VNU Journal of Science: Comp. Science & Com. Eng, Vol. 38, No. 1 (2022) 61-67

VNU Journal of Science:
Computer Science and Communication Engineering

Journal homepage: http://www.jcsce.vnu.edu.vn/index.php/jcsce

Original Article

TTS - VLSP 2021: The Thunder Text-To-Speech System

Nguyen Thi Ngoc Anh[1,2,*], Nguyen Tien Thanh[1], Le Dang Linh[1]

[1]*Viettel Cyberspace Center, Viettel Group, Cau Giay, Hanoi, Vietnam*
[2]*Hanoi University of Science and Technology,Dai Co Viet, Hai Ba Trung, Hanoi, Vietnam*

Received 27 December 2021
Revised 18 April 2022; Accepted 5 May 2022

Abstract: This paper describes our speech synthesis system participating in the Vietnamese Text-To-Speech track of the 2021 VLSP evaluation campaign. The goal of this challenge is to build a synthetic voice from a provided spontaneous speech corpus in Vietnamese. In this paper, we propose our implementation of FastSpeech2 model on spontaneous speech. We used a special strategy with spontaneous datasets using the TTS system. We present our utilization in generating mel-spectrograms from given texts and then synthesize speech from generated mel-spectrograms using a separately trained vocoder. In evaluation, our team achieved 3.943 mean score in MOS in-domain test, 3.3 in MOS out-domain test, and 85.00% SUS, which indicates the effectiveness of the proposed system.

Keywords: Text-to-speech, Spontaneous, Vietnamese, FastSpeech 2, Hifi-GAN.

1. Introduction

The field of speech synthesis is expanding, from reading news, virtual assistants, customer care to voiceover, simple communication dialogue. Human's speech is spontaneous and conversational, so building a natural speech synthesis system is essential. Spontaneous speech is especially important if you use text-to-speech (TTS) in natural conversations instead of just using TTS for readable text. With the training data being spontaneous speech, the quality of synthesized speech will be more natural in human-machine communication applications.

In recent years, speech synthesized systems have significantly improved with good quality in synthesized voice. Deep learning-based models need to be trained with a large number of high-quality (text, speech) data pairs to synthesize high-fidelity speech, and the dataset requirements are higher when synthesizing speech with specific prosody and emotion[1]. To our knowledge, not many studies achieve good results with spontaneous speech such as conversation, talk shows, or podcasts. Building a spontaneous speech training dataset in ideal studio conditions is not easy, expensive, and time-consuming when the speakers do not have a prepared script. Therefore, it is essential to

* Corresponding author.
E-mail address: ngocanh2162@gmail.com

https://doi.org/10.25073/2588-1086/vnucsce.342

exploit the available spontaneous speech data sources.

The VLSP Speech Synthesis Challenge 2021 [2] is focused on building Vietnamese spontaneous speech synthesizers provided speech data. Before receiving the dataset, participants must join to contribute to building it. Data announced this year's contest had exploited the voice source from a Hanoi female YouTubers channel "Giang Oi". Despite the fact that the data has been crawled and cleaned, the main challenges of the TTS task in this year's competition are as follows: Noises in the background, occasionally intermingled with other voices, changes in intensity, stress, and prosody across the dataset, and inexact transcripts (although validated by human).

Because of the random and separate properties of spontaneous speech, applying to adapt reading-style speech for synthesizing spontaneous speech doesn't improve natural quality. Other strategies, including copying spontaneous phenomena from natural speech (preparing some types in the data set), inserting them into the script in the reading-style record [3], or using language models, are used in spontaneous speech versions [4]. However, it is not evident how to harness the beneficial effects of spontaneous speech phenomena in TTS systems. Some research on Vietnamese TTS, such as using a prosodic boundary prediction model for improving the naturalness of speech synthesis [5], normalizing written text often found on newspapers to its spoken form [6].

In this work, we discuss the all procedures of developing our end-to-end TTS system with data preprocessing. Then we show how this combination has improved the quality of our synthesized system. This paper is organized as follows: Section 2 briefly describes data preparation. Section 3 shows the components of the proposed model and experimental setup. Results and evaluation are analyzedin Section 4. Finally, we conclude our paper in Section 5.

2. Data Processing

The dataset was used in this paper was provided by the TTS evaluation of the VLSP 2021[2]. The dataset contains 5341 utterances from a single speaker (approximately 7.23 hours) with the corresponding text; the sampling rate is 44.1 kHz, single/mono audio channel. To ensure the quality of training data, we have automatically preprocessed and augmented data Figure 1.

