and non-linear signal such as voice from highly compressed primitive data such as text. In this case, there is a lot of information gap between input and output. To reduce this information gap, two different models are generally used together. The first model, called the synthesizer, converts text to basic representations of sound, called mel-spectrograms, and the second model, called vocoder, generates sound signals in the time domain with mel-spectrograms.

Vocoders play a very important role in producing realistic sounds. In the early works, the Short Time Fourier Transform based Griffin-Lim[8] algorithm was used to obtain time domain signals from mel-spectrogram features. Robotic sound problems have been observed in the sounds synthesized with the Griffin-Lim algorithm. Since 2016, as a result of modeling vocoders with deep neural networks, more natural outputs have started to be obtained. With WaveNet[9], one of the pioneering works, artificial sounds that are almost indistinguishable from human speech began to be synthesized. Despite the naturalness of the synthesized sounds, it took a lot of time to produce the sound signals with this model. Due to its autoregressive nature, it also uses the calculations made in the previous stages in the next steps. Autoregressive models are very successful in producing natural sounds, but due to their inherent lack of parallelization, training and inference take a longer time than non-autoregressive models.

Autoregressive[9] and non-autoregressive[10-13] methods are available for the synthesizer and vocoder models. In current studies, non-autoregressive models can synthesize sounds with comparable quality with autoregressive models in real time on the CPU.

METHODOLOGY

In order to train speech synthesis models, sounds and phoneme sequences expressing sounds are needed. Since Turkish is a phonetic language, words are expressed as they are read. That is, character sequences are constructed words in Turkish. In non-phonetic languages such as English, words can be expressed with phonemes.

To synthesize Turkish speech with English data, the words in the English dataset first must be phonetically translated into Turkish. This situation can be likened to writing the words spoken in English as they are heard directly in Turkish. In order to express the phones in Turkish, the steps shown in **Figure 1** were followed.

In this study, CMUDict[14] and LJSpeech were used. CMUDict is a dictionary that phonetically expresses more than 120000 words in English. LJSpeech is a 24-hour dataset used in speech synthesis studies.

Phonetical Conversion

The CMUDict shown in **Figure 1** presents English words and their phonetic representations. The phones in this phonetic dictionary were replaced with the Turkish expressions of the English phones as in **Table 1**, and as a result, CMUDict' was created. For example, the word and phoneme sequence "ABANDONED [AH0 B AE1 N D AH0 N D]" in CMUDict is expressed as "ABANDONED [ɪ b n d ɪ n d]" in CMUDict'. **Table 1** was used directly while performing this conversion.

Training of G2P

CMUDict does not include all words in the LJSpeech dataset. It is necessary to obtain the Turkish phonetic representations of out-of-vocabulary words. To predict unknown words, a finite-state-transducer-based, n-gram Grapheme-to-Phoneme (G2P) model was trained with CMUDict'.

In the training phase, The Montreal Forced Aligner[16] tool was used. This tool implements the following steps during the training. Firstly, a unigram aligner is constructed with a finite state transducer using graphemes and phonemes, then to maximize probabilities, Viterbi training is used until the model convergence. After that, the best probability is computed with the Viterbi algorithm. Then, alignments are encoded with a finite-state acceptor. Therefore, each transaction matches grapheme and phoneme pairs. Then, using encoded alignments a higher-order n-gram model is constructed and then smoothed with Kneser-Ney[17] method[18]. In the decoding phase again

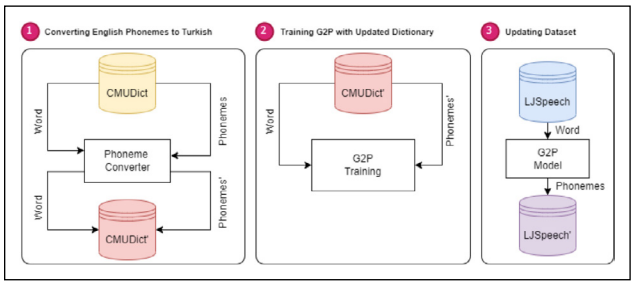

Figure 1. Phonetic Conversion Steps.

