SoftSpeech: Unsupervised Duration Model in FastSpeech 2

Yuanhao Yi, Lei He, Shifeng Pan, Xi Wang, Yuchao Zhang

Microsoft

{yuanhyi, helei, peterpan, xwang, yuchao.zhang}@microsoft.com

Abstract

In this paper, we propose a neural Text-To-Speech (TTS) system SoftSpeech, which employs a novel soft length regulated duration attention based decoder. It learns the encoder output mapping to decoder output simultaneously from an unsupervised duration model (Soft-LengthRegulator) without the requirement of external duration information. The Soft-LengthRegulator consists of a Feed-Forward Transformer (FFT) block with Conditional Layer Normalization (CLN), following a learned upsampling layer with multi-head attention and guided multi-head attention constraint, and it is integrated in each decoder layer and achieves accelerated training convergence and better naturalness within FastSpeech 2 framework. Soft Dynamic Time Warping (Soft-DTW) is adopted to align the mismatch spectrogram loss. Moreover, a Fine-Grained style Variational AutoEncoder (VAE) is designed to further improve the naturalness of synthesized speech. The experiments show SoftSpeech outperforms FastSpeech 2 in subjective tests, and can be successfully applied to minority languages with low resources.

Index Terms: TTS, Soft-DTW, Attention, FastSpeech, VAE

1. Introduction

Neural TTS models have rapidly improved with various neural networks, such as Deep Neural Network (DNN) [1] and Recurrent Neural Network with Long-Short Term Memory (LSTM-RNN) [2]. These neural networks have been applied to speech synthesis and demonstrated their superiority over traditional Hidden Markov Models (HMMs) [3]. Recently, Some new neural models, such as Tacotron [4], Tacotron 2 [5], Deep voice 3 [6], Clarinet [7], TransformerTTS [8] has been proposed to generate speech autoregressively from text input that can achieve performance very close to human quality. In order to increase inference speed and generate more robust speech, non-autoregressive TTS models such as FastSpeech [9], FastSpeech 2 [10] and Parallel Tacotron [11] are proposed to generate high quality speech in fast speed with robust and controllable ability.

However, to train such non-autoregressive TTS model, phoneme duration are necessarily obtained from autoregressive teacher models [4, 5, 8] or external alignment information [12, 13, 14] to map text to speech, and the distillation from teacher is complicated and time-consuming, while ground truth duration relies on external aligner model. Meanwhile, duration model as condition input is not propagated jointly with decoder and the discrepancy between predicted and groundtruth duration will critically affect the performance of the model. To that end, some TTS models including unsupervised duration modeling have been proposed which does not require supervised groundtruth phoneme duration, such as EATS [15], VITS [16], Glow-TTS [17], Parallel Tacotron 2 [18]. And our proposed model SoftSpeech leverages the upsampling layer from Parallel Tacotron 2 which enables error gradients to be propagated through all operations within FastSpeech 2 framework, and we make improvements on the unsupervised duration model and decoder, and conduct extensive experiments. The contributions of this paper are summarized as follows:

- We achieve a unsupervised duration model (Soft-LengthRegulator) within FastSpeech 2 framework by introducing FFT block with CLN, following a learned upsampling layer with multi-head attention and guided multi-head attention constraint.
- We integrate Soft-LengthRegulator in each decoder layer and achieve accelerated training convergence and better naturalness.

The rest of this paper is organized as follows: Details of the proposed model are presented in Section 2. Experimental results are shown in Section 3. Section 4 concludes this paper.

2. Proposed Model

The proposed model SoftSpeech aims to synthesize multi-speaker, multilingual speech which consists of a Text Encoder, Fine-Grained Style VAE, Fine-Grained Style Predictor, Soft-LengthRegulator and Decoder as illustrated in Figure 1.