2.1. Audio Preprocessing

First, we use a pretrained Speech Enhancement model to remove background music, make the audio waveform clear. Second, we use a pre-trained Speaker Verification system [7] to identify the main voice. All utterances with a Similarity Score below the threshold will be eliminated as they contain the other voices. Next, the utterances or sections containing no useful information in the audio data are removed. At the same time, with waves containing filled pauses, we replace silence with the same length. To handle volume differences between distinct utterances, we normalize the peak amplitude of all audio clips to a level of $1 - 3$ dB. Noise in silent parts is removed to achieve rhythm uniformity, and silence portions at the start and finish of utterances are adjusted to 0.2 seconds. In addition, the removal has resulted in a considerable increase in the number of utterances with a length of less than 2 seconds. Therefore, short utterances were randomly concatenated into longer utterances that lasted about $4 - 10$ seconds. To maintain naturalness, we only match utterances with similar rhythms (pitch, loudness), and we make sure the interrupted end-of-the-bound break section does not exceed 0.15 seconds. The longer utterances than 15 seconds were cut at the pause point. This ensures the Gaussian distribution of the data duration.

Our experiments found many long minutes of silence in audio, which do not have corresponding prosodic punctuation. This leads to a negative impact on TTS performance. In order to address this problem, we insert

punctuations to the transcript at positions of corresponding silences. Based on aligned time-stamps provided by the ASR system, we calculate the duration of internal silence and then insert the prosodic punctuation into the transcript at corresponding silences' positions. We simply choose the silence duration threshold to insert punctuation to be 0.2 seconds[8]. Thus, we can prevent the models from wrong alignments in pause frames.

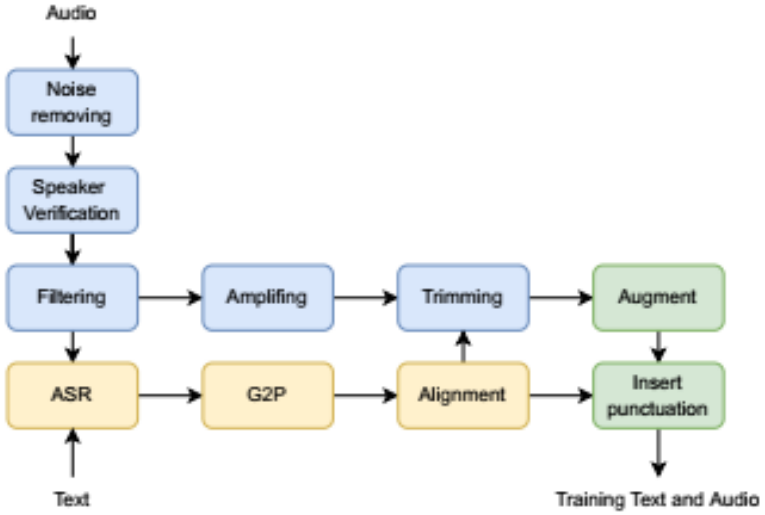

Figure 1. Data Processing

2.2. Transcript Preprocessing

We use a pre-trained ASR system [9] to decode the content of audio and then compare it to the original text. The sentences having WER higher than 10% are removed. All sentences are converted to lower-case ASCII characters and then transformed into phonemes sequences using a phonetic dictionary for 6, 700 Vietnamese widely-used syllables. With foreign words that appear in the dataset that are not in the dictionary, we utilize the phonemizer toolkit to convert words to IPA phoneme sequences. The IPA phoneme sequences are then mapped to Vietnamese phoneme sequences following a set of rules.

Finally, the dataset contains 7743 utterances across about 9.67 hours. 7643 utterances are used for training, 50 utterances are used for validation, and 50 utterances are used for testing.

3. System Architecture

The application of the end-to-end speech synthesis technology has yielded some outstanding results in recent years. One of the most advanced models - FastSpeech

[10] - has been used in our prior studies [11], and the potential of this model has been acknowledged. With the advantages of FastSpeech, we experiment with the new version, which is FastSpeech 2 to build the system. The overall model architecture of the system is shown in Figure 2.