Table 1. English to Turkish Phonetic Conversion Table[15] (The phonemes shown with lowercase letters represent the Turkish pronunciation of the phonemes on the leftside.)

AA0	a	AW0	au	EH2	e	IH2	i	OW2	o	UH2	ü
AA1	a	AW1	au	ER0	ır	IY0	i	OY0	oy	UW0	u
AA2	a	AW2	au	ER1	ör	IY1	iy	OY1	oy	UW1	u
AE0	a	AY0	ay	ER2	ör	IY2	i	OY2	oy	UW2	u
AE1	e	AY1	ay	EY0	ey	JH	c	P	p	V	v
AE2	a	AY2	ay	EY1	ey	K	k	R	r	W	v
AH0	ı	B	b	EY2	ey	L	l	S	s	Y	y
AH1	o	CH	ç	F	f	M	m	SH	ş	Z	z
AH2	a	D	d	G	g	N	n	T	t	ZH	j
AO0	o	DH	d	HH	h	NG	n	TH	s		
AO1	o	EH0	e	IH0	i	OW0	o	UH0	u		
AO2	o	EH1	e	IH1	i	OW1	o	UH1	u		

Viterbi algorithm is used to find the most probable path for Turkish phoneme sequence against English word input.

In **Figure 2**, first and second lines show a sound signal and its transcription, respectively. Third line shows the G2P model's output against transcription input.

Training of Synthesizer

FastSpeech2[11] architecture, which can work in real time, is used as a synthesizer. It is difficult to synthesize realistic sound with direct text and voice pairs for this non-autoregressive network. For this, extra information is needed to establish a relationship between text and voice in the created network. In particular, the duration information to be obtained at the phone level is of great importance. The length of the audio signal to be synthesized is determined by the duration information. In order to extract the phone level duration, the audio signal must be aligned with the text. The result of the alignment process is shown in **Figure 2**. To do this, the Montreal Forced Aligner[16] tool was used.

In addition to the duration information, other features such as pitch and energy of the relevant audio segment are extracted in the preprocessing step. During the training, the weights of these features are also updated within the network. In the inference phase, only text input is used and finally, the mel-spectrogram is obtained.

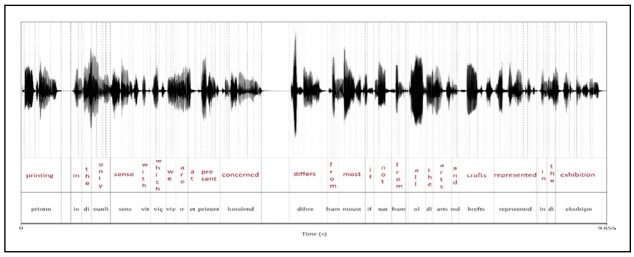

Figure 2. A segment from updated LJSpeech data set.

Training of Vocoder

Vocoders are trained with mel-spectrogram and audio signals in the time domain. The network tries to predict the audio signals against the mel-spectrogram input.

As a vocoder, Multiband-MelGAN[13] architecture, which is an optimized version of MelGAN[12] was preferred. Both of these networks use Generative Adversarial Network (GAN)[19] architecture as backbone. The generative network tries to synthesize sound with the mel-spectrogram input, while the discriminator network tries to understand whether the result obtained from the generative network is real or fake (synthesized).

Multiband-MelGAN architecture has basically three different advantages over MelGAN. First, the number of receptive fields in the MelGAN architecture was increased, multi-resolution STFT loss was used instead of feature matching loss in order to better distinguish between the synthesized voice and the target voice, and finally, the ability to operate in multi-band instead of single band was gained. This network attempts to estimate the mel-spectrogram input and the signals in the time domain for subbands in parallel, and combines the final results. Thus, the inference time has been accelerated compared to the classical MelGAN. A very fast synthesis process is achieved with a real-time factor of approximately 0.03 on the CPU[13].

