Submission from CMU for Blizzard Challenge 2019

SaiKrishna Rallabandi, Peter Wu and Alan W Black

Language Technologies Institute, Carnegie Mellon University, Pittsburgh, PA, USA.

Abstract

In this paper we present the entry from CMU to Blizzard speech synthesis challenge 2019. We begin with a description of build process for our base voice. We then present the following modifications to base voice: (1) We investigate the effectiveness of sub-sentence training of acoustic models aimed at better utilization of available aligned data (2) We investigate the applicability of strategic gradient backpropagation to accelerate the training (3) We experiment with iterated dilated convolutions in WaveNet to obtain compact models. Although our current performance seems very inefficient, we are actively pursuing approaches to strengthen our voice building framework. We believe we are progressing in the right direction and anticipate a much stronger performance in the coming evaluations.

Index Terms: speech processing, convolutional neural networks, Tacotron, WaveNet

1. Introduction

Blizzard speech synthesis challenges were devised to better understand different corpus driven speech synthesis techniques on a common dataset. As a part of this, current evaluation focuses on building voices based on data resources from the internet. Language for the current evaluation is Mandarin. Our submission to this year’s challenge was based on statistical parametric speech synthesis framework. Specifically, we have employed Sequence to Sequence neural network based approach to map the textual content to corresponding acoustics. There have been continuous and significant improvements in all aspects of this framework of speech synthesis from textual representations[1] through post filtering[2].

Approaches such as[3, 4, 5, 6] have demonstrated that Seq2Seq models are capable of reliably learning reasonable associations between the textual and acoustic modalities. These approaches have been utilized in building systems for new languages[7] as well as improving the models for existing ones[8]. Moreover, adaptation of these approaches to various tasks has been investigated[9]. Similarly, the world of vocoding has seen tremendous progress. Deep Neural Generative models aimed at vocoding[10] aim to model the joint probability of the data distribution and the conditioning information as a product of conditional distributions. Typical implementation of such models follows an autoregressive framework[10] although other formulations[11] have been suggested as well. Such models have proven very effective in addressing one of the major challenges with conventional vocoding techniques - fidelity. These advances have led to flexible systems capable of generating different styles[12, 13, 14, 13] of speech and ability to build voices from noisy[15] or very minimal data[16]. The elegance of such Seq2Seq models comes from the fact that they can be trained without making assumptions based on prior knowledge specific to speech. Therefore, we have employed Seq2Seq based approach as our cornerstone toward building our submission and have made extensions to the same.

Broadly, we have investigated the following approaches via our current submission:

• We investigate the possibility of jointly training acoustic model and the vocoder.
• Since the amount of aligned speech and text after our initial alignment step was limited, we investigate the applicability of sub sentence training in the context of acoustic models. We accomplish this using an external duration model at the phone level.
• To accelerate the training, we investigate the usage of strategic back propagation of gradients.
• To build compact and stable vocoders, we investigate building vocoders with shared parameters. Specifically, we share the parameters of all dilated convolutions with the same dilation rate in WaveNet.

The rest of this paper is organized as follows: We begin with a description of our base voice in section 2. We then present various approaches we investigated in section 3. This is followed by evaluation results and discussion. We present the observations made from post evaluation analysis and conclude this report.

2. BaseVoice

In this section, we describe our base voice built using a new module, FALCON Seq2Seq within Festvox[17] framework for statistical parametric speech synthesis. In brief, our system consists of two jointly trained components: (1) Acoustic model to predict acoustic vectors on a per frame basis and (2) Vocoder which generates speech on a per sample basis conditioned on the predicted acoustic vectors. For the current submission we
have not performed any postfiltering at the acoustic vector or sample level.

2.1. Data
The database used for building our submission was collected from an internet talk show by a well known Mandarin character. Topics of the utterances seemed contemporary and dealt with a variety of issues. We are provided 8 hours of speech data and corresponding transcription. Each utterance was one minute long, leading to a total of 480 utterances.