3.1. Model

Our end-to-end TTS system has two major components: 1) an acoustic model which generates mel-spectrograms from a sequence of phonemes as input, and 2) the vocoder model which generates time-domain waveform samples conditioned on the predicted mel spectrogram frames. In the paper, we utilized and modified FastSpeech 2 [12] for acoustic modeling and Hifi-GAN vocoder [13].

Fast-Speech 2: Fast-Speech 2 addresses the issues in FastSpeech [10] and better solves the one-to-many mapping problem in TTS. Our network architecture is based on FastSpeech 2's original architecture, with a few modifications:

First, we apply an additional Post-Net layer (consistent with Tacotron2 [14]) to generate a new mel-spectrogram. The output of Post-Net layer is added to mel-spectrograms to generate the final mel-spectrograms. In our experiments, Post-Net layer significantly improves predicted mel-spectrograms quality. Second, we remove the pitch predictor and energy predictor modules. With data processing including amplifying pitch and volume, using this layer not only does not improve quality but also increases the model size.

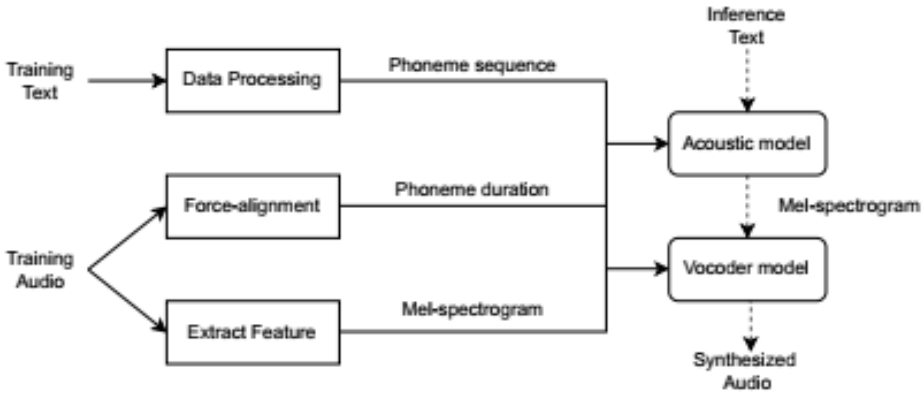

Figure 2. The proposed system.

Hifi-GAN: To generate high-quality speech from mel-spectrogram, we utilize Hifi-GAN model [13]. Hifi-GAN model consists of one generator and two discriminators: multi-scale and multi-period discriminators. The generator and discriminators have trained adversarially, along with two additional losses for improving training stability and model performance. The Hifi-GAN also enables quicker synthesis. To reduce noise of speech synthesis, we utilize a Denoiser module of Waveglow model [15].

3.2. Experimental Setup

Parameters of the acoustic model are primarily based on FastSpeech 2 [12]. Model consists of 4 feed-forward Transformer (FFT) blocks in the encoder and the mel-spectrogram decoder. In each FFT block, the dimension of phoneme embeddings and the hidden size of the self-attention are set to 256. The number of attention heads is set to 2 and the kernel sizes of the 1D-convolution in the 2-layer convolutional network after the self-attention layer are set to 9 and 1, with input/output size of 256/1024 for the first layer and 1024/256 in the second layer. In the variance predictor, the kernel sizes of the 1D-convolution are set to 3, with input/output sizes of 256/256 for both layers. We use dropout rate of 0.1 in every dropout layer, including Dropout on attention heads.

To improve the overall reconstructed mel-spectrogram, we leverage the Post-Net layer described in Tacotron2 [14] paper. The predicted mel-pectrogram is passed through a 5-layer convolutional Post-Net. Each Post-Net layer is comprised of 512 filters with shape 5×1 with batch normalization, followed by tanh activations on all but the final layer. We minimize the summed mean squared error (MSE) from both before and after the Post-Net to aid convergence and the duration predictor layer.

The parameters of Hifi-GAN model [13] mostly follow version 1 of the original paper to archive the best quality of speech synthesis.

We use the Kaldi-based ASR model for forced alignment. FastSpeech 2 and Hifi-GAN models are trained on 4 NVIDIA A100 GPUs with automatic mixed precision. Model FastSpeech 2 is trained up to 8, 000 epochs using Adam optimizer with a learning rate of 10^{-3}. Learning rate is increased for a warm-up period of 100, 000 steps and then decayed according to the Transformer schedule. We train model FastSpeech 2 with a batch size of 256. Due to the limited VRAM of the GPU, we utilize the

gradient accumulation trick to gain that batch size. The Hifi-GAN vocoder is trained up to 1, 000, 000 iterations using weight normalization and an Adam optimizer with a fixed learning rate of $2 * 10-4$ with a batch size of 16 samples. Each sample is a one-second segment randomly selected from the training dataset. To help model convergence faster, we leverage pre-trained Hifi-GAN model on LJ speech dataset [16] as the starting point of the model.