2.1. Text Encoder

Text Encoder is composed of 6 layers of FFT block [19], and the block is stacked by a multi-head self-attention layer with dropout 0.1 with CLN [20] before and residual connection [21] after, followed by two Feed-Forward layers with CLN before and residual connection after. We refer to the structure of CLN in AdaSpeech [22] that replace the scale and bias which generated by two linear layers using 128-dimensional language embedding. And we find the CLN with language embedding is important when it comes to multilingual task. The number of attention heads is set to 4 and the attention dimension is 384. The Feed-Forward layers have 384 and 1536 hidden units respectively. The Text Encoder takes 384-dimensional phoneme embedding followed by Positional Encoding [19] as input. The output of Text Encoder are concatenated with 128-dimensional speaker embedding, language embedding and the style latent representation which generated from Fine-Grained Style VAE to output phone-level contextual feature.

2.2. Fine-Grained Style VAE and Predictor

Fine-Grained style VAE uses multi-head attention mechanism [19] to align the frame-level Mel-Spectrogram with the phone-level Text Encoder output, to get a sequence of phone-level style latent representation. To that end, we first concatenate the Mel-Spectrogram with Positional Encoding and the speaker embedding, and following an aforementioned FFT block to form the Query in the attention mechanism. Then we concatenate the encoder output with the speaker embedding and language embed-

Audio examples at: https://luckeryi.github.io/microsoft/SoftSpeech
Phoneme Embedding
Positional Encoding
Speaker Embedding
Language Embedding
Text Encoding

Figure 1: The overall architecture for SoftSpeech.

We hypothesize this expectation of the KL's distribution positive as follow:

\[
\mu = \text{BN}(\mu_\theta) \cdot \sqrt{\tau + (1 - \tau) \cdot \text{Sigmoid}(\varphi)},
\]

where BN is batch normalization without learnable affine parameters. \(\tau \in (0, 1)\) and we empirically set \(\tau = 0.5, \varphi\) is a trainable parameter.

Like [11, 24], we jointly train a Fine-Grained Style Predictor to predict the style latent representation used in inference. It consists of two GRU layers with hidden size of 128 followed by a linear layer which uses phoneme embedding concatenating the speaker embedding and language embedding as input and predicts the regularized posterior mean \(\hat{\mu}_\theta\).

### 2.3. Soft-LengthRegulator

Soft-LengthRegulator is designed to upsample the phone-level contextual feature \(H\) to frame-level \(O\). As shown in Figure 3, it consists of the aforementioned FFT block with CLN followed by a learned upsampling layer. The scale and bias in CLN which generated by two linear layers using the speaker embedding. The learned upsampling is leveraged from Parallel Tacotron 2 [18], and we modify the primitive single head attention to multi-head attention. First, we calculate the duration start and end Token Boundary Grid matrices \(S_{m \times n}\) and \(E_{m \times n}\) by

\[
S_{i,j} = i - \sum_{k=1}^{i-1} d_k, E_{i,j} = \sum_{k=1}^{i} d_k - i,
\]

where \(S_{i,j}\) indexes the \((i, j)\)-th element in the matrix. We calculate the primary multi-head attention matrix \(W_{m \times n \times q}\) and auxiliary context matrix \(C_{m \times n \times q}\) following:

\[
W = \text{Softmax}(\text{MLP}(\{S, E, \text{Exp(Conv1D(Proj(H))))\})), \quad (4)
\]

\[
C = \text{MLP}(S, E, \text{Exp(Conv1D(Proj(H))))}, \quad (5)
\]

Where \(\text{Exp}()\) and \(\text{Proj}()\) represent adding an extra dimension by repeating the input matrix by \(m\) times, and one linear layer with input and output dimensions of \(h\), respectively. \(\text{Conv1D}()\) is one-dimensional convolution operation with layer normalization and Swish activation [25]. The input and output dimensions of \(\text{Conv1D}()\) are \(h\) and 8. \([\cdot]\) stands for matrix concatenation along the hidden dimension, and gets a hidden dimension of \(10 = 1 + 1 + 8\). \(\text{MLP}()\) is a two-layer full-connected network with Swish activations. The numbers underneath \(\text{MLP}\) denote the input and output hidden dimensions. We set head numbers \(p = 2\) and \(q = 4\). The \(\text{Softmax}()\) operation is performed on the phoneme sequence time dimension. We calculate the frame-level contextual feature output \(O_{n \times h}\) with the following equation:

\[
O = \text{Proj}(WH) + \text{Proj}(\text{Einsum}(W, C)) \quad (6)
\]

where \(\text{Einsum()}\) represents the einsum operation (‘qmn, mnp \rightarrow qmp’, \(W, C\)). We first permute \(W\) from \(m \times n \times q\) to \(q \times m \times n\) for computation, and after we get \(WH\) with shape \(q \times m \times h\) and \(\text{Einsum}(W, C)\) with shape \(q \times m \times p\), we reshape them to \(m \times qh\) and \(m \times qp\) respectively for final projection to dimension \(m \times h\).