2.2. Voice Building
As preprocessing, we perform the following steps:

- **Alignment and Segmentation:** Since the provided utterances seemed too long to build a sequence to sequence acoustic model, we have performed segmentation of the utterances into smaller chunks (referred to as sub utterances hereafter) using HMM based alignments. For acoustic models to perform alignment we have used multilingual models built as part of [18]. This resulted in 2178 sub utterances totaling three hours of speech. The mean duration of sub utterances was 5 seconds while the maximum utterance was of length 23 seconds. The normalized histogram of sub utterance durations can be seen in figure 2 (b).

![Normalized histogram plot of utterance lengths in Arctic dataset](image)

**Figure 2:** (a) Normalized histogram plot of utterance lengths in Arctic dataset (b) Normalized histogram plot of sub utterance lengths from our alignment module

- **Conversion of aligned text to Pinyin:** Our front end currently cannot process Mandarin characters. Hence we have converted the text data from the sub utterances into Pinyin.

- **Tokenization and G2P:** We consider any text entry separated by white space as a token. Once tokens are obtained, we have used US phoneset to perform mapping from graphemes to phonemes based on CMU pronunciation dictionary.

- **Acoustic Feature Extraction:** We perform acoustic feature vector extraction over a 50ms frames obtained by applying a hamming window with a frame shift of 12.5 msec. The speech files have been high pass filtered with cut off frequency of 60 Hz prior to feature extraction. We obtain 1025 dimensional linear and 80 dimensional mel frequencies.

- **Acoustic Model Component:** Our acoustic model is based on Tacotron [3] Seq2Seq speech synthesis system and is shown in the figure 1. We have used phones as the input instead of characters. We have not performed masking the loss value for padded frames as is typically done in Seq2Seq models. We found that not masking forces to predict (zero) padded frames as well and helps the model better predict end of sentence as mentioned in [3]. Since adjacent frames seem to be correlated, our decoder predicts 3 frames per timestep. We have used a batch size of 64 to train the baseline model.

  - **Vocoder:** Our vocoder is based on WaveNet [10]. Speech signal was power normalized and squashed to the range (-1,1). We have used 16 bit mulaw quantization to encode individual samples. Instead of transposed convolutions we have employed linear interpolation to upsample the acoustic frames to match the time resolution of speech samples. To optimize the model, we use discretized Mixture of Logistiscs loss [19, 20] with 12 logistic classes.

Our acoustic model and vocoder are trained jointly. While forcing the model to predict padded frames facilitates better prediction of end of sentence, authors are not aware of similar constraints to tighten the internal segment boundaries. We hypothesize that employing joint training would force local attention to better learn acoustic correspondences between segment boundaries. Our model consists of three loss components: (1) L1 Divergence between predicted and original mel spectrograms (2) L1 Divergence between predicted and original linear spectrograms (3) Mixture of Logistics loss from the vocoder. To make the training of vocoder faster, we have used chunks of randomly selected 8000 timesteps of raw signal and corresponding predicted acoustic frames. We have used a batch size of 16 and an upsampe convolution block with 4 layers that upsamples at {2, 2, 4 and 5} times respectively.

3. Experiments

### 3.1. Sub sentence Training using aligned segment durations

From figure 2, it can be observed that sub utterances from our alignment module are skewed towards longer lengths. In our informal model component evaluations, we observed that the acoustic model training seemed unstable compared to other models we built using English data. Inspection of attention mechanism led us to believe that the model was failing to converge. We hypothesized that this might be due to the wide variety in the length of our sub utterances. Based on analysis in section 3.1.1, we postulate that a reasonable way to circumvent this issue is to use sub-sentence training: To train smaller chunks within the sub-utterances since we already have duration information from our alignment module.