4. Experiment

4.1. Evaluation Metric

We conduct the MOS (mean opinion score) evaluation on the test set to measure the audio quality. 50 random examples from the original dataset are randomly selected as the evaluation set. Each of the 40 listeners, including 5 experts, listens less than 30 audios per comparison. At each trial, a listener was asked to rate the quality of a utterance in a 5-point scale: "excellent" (5), "good" (4),"fair" (3), "poor" (2), "bad" (1).

4.2. Effect of data processing

We first evaluate the effectiveness of our data preprocessing methods, which include

1) filter, which removes bad quality; 2) peak amplitude and silence normalization at the start and stop of audio; 3) punctuation insertion; 4) concatenate; 5) amplify volume and pitch. As shown in Table 1, we conduct a comparative MOS (CMOS) to evaluate the proposed data preprocessing methods on speech naturalness. Listeners in the CMOS test listen to audios (generated by models trained by data processed in different versions with the same text) each time and evaluates how the latter feels compared to the ground-truth. The result shows that employing our data processing approach significantly improves speech quality.

4.3. Evaluation

We compare the MOS of the generated audio samples by our system with other systems. Specifically, we compare 3 systems i) GT, the ground truth audio; ii) FastSpeech 2; iii) FastSpeech 2 + PostNet, model FastSpeech 2 combines PostNet layer, which is trained on.

Table 1. CMOS comparison in our data preprocessing methods:

Method	CMOS
Original dataset	0
+ f ilter	+ 0.153
+ normalize audio	+ 0.186
+ insert punctuation	+ 0.231
+ concatenate	+ 0.307
+ amplify	+ 0.324

Table 2. The comparison of MOS in mel-spectrogram synthesis:

Method	MOS
GT	4.43 ± 0.05
FastSpeech 2	3.79 ± 0.04
FastSpeech 2 + PostNet	3.95 ± 0.03

the ASR's alignments, the results of which are presented in Table 2. On the other hand, the results of a comparison of vocoder performance and quality, comprising systems i) GT; ii) FastSpeech 2 + Waveglow; iii) FastSpeech 2 + Hifi-GAN; iv) FastSpeech 2 + Hifi-GAN + Denoiser are provided in Table 3.

4.4. Discussion

As shown in Table 1, applying our data processing approach significantly improves the quality of synthesized audio.

Suppose there is no step to concatenate short utterances. In that case, the model will not learn alignment in long utterances due to a lack of significant quantities of lengthy utterances, resulting in the synthesis of words longer than 8 seconds having broken alignment at the end of the sentence, resulting in a distorted sound.

From the Table 2, we can see that our proposed system (last row in Table 2) provides the best MOS. Using duration as an input feature helps the text sequence to match the length of the mel-spectrogram sequence. The PostNet layer is added to help filter noise on mel-spectrogram more effectively.

With the result from Table 3, using the identical mel-spectrogram generated by FastSpeech 2, the sentence synthesized by Waveglow Vocoder is more puzzled by Hifi-GAN Vocoder after the testing process. In our experiments, the mel-spectrogram whose noise is removed by Speech Enhancement model will reduce the quality of the speech synthesis. Meanwhile, the Hifi-Gan model still remains the quality of speech synthesis.

With the evaluation in a VLSP 2021 shown in Table 4, only one system out of all those submitted to the challenge achieved a better result than ours on in-domain, and our MOS out-domain is significantly higher than all other systems (mean score 2.905). In SUS (Semantically Unpredictable Sentences) intelligibility test, we reach 15.00% SER (Sentences Error Rate), the best performing system in TTS task. We believe that one of the main factors to explain the results is the data processing, while most teams almost similarly use the technique [2].