The training of the vocoder model is completed by training the discriminator network until it cannot distinguish between the synthesized sound and the target sound.

To briefly summarize the operations performed, first of all, the LJSpeech dataset was updated by bringing it into Turkish phonetic order using the G2P model. Then Synthesizer and vocoder networks are trained. Now, all the requirements for speech synthesis are fulfilled. Mel-spectrograms are synthesized with the text given to the input of the synthesizer, then the outputs from this network are given directly to the vocoder and finally audio signals in the time domain are generated.

CONCLUSION AND FUTURE WORKS

Finding data for some tasks can be quite difficult. It is sometimes impossible to find publicly available data to use, especially in speech technologies. One of the biggest reasons for this is the law on the protection of personal data. There are also task specific challenging situations. For example, the data required for speech synthesis studies are usually created by professional sound artists in the studio environment. This affects both cost and availability of the data. Parallel to this situation, there is no suitable dataset to train or evaluate the Turkish TTS model. In this context, the main purpose of this work is to investigate whether the problem arising from the lack of data can be solved with the proposed method.

In this study, a Turkish TTS model was trained by converting the transcription of the English LJSpeech dataset into phonetically in Turkish. Although there are no "ı, ü, ş, ö, ç" characters in English, there are phones corresponding to these characters. The trained TTS model can generate phones corresponding to the "ı, ü, ş, ö, ç" characters. Because phonetic conversion produced some words that are included these characters. The only problem is that there is no phone that can correspond to "ğ".

As a result of the study, it has been proven that a Turkish speech synthesis model can be produced using the English dataset. Interestingly, the synthesized sounds resemble the speech of an English tourist trying to speak in Turkish. However, in practice, using this model directly may not be the right approach. Because, it is not easy to understand generated sentences due to its heavy accent. The resulting model can be fine-tune with a smaller Turkish dataset to break accent problem. In this way, more realistic results can be obtained.

Similar to the method used in phonetic conversion, it is considered to repeat the study in future by using International Phonetic Alphabet (IPA). Thus, it is foreseen that the phones will be better expressed and the British accent will be suppressed a little more.

AUTHORSHIP CONTRIBUTIONS

Authors equally contributed to this work.

DATA AVAILABILITY STATEMENT

The authors confirm that the data that supports the findings of this study are available within the article. Raw data that support the finding of this study are available from the corresponding author, upon reasonable request.

CONFLICT OF INTEREST

The author declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

ETHICS

There are no ethical issues with the publication of this manuscript.

REFERENCES

[1] L. Lee, C. Tseng, and C. Hsieh, "Improved tone concatenation rules in a formant-based Chinese text-to-speech system," IEEE Transactions on Speech and Audio Processing, Vol. 1(3), pp. 287–294, 1993. [CrossRef]

[2] A. Pradhan, A. Shanmugam, A. Prakash, K. Veezhinathan, and H. Murthy, "A syllable based statistical text to speech system," in 21st European Signal Processing Conference (EUSIPCO 2013), pp. 1–5, 2013.

[3] Z. Alom, T. M. Taha, C. Yakopcic, S. Westberg, P. Sidike, M. S. Nasrin, M. Hasan, B. C. Van Essen, A. A. S. Awwal, and V. K. Asari, "a state-of-the-art survey on deep learning theory and architectures," Electronics, Vol. 8, pp. 292, 2019. [CrossRef]

[4] H. Zen, V. Dang, R. Clark, Y. Zhang, R. J. Weiss, Y. Jia, Z. Chen, and Y. Wu, "LibriTTS: a corpus derived from librispeech for text-to-speech," ArXiv190402882 Cs Eess, 2019, https://doi.org/10.48550/arXiv.1904.02882 Accessed Jan 25, 2022. [CrossRef]

[5] K. Ito and L. Johnson, "The LJ speech dataset," 2017. Available: https://keithito.com/LJ-Speech-Dataset/ Accessed Apr 14, 2022.