#### 3.1.1. Role of Attention in Sub Utterance Alignment

Let $p_{t_1}...p_{t_m}$ denote the phonemes in the textual domain that have been transformed by an encoder network to state vectors $p_{t_1}...p_{t_m}$. Let $y_{t_1}...y_{t_m}$ denote acoustic frames in the target sequence. A typical attention based encoder decoder network such as Tacotron factorizes the joint probability of acoustic frames $P_r(y_{t_1}...y_{t_m}|p_{t_1}...p_{t_m})$ as $\Pi_{t=1}^{m} Pr(y_{t} | p_{t}...p_{t_n}, s_{t})$ where $s_{t}$ is a decoder state summarizing $y_{t_1}...y_{t_m}$. For each time step $t$, an attention variable $a_t$ is used to denote which encoded phoneme state of $p_{t_1}...p_{t_m}$ aligns with $y_{t}$. Let $P(a_t = j|p_{t}...p_{t_n}, s_{t})$ denote the probability that encoder state $p_{t}$ is relevant for output $y_{t}$. Typically this conditional probability is estimated using a softmax function over attention scores computed from $p_{t}$ and decoder state $s_{t}$ as follows.
\[ P(a_t = j|p_1\ldots p_m, s_t) = \frac{e^{A_\theta(p_j, s_t)}}{\sum_k e^{A_\theta(p_k, s_t)}} \]  

(1)

where \( A_\theta \) is the attention unit that scores each input state \( p_j \) as per the decoder state \( s_t \). This is followed by a convex combination of the input states to model log likelihood for each output acoustic vector \( y_t \).

\[ \log Pr(y_t|x_1\ldots x_m) = \log Pr(y_t|\sum_a P_t(a)x_a) \]  

(2)

In this scenario, attention is essentially a latent deterministic variable that is conditionally dependent on the convex combination from encoded representation of input phonemes. It is responsible for the sentence internal association between textual and acoustic modalities.

### 3.1.3. Refinement of Segment Boundaries

We posit that it is possible to improve the association learnt by acoustic model by using sub sentence training: selecting aligned segments of text and acoustics within a sentence. In addition, selecting segments within a sentence might lead to the model utilizing the available data more efficiently. We note that such an approach is already used for vocoding: Typical vocoders the authors are aware of are trained using aligned chunks of acoustic vectors and corresponding speech samples as opposed to full utterances. While this is due to GPU memory constraints in the context of vocoders, we investigate applying similar strategy to acoustic modeling. We believe that this facilitates local attention mechanism to better learn the mapping from phonetic space to the acoustic space.

The steps we have used for sub sentence training are mentioned below:

**Algorithm 1** Sub sentence Training of Acoustic Model

```
Obtain segment durations using EHMM
Refine segment durations using MoveLabel

While n < Global Updates do
  For instance in batch do
    StartIdx = random(0, SubUtteranceLength)
    StartSegment, StartDur = findClosest(StartIdx, AlignedSegments)
    EndIdx = StartIdx + SelectedDuration
    EndSegment, EndDur = findClosest(EndIdx, AlignedSegments)
  End for
  Add selected sub sentence <text, frames> pair to batch
  Iterate over batch
End while
```

### 3.2. Strategic Backpropagation to enable joint training

While training our base model, inspection of scalar loss value indicated that the model reaches around 95% of the optimal loss value within 10K steps (5% of training steps). Subsequent gradient updates result in minute contributions towards the transition towards optimal loss value. The authors have not found any existing literature analyzing this region in the context of Seq2Seq. Text to speech models. We hypothesize that the model compensates for macro issues (e.g., pronunciation of vowels vs consonants) in the initial stages and spends rest of training phase in filling the micro details (e.g., pronunciation of “ax” vs “ah”).

We posit that contribution from smaller gradient updates in the micro detail phase can be ignored to result in better training: Since the absolute magnitude of the gradients is small while they contribute equal weight to normalization, ignoring such gradients might lead to sharper overall update helping the model in the micro detail phase. Note that L1 divergence which is employed in Seq2Seq TTS models also results in sharper gradients compared to L2 divergence which was employed in segment based statistical parametric methods.

