Text-to-Speech in Azerbaijani Language via Transfer Learning in a Low Resource Environment

Dzhavidan Zeinalov[1], **Bugra Sen**[2], **Firangiz Aslanova**[2]
[1]Digital Research Lab, Baku, Azerbaijan
[2]Kapital Bank, Baku, Azerbaijan
javidan.zeynalov@researchlab.digital
{bugra.sen, firangiz.aslanova}@kapitalbank.az

Abstract

Most Text-to-Speech models cannot operate well in low-resource languages, and require a great amount of high-quality training data to be considered good enough. Yet, with the improvements made in ASR systems, it is now much easier than ever to collect data for the design of custom Text-to-Speech models. In this paper, our work on using ASR model to collect data to build a viable Text-to-Speech system for one of the leading financial institutions of Azerbaijan will be outlined. NVIDIA's implementation of the Tacotron 2 model was utilized along with the HiFiGAN vocoder. As for the training, the model was first trained with high-quality audio data collected from the Internet, then fine-tuned on the bank's single-speaker Call Center data. The results were then evaluated by 50 different listeners and got a Mean Opinion Score of 4.17, displaying that our method is indeed viable. With this, we have successfully designed the first Text-to-Speech model in Azerbaijani, and publicly shared 12 hours of audiobook data for everyone to use.

1 Introduction

Text-to-speech systems are generally made up of two parts to gain more control over the whole process: mel – spectrogram generator, which learns based on our labeled audio data to synthesize mel – spectrogram of input text, and a neural vocoder which is essentially what turns mel – spectrograms into a waveform (Shen et al., 2018). One of the most successful TTS systems, namely Tacotron 2, was released in 2016, with a performance that rivals that of professional speakers. Tacotron 2 is a system that first maps character embeddings to mel-scale spectrograms, and then utilizes a vocoder to generate audio waveforms from the spectrograms (Shen et al., 2018). By using the LJSpeech dataset that contains a single speaker data of around 24 hours, Tacotron 2 achieved an incredible performance of 4.53 MOS, almost the same score that would be given to a recording of a professional voice actor (Ito and Johnson, 2017). While TTS systems for popular languages such as English have existed for quite some time, many low–resource languages struggle in this regard. One interesting recent development is the utilization of Speech Recognition models to collect data. One of the most popular models, Whisper, which is an open-source speech recognition model released in 2022, performs extremely well in numerous languages (Radford et al., 2022). Its largest version currently supports over 100 languages, and it can be run in a Google Colab environment, making it quite accessible to users. Azerbaijani language, also known as Azeri, is a Turkic language spoken primarily in Azerbaijan. It is also spoken by many across other countries, mainly Turkey, Iran, Georgia, and Russia. There is currently little work being done about the data collection of Azerbaijani speech, making the development of Speech models from scratch impossible. Even Whisper's largest version offers only around 24.8 WER percentage, which does not make it into the top 30 (Radford et al., 2022). One interesting work we have come accross is (Kamil Aida-Zade, 2010) , which uses a simple TTS architecture. However, it is not up to date as the paper has been around for years, and there are many new TTS models that would outperform the rather statistical and probabilistic approach used by them.

In this work, our task is to develop a TTS solution for one of the leading banks of Azerbaijan, with naturalness being our chief goal. We also show the effectiveness of using 2 pre-trained models for the complete training to overcome Tacotron 2's need for large amounts of data.

2 Approach

We developed Azerbaijani TTS by using Tacotron 2 architecture together with the HiFiGAN vocoder model. The HiFiGAN model was not trained by us, and instead, the universal version – which was trained on the LJSpeech dataset – was used for inference (Kong et al., 2020).