Table 3. The comparison of MOS in waveformsynthesis:

Method	MOS
GT	4.43 ± 0.05
FastSpeech 2 + Waveglow	3.76 ± 0.04
FastSpeech 2 + Hifi-GAN	3.88 ± 0.03
FastSpeech 2 + Hifi-GAN	3.95 ± 0.03
+ Denoiser	

Table 4. The result from organizer:

Test	Score
MOS (in-domain)	3.94
MOS (out-domain)	3.30
SUS Intelligibility (SER %)	0.15

5. Conclusion

In this paper, we have presented the joint submission to VLSP Workshop 2021. The system consists of a data preprocessing pipeline, an acoustic model based on FastSpeech 2 and a Hifi-Gan neural vocoder. To summarize, we have demonstrated that the data processing method described in this research aids in improving the naturalness of the speech synthesizer. It is difficult to collect and construct data for spontaneous voices, and it is dependent on the accessibility of the domain. Therefore, optimizing processing stages and taking advantage of the standard TTS model to build a spontaneous speech synthesis system is essential for the above problem. Despite its overall performance, we believe that our system would have obtained better results if the vocoder had been given more time to adapt to the voice provided. In future work, we will consider modeling spontaneous speech characteristics and experiment with different training techniques. The aim is to reduce the number of artifacts and improve the overall quality of the synthetic speech in terms of naturalness of the synthesized speech.

Acknowledgments

This work has been supported by Viettel Cyberspace Center, Hanoi (VTCC).

References

[1] Y. D. Zhaoxi Mu, Xinyu Yang, Review of end-to-end speech synthesis technology based on deep learning, arXiv:2104.09995.

[2] N. T. T. Trang, N. H. Ky, TTS Challenge: Vietnamese Spontaneous Speech Synthesis, VLSP 2021.

[3] S. S. N. Shiva Sundaram, An Empirical Text Transformation Method for Spontaneous Speech Synthesizers, Interspeech, 2003, pp. 1221–1224.

[4] J. B. J. G. Éva Székely, Gustav Eje Henter, How To Train Your Fillers: Uh And Um In Spontaneous Speech Synthesis, The 10th ISCA Speech Synthesis Workshop , 2019, pp. 245–250.

[5] R. C. d. Nguyen Thi Thu Trang, Nguyen Ky, Prosodic Boundary Prediction Model for Vietnamese Text-To-Speech, INTERSPEECH, 2021, pp. 3885–3889.

[6] N. T. T. Trang, D. X. Bach, N. X. Tung, A Hybrid Method for Vietnamese Text Normalization, NLPIR 2019: Proceedings of the 2019 3rd International Conference on Natural Language Processing and Information Retrieval.

[7] K. D. Brecht Desplanques, Jenthe Thienpondt, ECAPA-TDNN: Emphasized Channel Attention, Propagation and Aggregation in TDNN Based Speaker Verification, arXiv:2005.07143.

[8] A. T. D. Q. B. N. Viet Lam Phung, Phan Huy Kinh, Data Processing for Optimizing Naturalness of Vietnamese Text-to-speech System, arXiv:2004.09607.

[9] P. Z. Y. Y. Yukun Liu, Ta Li, Improved Conformer-based End-to-End Speech Recognition Using Neural Architecture Search, arXiv:2104.05390.

[10] X. T. T. Q. S. Z. Z. Z. Yi Ren, Yangjun Ruan, T.-Y. Liu, Fastspeech: Fast, robust and controllable text to speech, In Advances in Neural Information Processing Systems, 2019, pp. 3165–3174.

[11] Q. T. L. N. A. N. T. Linh Dang Le, Tien Thanh Nguyen, The Falcon team's system for VLSP 2020 TTS, VLSP 2020 Evaluation Campaign.

[12] X. T. T. Q. S. Z. Z. Z. T. Y. L. Yi Ren, Chenxu Hu, Fastspeech 2: Fast and high-quality end-to-end text to speech, arXiv:2006.0455.

[13] J. B. J. Kong, J. Kim, HiFi-GAN: Generative Adversarial Networks for Efficient and High Fidelity Speech Synthesis, https://arxiv.org/abs/2010.05646

[14] R. J. W. M. S. N. J. Z. Y.-Z. C. Y. Z. Y. W. R. S.-R. R. A. S. Y. A. J. Shen, R. Pang, Y. Wu, Natural TTS Synthesis by Conditioning WaveNet on MEL Spectrogram Predictions, in IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2018 pp. 4779–4783.

[15] R. V. R. Prenger, B. Catanzaro, Waveglow, A Flow-Based Generative Network for Speech Synthesis, In ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP, 2019, pp. 3617–3621.