To facilitate this, we select the gradient updates to include in the model training based on the absolute loss values after each forward pass during training. The steps are mentioned in Algorithm 2. This approach is inspired by selective back propagation[23]. However, they disable backpropagation using a hard rule: When the threshold hyperparameter is set to 0, the gradient is backpropagated only when the model makes an error. While this may be suitable in the context of classification tasks for which their model was deployed, tasks involving regression

<table>
<thead>
<tr>
<th>Pass</th>
<th>No. of Moves</th>
<th>MCD</th>
<th>F0 Error</th>
<th>Duration Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>58941</td>
<td>7.065</td>
<td>32.808</td>
<td>0.964</td>
</tr>
<tr>
<td>2</td>
<td>55467</td>
<td>7.049</td>
<td>32.712</td>
<td>0.964</td>
</tr>
<tr>
<td>3</td>
<td>51224</td>
<td>7.047</td>
<td>32.7</td>
<td>0.963</td>
</tr>
<tr>
<td>4</td>
<td>48930</td>
<td>7.041</td>
<td>32.4</td>
<td>0.959</td>
</tr>
<tr>
<td>5</td>
<td>47844</td>
<td>7.036</td>
<td>32.35</td>
<td>0.948</td>
</tr>
<tr>
<td>6</td>
<td>47342</td>
<td>7.022</td>
<td>32.23</td>
<td>0.946</td>
</tr>
<tr>
<td>7</td>
<td>46541</td>
<td>7.021</td>
<td>31.91</td>
<td>0.942</td>
</tr>
<tr>
<td>8</td>
<td>43451</td>
<td>7.018</td>
<td>31.82</td>
<td>0.942</td>
</tr>
<tr>
<td>9</td>
<td>42872</td>
<td>7.020</td>
<td>31.86</td>
<td>0.947</td>
</tr>
</tbody>
</table>
sent a filter and a gate, respectively. The joint probability of a
represents a convolution weight. The subscripts
activation function is given by following equation:
ing audio signal. The input to WaveNet is subjected to corre-
WaveNet [10] is an autoregressive neural model with a stack
vlar neural network design (WTA approach in [27]).
the discrete state representation at time
to parameter tying.
our approach can be seen as a softer version, similar to [24]:
When absolute value of a gradient is less than preset thresh-
hold, its inclusion in the backpropagation is stochastically deter-
ined. Other variants of this approach can be employed such as
such as scheduling the threshold as well as the stochasticity. However,
in our current submission we have employed a simple stochastic-

tivity component: If a gradient is less than the threshold, it is
cluded in the backpropagation based on a coin toss.
Our approach can also be interpreted from the view of se-
lective attention[25] to the most relevant contributors[26]. In
its extreme case, choosing to only use the gradient with the
most loss magnitude is similar computationally to hard atten-

tic component: If a gradient is less than the threshold, its inclusion in the backpropagation is stochastically deter-


\begin{algorithm}
\caption{Strategic Back-propagation}
\begin{algorithmic}
  \STATE Forward pass through the model
  \STATE Compute divergence values for each instance in batch.
  \STATE loss=[]
  \STATE Pick a threshold value
  \FOR {divergence in BatchOfDivergences}
    \IF {divergence > threshold}
      \STATE loss+=divergence
    \ELSE
      \STATE idx=random(0,1)
      \IF {idx > 0.5}
        \STATE loss+=divergence
      \ENDIF
    \ENDIF
  \ENDFOR
  \STATE BatchLoss=sum(loss)/len(loss)
  \STATE Back propagate loss
\end{algorithmic}
\end{algorithm}


\begin{equation}
\hat{y}_t \sim \sum_{d=0}^{D} h_d * r_d(x) \tag{5}
\end{equation}

where \( x, y \) represent input and output vectors; \( D \) is the num-
ber of different dilation used and \( d \) is the dilation factor; \( h_d \)
is the convolution weights. This stack of convolutions is re-
peated multiple times in the original WaveNet. Optimization
in WaveNet is performed based on the error between predicted
and the ground truth sample conditioned on previous
samples in the receptive field alongside the local conditioning.
Expressing the loss function being optimized mathematically
the error at sample \( t \) is:

\begin{equation}
l_t = Div(\hat{y}_t || y_t) \tag{6}
\end{equation}

Here, we define the divergence similar to the [19]. To opti-
imize this loss, the contribution from the individual convolution
layers towards this global error function must be nullified. Now
let us consider the expression for intermediate output for a sin-
gle filter in Eqn 5:

\begin{equation}
x_{out}(t) = \sum_{\tau=0}^{t} h(\tau)x(t - \tau) \tag{7}
\end{equation}