2.1 TTS Model Architecture

The reason for choosing Tacotron 2 model was simple: our model will later be used for a plethora of possible utilizations, in each of which the voice naturalness, rather than inference speed, is crucial. Our choice of implementation was that of NVIDIA's, which is PyTorch implementation of Tacotron 2, providing faster-than-realtime inference as a nice bonus. The Tacotron 2 model itself has an encoder-decoder architecture, with location-sensitive attention being utilized. Audio data, accompanied by its transcriptions, is needed for the training of Tacotron 2. In the overall flow of the said model, text data first follows a few steps of preprocessing, namely normalization, removal of punctuations, and conversion of numerics into words. The audio data experiences the same, as the Tacotron 2 model is coded to handle audio with specific parameters, namely 22,05 kHz, 16-bit format, mono encoding, and wav file type (Shen et al., 2018). The above preprocessing was done for all the data utilized in our work before doing any training. The encoder part is responsible for turning text – a sequence of characters, to be precise – into embeddings. This enables our models to understand our data, as text data by itself cannot be processed, and embedding gives our model the semantic meaning of the text data. Then, 3 Convolutional Layers, followed by a Bidirectional LSTM layer capture the long-term dependencies within the text. This step allows our model to extract features that will later be relevant to the mel -spectrogram generation. The attention mechanism utilized in the implementation is location-sensitive attention. The mechanism allows the model to virtually direct its "attention" on the text sequence's specific parts when doing the predictions of mel – spectrogram frames (Zhang et al., 2021). There are also pre–net, and post–net layers, which are responsible for enhancing feature extraction from text and quality of synthesized mel – spectrograms respectively. Last but not least, the generated mel – spectrogram is then provided as the input to the HiFiGAN model, which generates the audio waveforms. Little to no changes were made to the model architectures for both Tacotron 2 and HiFiGAN, as they both demonstrate outstanding results on their own (Shen et al., 2018). Figure 1 display the overall flow of our development.

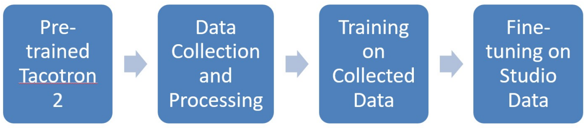
Pre-trained Tacotron 2
Data Collection and Processing
Training on Collected Data
Fine-tuning on Studio Data

Figure 1: Overall Process Flow of Azerbaijani TTS Development.

The parts that were changed are as follows:

- The letters variable in the symbols.py was changed to accommodate the Azerbaijani alphabet.
- The chosen cleaner was changed to a basic cleaner and adapted to the Azerbaijani language – specifically, handling abbreviations and numbers.
- Hyperparameters were changed as both the amount and type of data differ from the original implementation.

2.2 Data Collection and Preprocessing

In this sub-section, the data utilized, its collection as well as preprocessing will be outlined. Our first thoughts were to utilize available datasets such as Common Voice by Mozilla Foundation, or FLEURS (Ardila et al., 2020; Conneau et al., 2022). However, the data quality across many audios was too low, and as the location-sensitive attention is sensitive to the quality of training data, the idea was rejected (Zhang et al., 2021). The financial institution provided us with audio recordings that are currently utilized in its Call Center. The audios were of studio quality. In our experiments, we found the data to be not enough to capture many phonetic features of the language and therefore collected additional data. We found open-sourced audiobook recordings, which totaled 11 hours. The recordings were high quality but did not have any transcriptions, as the audiobook was based on a really old PDF edition. Hence why the transcriptions of the audio recordings for both studio data and audiobook were obtained by using OpenAI's Whisper model's large version 3 (Radford et al., 2022). By changing decoding options as well as making use of a Voice Activity Detection filter, namely Silero - VAD, we achieved accurate timestamping of the recording along with its corresponding transcriptions (Team, 2021). The decoding parameters that were changed are outlined in Table 1. As the dataset utilized for the original Tacotron 2 training was between the length of 2 to 20 seconds, we followed the same rule when segmenting our data (Shen et al., 2018). In case any audio segments were longer than the aforementioned value, it was split into parts manually, as there was a limited number of them after splitting via Whisper model (audiobook).

Table 1: Decoding Parameters of Whisper Model

Parameter Name	Value chosen
Beam size	5
Best of	5
Temperature	$(0.0, 0.2, 0.4, 0.6, 0.8, 1.0)$
Vad filter	silero:v3.1

The information regarding our data is provided as follows in Table 2.

Table 2: Audio Source, Amount, and the corresponding quality

Source	Amount(hr)	Quality	Segments
Call Center	0.76	Studio Level	532
Audiobook	11.3	High	7723