### 2.4. Decoder

The network architecture of the Decoder is illustrated in Figure 2. It consists of 6 layers of aforementioned FFT block with CLN and a Soft-LengthRegulator which takes phone-level contextual feature in each layer as input. For the motivation we integrate the Soft-LengthRegulator to decoder, first, we hypothesize this extra introduced Soft-LengthRegulator achieves more information

---

Soft-LengthRegulator module can be easily leveraged and the related code at: https://github.com/LuckerYi/SoftSpeech
interaction between phone-level contextual and frame-level approximate Mel feature that makes the duration model more robust and improves flexibility and robustness of decoder at the same time, second, the internally predicted durations from only outsider Soft-LengthRegulator may add the difficulty to the decoder, and affect the performance of generated speech. During inference, sentence frames $T$ predicted by the outside Soft-LengthRegulator will be used as groundtruth sentence frames to calculate the corresponding multi-head attention matrices of Soft-LengthRegulator in each Decoder layer.

2.5. Final Loss

A differentiable loss function based on Soft-DTW [26] is introduced as reconstruction loss to handle the length mismatch between predicted and groundtruth Mel-Spectrogram. We set the Sakoe-Chiba bandwidth to 120 for pruning computation. The warp penalty is set to 1.0. The soft minimum temperature is set to 0.01. The point-wise distance function is Manhattan Distance. $\text{KL}$ divergence between prior and posterior, $\beta$ is set 0.5, and a beta value of (0.9, 0.998). We use Noam’s learning rate decay scheme [19], warmup-steps is set 8,000. A MelGAN vocoder [30] was built to reconstruct audio from Mel-Spectrograms for all models.

3. Experiments

We evaluate the performance of SoftSpeech on two tasks: multi-lingual multi-speaker TTS and minority low resource TTS. We also conduct ablation experiments to explore the effectiveness of Soft-related modules. We will describe the experimental details and results in following subsections.

3.1. Setup

We use a sampling rate of 16000 Hz and Mel-Spectrograms with 80 bins using librosa mel filter defaults. We apply the STFT with a FFT size of 2048, hop size of 200, and window size of 800 samples. And we use FastSpeech 2 from [10] as our baseline model. For multi-lingual multi-speaker TTS task, we used a proprietary speech dataset around 2400 hours including 51 locales and 394 speakers. We evaluate English (US 1 male & 1 female) and Mandarin (China, 1 male & 1 female), and the amount of training data for the evaluated speakers varied from 2 hours to 49 hours. For minority low resource TTS task, we used a proprietary minority low resource speech dataset around 2 hours including Estonian (Estonia, 1 male & 1 female) and Maltese (Malta, 1 male & 1 female).

We conducted crowd-sourced Mean Opinion Score (MOS) tests to evaluate the audio quality. We generate 45 sentences for each test, and each sentence is listened by 20 crowd-sourcing judges. Judges listened to randomly selected audio samples, and rated their naturalness on a 5 point scale (1: Bad, 2: Poor, 3: Fair, 4: Good, 5: Excellent). We convert the text sequence into the phoneme sequence with our internal grapheme-to-phoneme conversion tool [28] in order to alleviate the mispronunciation problem for multi-lingual multi-speaker TTS task, while using the raw characters as input text sequence for minority low resource TTS task. We use the ADAM [29] optimizer with an initial learning rate of 0.5, and a beta value of (0.9, 0.998). We use Noam’s learning rate decay scheme [19], warmup-steps is set 8,000.

Table 1: MOS between SoftSpeech and baseline model FastSpeech 2 on multi-lingual multi-speaker TTS task. S-L-D represents Soft-LengthRegulator in Decoder.