[6] T. Müller and D. Kreutz, "Thorsten - open german voice (neutral) dataset," Zenodo, 2021. https://github.com/thorstenMueller/Thorsten-Voice Accessed Apr 14, 2022.

[7] "Russian LibriSpeech dataset," https://www.openslr.org/96/ Accessed Jan 25, 2022.

[8] D. Griffin and J. Lim, "Signal estimation from modified short-time Fourier transform," IEEE Transactions on Acoustics, Speech, and Signal Processing, Vol. 32(2), pp. 236–243, 1984. [CrossRef]

[9] 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," ArXiv160903499 Cs Eess, 2016. http://arxiv.org/abs/1609.03499 Accessed Jan 25, 2022.

[10] Y. Ren, Y. Ruan, X. Tan, T. Qin, S. Zhao, Z. Zhao, and T. Y. Liu. "FastSpeech: Fast, robust and controllable text to speech," ArXiv190509263 Cs Eess, 2019. https://doi.org/10.48550/arXiv.1905.09263 Accessed Jan 25, 2022.

[11] Y. Ren, C. Hu, T. Qin, S. Zhao, Z. Zhao, and T. Y. Liu, "FastSpeech 2: Fast and high-quality end-to-end text to speech," ArXiv200604558 Cs Eess, 2021. Accessed Jan 25, 2022. http://arxiv.org/abs/2006.04558

[12] K. Kumar, R. Kumar, T. de Boissiere, L. Gestin, W. Zhen, Teoh, J. Sotelo, A. de Brebisson, Y. Bengio,

and A. Courville, "MelGAN: Generative adversarial networks for conditional waveform synthesis," ArXiv191006711 Cs Eess, 2019. http://arxiv.org/abs/1910.06711 Accessed Jan 25, 2022.

[13] G. Yang, S. Yang, K. Liu, P. Fang, W. Chen, and L. Xie, "Multi-band MelGAN: Faster waveform generation for high-quality text-to-speech," ArXiv200505106 Cs Eess, 2020. http://arxiv.org/abs/2005.05106 Accessed Jan. 25, 2022.

[14] "CMUDict," http://svn.code.sf.net/p/cmusphinx/code/trunk/cmudict/cmudict.0.6d Accessed Jan 25, 2022.

[15] A. A. Akın, "Zemberek-NLP" 2022. https://github.com/ahmetaa/zemberek-nlp/blob/a9c0f88210dd6a4a1b6152de88d117054a105879/morphology/src/main/resources/tr/phonetics/english-phones-to-turkish.txt Accessed: Jan 25, 2022.

[16] M. McAuliffe, M. Socolof, S. Mihuc, M. Wagner, and M. Sonderegger, "Montreal forced aligner: Trainable text-speech alignment using Kaldi," Interspeech, pp. 498–502, 2017. [CrossRef]

[17] H. Ney, U. Essen, and R. Kneser, "On structuring probabilistic dependences in stochastic language modelling," Computer Speech and Language, Vol. 8(1), pp. 1–38, 1994. [CrossRef]

[18] J. L. Lee, L. F. E. Ashby, M. E. Garza, Y. Lee-Sikka, S. Miller, A. Wong, A. D. McCarthy, and K. Gorman, "Massively multilingual pronunciation modeling with WikiPron," in Proceedings of the 12th Language Resources and Evaluation Conference, Marseille, France, pp. 4223–4228, 2020. https://aclanthology.org/2020.lrec-1.521 Accessed Jan 30, 2022

[19] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Falsey, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," ArXiv14062661 Cs Stat, 2014. http://arxiv.org/abs/1406.2661 Accessed Jan 25, 2022.