3 Experiments

In all our experiments, we divided 95 percent of our data to be training set, and the rest to be validation set. At first, only 46 minutes of Call Center audio data was available. We first conducted a fast trial by excluding any audio that was longer than 12 seconds. This left us with a total of 30 minutes of studio-quality data. The results seemed to have overfit, as some letters that were present in our training set would sometimes be mispronounced, or skipped entirely – this model will be referred to as the First Model. For this reason, we manually split the rest of the data, giving us a total of 46 minutes of audio data entirely – this model will be referred to as the Second Model. This time, some degree of hyperparameter tuning was also conducted to see the effects of longer training, decay rate, different learning rates as well as batch sizes. This second model generated intelligible results, especially in cases when the text to be synthesized contained words close to our training data – bank terminology. That said, it lacked generalization, which was a crucial aspect. Hence, the search for more data began, and we later found an audiobook of about 11 hours of data. It was split into 46 parts, each being read by the same Female speaker with a quality that was considered good enough. As we already know using a pre-trained model, even if in a different language, will still produce better results, it was decided to train the model beginning from the English language checkpoint using audiobook data (Pine et al., 2022; Byambadorj et al., 2021). Then, we would fine-tune the model with our Call Center data, not only introducing audio recordings of a higher quality but also the terminology related to finance. After training the model on audiobook for around 150,000 iterations, the results were already amazing as the model could generalize as well as produce intelligible results entirely – this model will be referred to as the Third and the Final Model. The hyperparameters we chose for this were based on the original implementation, as the model might have overfitted if we used the parameters as before (Shen et al., 2018). We stopped the training at 500 epochs and used our 46-minute Call Center data to further fine-tune it for another 300 epochs entirely – this model will be referred to as the Final Model (Byambadorj et al., 2021). The hyperparameters and additional information regarding different models are provided in Table 3.

Table 3: Parameter and data changes across models

Parameters	1st Model	2nd Model	Final Model
Epochs	250	500	500
Learning Rate	$1e^{-4}$	$5e^{-4}$	$1e^{-3}$
Weight Decay	0	$1e^{-6}$	$1e^{-6}$
Beta 1	0.99	0.99	0.99
Beta 2	0.999	0.999	0.999
Batch Size	8	16	16

4 Results

The evaluation of TTS systems is still a challenge, as there is not one metric that is universally accepted. In the case of speech recognition, there are 2 prominent methods, namely Word Error Rate (WER) and Character Error Rate (CER) (Wang et al., 2003). For TTS, the only viable metric is the Mean Opinion Score (MOS) (Viswanathan and Viswanathan, 2005). To evaluate our models, we generated a total of 100 sentences, 70 sentences similar to our training data, and 30 sentences completely new. The reason for such distribution was due to the core reason for TTS development, which was to be utilized in the banking sector. 10 independent subjects rated the model samples across 5 metrics such as naturalness, overall quality, prosody, pronunciation, and intelligibility. The subjects are all native listeners, and they have all been informed about the MOS metric and how it is used to evaluate the performance of TTS models. Despite not being experts on financial domain specifically, we believe their knowledge of the language is still enough, as the model does not dive too deep into financial terms, and generates sentences known by most speakers. Then, the average score given by each subject per sample was summed up, and divided by the number of participants to evaluate a model. The sentences were unforeseen in our training data, and the overlapping words were kept to a minimum. The scores received by the trained models are given in Table 4. However, only the Second and Final Models were evaluated due to the scarcity of time and resources for evaluators. For the convenience, Second Model will be denoted with number 2, and Final Model will be denoted with number 3. To increase the readability of the table, the following abbreviations are utilized:

- Intelligibility – I.
- Naturalness – N.
- Prosody – Py.
- Quality – Q.
- Pronunciation – Pn.
- Average – Avg.

Table 4: Mean Opinion Score for each Model

Model	N	Q	I	Pn	Py
2	$2,45$	$2,71$	$2,42$	$2,22$	$2,32$
3	$4,12$	$4,3$	$3,98$	$3,92$	$4,23$

Our Final Model received a Mean Opinion Score of 4.17, with a confidence score of ± 0.4, rivaling some high-resource languages.

5 Discussion

Our results show that currently, even for a language that ranks 39th on the WER evaluation of the Whisper model, it is possible to collect enough data for the design and training of a high-quality TTS system (Radford et al., 2022). Tacotron 2 architecture, despite being sensitive to data quality, is more than capable of utilizing transfer learning in the same language for a different speaker to provide a high-quality mel – spectrogram generation and the HiFiGAN model does not necessarily need to be fine-tuned for effective voice synthesis (Kong et al., 2020; Pine et al., 2022).

6 Conclusion

In this paper, we outline the works done to develop a Text-to-Speech System for the Azerbaijani language for one of the leading financial institutions of the said country. The problem of not having enough data was overcome by the collection of high-quality data from the Internet, and some hyperparameter tuning as well as additional tests were carried out to see the impact on convergence and model performance. With even further development of ASR systems, it will soon be possible to train TTS models for languages that are low-resourced. Additionally, we would like to next time set up a phoneme-dictionary-based training, which is said to improve convergence speed even further.