<table>
<thead>
<tr>
<th>Model</th>
<th>MOS (CI)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Recording</td>
<td>English: 4.44 ± 0.07, Mandarin: 4.47 ± 0.06</td>
</tr>
<tr>
<td>FastSpeech 2</td>
<td>English: 4.21 ± 0.08, Mandarin: 4.19 ± 0.08</td>
</tr>
<tr>
<td>SoftSpeech</td>
<td>English: 4.32 ± 0.07, Mandarin: 4.31 ± 0.08</td>
</tr>
<tr>
<td>w/o S-L-D</td>
<td>English: 4.25 ± 0.08, Mandarin: 4.22 ± 0.08</td>
</tr>
</tbody>
</table>

3.2. Result

For multi-lingual multi-speaker TTS task, we conduct MOS between Recording, SoftSpeech and the baseline model FastSpeech 2. As illustrate in Table 1, SoftSpeech outperforms the baselines in naturalness. And it can be seen that the integration of Soft-LengthRegulator in Decoder can further improve the naturalness.
Table 2: MOS between SoftSpeech and Recording on minority low resource TTS task using raw character as input text sequences.

<table>
<thead>
<tr>
<th>Model</th>
<th>Estonian MOS (CI)</th>
<th>Maltese MOS (CI)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Recording</td>
<td>4.45 ± 0.08</td>
<td>4.36 ± 0.09</td>
</tr>
<tr>
<td>SoftSpeech</td>
<td>4.19 ± 0.09</td>
<td>4.14 ± 0.10</td>
</tr>
</tbody>
</table>

For minority low resource TTS task, it is hard to get accurate raw character or phoneme duration due to very low resources of data. As the performance of baseline model is highly dependent on the duration, so we conduct MOS only between Recording and SoftSpeech. As illustrate in Table 2, SoftSpeech can be well extended to minority languages with low resources, and achieve good synthetic speech naturalness even using raw character as input text sequences.

Table 3: Real-Time Factor (time used per second audio) of different models.

<table>
<thead>
<tr>
<th>Model</th>
<th>RTF (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>FastSpeech</td>
<td>0.0344s</td>
</tr>
<tr>
<td>SoftSpeech</td>
<td>0.0417s</td>
</tr>
</tbody>
</table>

As illustrate in Table 3, we also have conducted the Real-Time Factor (RTF) which could have helped to check the inference time. Our device for testing was a V100 GPU and we end to end synthesis 210 sentences and count the total audio time as well as the synthesis time to calculate the RTF; the result shows that SoftSpeech gets slightly slower generation speed, but still competitive.

As illustrate in Table 4, we conduct Comparative Mean Opinion Score (CMOS) for ablation studies to verify the effectiveness of each component in SoftSpeech. And the CMOS in our paper which already contains statistical significance test when the result of CMOS score is greater than 0.1, it shows that there is a significant difference ($p < 0.001$). We can see that removing any module will result in performance drop in voice quality, especially the Soft-LengthRegulator, demonstrating the effectiveness of each component in SoftSpeech.

As shown in Figure 4, we conduct unsupervised alignment comparison, and we select the dominant head attention in S-L and S-L-D for comparison. It can be seen that at the same training steps (only 30K), the integration of Soft-LengthRegulator in Decoder (S-L-D) can achieve accelerated training convergence and get better alignment matrix. We also compare the phoneme durations accuracy between predicted and ground truth phoneme durations, which are obtained by predicted alignment matrix and HMM-based force alignment, respectively. As shown in Table 5, our proposed SoftSpeech can also achieve better duration prediction.

4. Conclusion

This paper has proposed SoftSpeech, a novel neural TTS model with an unsupervised duration model (named Soft-LengthRegulator) based on FastSpeech 2 framework. The integration of Soft-LengthRegulator in each decoder layer can achieve accelerated training convergence, better robustness and naturalness. The experiments show our proposed model outperforms the baseline FastSpeech 2 and can be well extended to minority languages with low resources. Future work includes investigating the generalization of Soft-LengthRegulator based on Transformer.
5. References