where \( \tau \) is the receptive field covered by the model and \( h(\tau) \)
represents the discrete state representation at time \( t \). Without
loss of generality and dropping the term \( \tau \) for brevity, the spec-
tral representation generated by the model can be expressed as:

\begin{equation}
Y(z) = H(z)X(z) \tag{8}
\end{equation}

Considering the discrete nature of input from Eqn 6, an
interpretation of Eqn 8 is that the neural auto regressive model
acts as the transfer function and is discretized by convolving
with the samples from original signal. It has to be noted that this
is similar to the formulation of source filter model of speech,
specifically the periodic components corresponding to voiced
sounds. Voiced sounds typically represented as impulse train
are convolved with the transfer function to generate spectral en-
velope. As a corollary, from equation 6 and 8, we posit that
the optimization in WaveNet model is performed by minimiz-
ing the divergence between true and approximate spectral en-
velopes. With the presented interpretation, we hypothesize that
sharing parameters across the filters(individual dilated convolu-
tion components) facilitates the filters to be more stable com-
pared to a non shared scenario. This is also inspired by obser-
vations that show efficiency of such parameter sharing [28, 29].
In terms of model size, we obtain a reduction of two thirds due
to parameter tying.

\section{Evaluation}

The subjective evaluation was conducted based on various cate-
gories: pleasantness, speech pauses, stress, intonation, emotion,
listening effort. The identifier of our system is P. Mean opinion
score of our system as provided by all listeners is depicted in
figure 3.
We have ranked last in the current evaluation. Informal evaluations of our submission revealed that speech generated by our model lacks comprehensibility. To analyze the reason for this, we have performed an internal evaluation of individual approaches proposed in the current submission. For this we have used Arctic[30], LJSpeech[31] and Indic[32] datasets. Although these datasets differ in the language of the current evaluation and the findings might not be transferable, we have chosen them for initial analysis since we have worked with these datasets in the past. Based on these ablation evaluations, we currently believe that the component responsible for most degradation of quality in our system is that of sub sentence training.

4.1.1. Sub Sentence Training - Only Acoustic Model

We performed sub sentence training of acoustic model using Arctic and Indic datasets. The duration of sub sentences was varied from 1.5 to 8 seconds. We have observed that while during training the model seems to learn association between the textual input and acoustic vectors, there were instances where the synthesized speech was either distorted or incoherent with the text. Most of these instances were found when the ratio of selected duration to the mean length in the dataset was less than 40%. In addition, a histogram analysis showed that the sub utterances obtained from our alignment module are significantly longer compared to the other datasets we have employed attention based models thus far. For comparison, a histogram plot of Arctic dataset can be seen in figure 2. While the length distribution was not as regular as Arctic for the other dataset/LJSpeech we compared against, we observed that the maximum length in [31] was still less than in our sub utterances.

Here are some of the things we believe we should improve for our next submission:

- Data size from the output of our alignment module was 3 hours compared to 8 hours that was originally provided. Although we implemented sub sentence approach to handle this, we believe a much stronger approach would be to investigate approaches to use the whole data by realignment.
- Our current implementation of jointly training acoustic model and vocoder seems very naive: (1) Data resource utilisation by vocoder in such a training paradigm is very inefficient. (2) Vocoder always receives predicted spectrogram. We believe (a)scheduling the vocoder input by starting from original spectrogram but slowly transitioning to predicted spectrogram (b) adopting wake sleep training procedure as opposed to the current implementation might lead to a better base architecture.
- We plan to further investigate the performance of sub sentence based training approach for acoustic model.

5. Conclusion

In this paper we have presented the entry from CMU to Blizzard speech synthesis challenge 2019. We have made these modifications to our previous submission: (1) We investigate the effectiveness of sub-sentence training of acoustic models aimed at better utilization of available data (2) We investigate the applicability of selective gradient backpropagation to accelerate the training (3) We experiment with iterated dilated convolutions in WaveNet to obtain compact models. We are actively pursuing approaches to strengthen our voice building framework. We believe we will have a much stronger framework and hence a more competitive submission in the coming evaluations.

6. References