Limitations

While we do believe the work we have done could be helpful to others who are also trying to use ASR models for data collection purposes for Text-to-Speech applications, there is a limitation to this. We have noticed that the quality of audio largely depends on the speaker's prosody as well as the quality of audio. That is to say, to get a natural voice, it is also needed that the data collected is not monotonous, but rather rich in sounds. The Audiobook-only model that we tried was average in quality, but there was a huge difference between this model and the model that was trained on top of the audiobook with only 46 minutes of high-quality data. It is possible that a model that was trained with only 2 hours of studio quality data could surpass that of 15 hours of average quality data. In summary, if there is no high-quality data available on the Internet, the quality of TTS model might still be lacking, even if manual corrections are made to the labels.

Ethics Statement

While we are excited with the improvements made in ASR technology fields, it is crucial that the data collected is done with consent, or with data that is openly sourced. We have obtained our data from a local public library that belongs to the government with their consent. Collection of audio data and building a TTS model on someone's voice without their knowledge or consent is something we discourage strongly. With power, comes great responsibility.

Acknowledgment

The authors would like to thank the financial institution for being generous and sharing its data, as well as all that had parts in the design of the said Text-to-Speech model, no matter how small of a contribution it was.

References

Rosana Ardila, Megan Branson, Kelly Davis, Michael Henretty, Michael Kohler, Josh Meyer, Reuben Morais, Lindsay Saunders, Francis M. Tyers, and Gregor Weber. 2020. Common voice: A massively-multilingual speech corpus.

audiobook. Audiobook dataset. [link].

Zolzaya Byambadorj, Ryota Nishimura, Altangerel Ayush, Kengo Ohta, and Norihide Kitaoka. 2021. Multi-speaker tts system for low-resource language using cross-lingual transfer learning and data augmentation. In *2021 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, pages 849–853.

Alexis Conneau, Min Ma, Simran Khanuja, Yu Zhang, Vera Axelrod, Siddharth Dalmia, Jason Riesa, Clara Rivera, and Ankur Bapna. 2022. Fleurs: Few-shot learning evaluation of universal representations of speech.

Keith Ito and Linda Johnson. 2017. The lj speech dataset. https://keithito.com/LJ-Speech-Dataset/.

Aida Sharifova Kamil Aida-Zade. 2010. Azerbaijan text-to-speech synthesis system. In *The Third International Conference "Problems of Cybernetics and Informatics"*, pages 33–40.

Jungil Kong, Jaehyeon Kim, and Jaekyoung Bae. 2020. Hifi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis.

Aidan Pine, Dan Wells, Nathan Brinklow, Patrick Littell, and Korin Richmond. 2022. Requirements and motivations of low-resource speech synthesis for language revitalization. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7346–7359, Dublin, Ireland. Association for Computational Linguistics.

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. 2022. Robust speech recognition via large-scale weak supervision.

Jonathan Shen, Ruoming Pang, Ron J. Weiss, Mike Schuster, Navdeep Jaitly, Zongheng Yang, Zhifeng Chen, Yu Zhang, Yuxuan Wang, Rj Skerrv-Ryan, Rif A. Saurous, Yannis Agiomvrgiannakis, and Yonghui Wu. 2018. Natural tts synthesis by conditioning wavenet on mel spectrogram predictions. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4779–4783.

Silero Team. 2021. Silero vad: pre-trained enterprise-grade voice activity detector (vad), number detector and language classifier. https://github.com/snakers4/silero-vad.

Mahesh Viswanathan and Madhubalan Viswanathan. 2005. Measuring speech quality for text-to-speech systems: development and assessment of a modified mean opinion score (mos) scale. *Computer Speech Language*, 19(1):55–83.

Ye-Yi Wang, A. Acero, and C. Chelba. 2003. Is word error rate a good indicator for spoken language understanding accuracy. In *2003 IEEE Workshop on Automatic Speech Recognition and Understanding (IEEE Cat. No.03EX721)*, pages 577–582.

Xiangzhou Zhang, Bingpeng Ma, Hong Chang, Shiguang Shan, and Xilin Chen. 2021. Location sensitive network for human instance segmentation. *IEEE Transactions on Image Processing*, 30:7649–7662.