One slide for this course

Text-to-Speech (TTS) Synthesis

Model

Model

Model

Model

Model

class

class
Outline

- TTS before End-to-end
- Tacotron: End-to-end TTS
- Beyond Tacotron
- Controllable TTS
VODER (1939)

Source of video: https://www.youtube.com/watch?v=0rAyrmm7vv0

https://en.wikipedia.org/wiki/Voder
IBM computer (1960s)

• In 1961, John Larry Kelly Jr. using an IBM computer to synthesize speech at Bell lab.

Source of video and audio: https://youtu.be/UGsfwhb4-bQ
https://www.vintagecomputermusic.com/mp3/s2t9_Computer_Speech_Demonstration.mp3
Concatenative Approach

speeches from a large database

Source of image:
Parametric Approach

HMM/DNN-based Speech Synthesis System (HTS)

Source of image: http://hts.sp.nitech.ac.jp/?Tutorial
All the components are deep learning based.

Deep Voice

[Arik, et al., ICML’17]

Deep Voice 3 is end-to-end.

[Ping, et al., ICLR’18]
Outline

1. TTS before End-to-end
2. Tacotron: End-to-end TTS
3. Beyond Tacotron
4. Controllable TTS
Tacotron

[Wang, et al., INTERSPEECH’17]
[Shen, et al., ICASSP’18]

TACOTRON: TOWARDS END-TO-END SPEECH SYNTHESIS

Yuxuan Wang*, RJ Skerry-Ryan*, Daisy Stanton, Yonghui Wu, Ron J. Weiss†, Navdeep Jaitly,
Zongheng Yang, Ying Xiao*, Zhifeng Chen, Samy Bengio†, Quoc Le, Yannis Agiomyrgiannakis,
Rob Clark, Rif A. Saurous*

Google, Inc.
{yxwang,rjryan,rif}@google.com

*These authors really like tacos.
†These authors would prefer sushi.
Before Tacotron ...

• Tacotron:
  • Input: character
  • Output: (linear) spectrogram

• First Step Towards End-to-end Parametric TTS Synthesis
  [Wang, et al., INTERSPEECH’16]
  • Input: phoneme
  • Output: acoustic features for STRAIGHT (vocoder)

• Char2wav
  [Sotelo, et al., ICLR workshop’17]
  • Input: character
  • Output: acoustic features for SampleRNN (vocoder)
Tacotron

Encoder

Input embeddings

CBHG

Pre-net

Attention

Attention is applied to all decoder output

Decoder

Post-processing

CBHG

Vocoder

Hello!
**Encoder** = Grapheme-to-phoneme?

- **Pre-net**
  - CBHG
  - Conv1D layers
  - Highway layers
  - Bidirectional GRU
  - Conv1D projections
  - Max-pool along time
  - Conv1D bank + stacking

- **Input embeddings**: hello!

(v2)
Attention $\quad$ = Modeling Duration ?

- The output audio and input text must be monotonic aligned.
Using teacher forcing, but dropout acts like schedule sampling

Mel-spectrogram → Audio Synthesis

Decoder

Generating \( r \) frames each time

\( r = 1 \) in v2

Attention

dropout

Pre-net

zero vector
Using teacher forcing, but dropout acts like schedule sampling.
Post processing

Mel/Linear-spectrogram

Two loses

Mel-spectrogram

Non-causal

CBHG

generated by RNN-based decoder

Vocoder:
Griffin-Lim in v1
Wavnet in v2
How good is Tacotron?

<table>
<thead>
<tr>
<th>System</th>
<th>MOS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parametric</td>
<td>3.492 ± 0.096</td>
</tr>
<tr>
<td>Tacotron (Griffin-Lim)</td>
<td>4.001 ± 0.087</td>
</tr>
<tr>
<td>Concatenative</td>
<td>4.166 ± 0.091</td>
</tr>
<tr>
<td>WaveNet (Linguistic)</td>
<td>4.341 ± 0.051</td>
</tr>
<tr>
<td>Ground truth</td>
<td>4.582 ± 0.053</td>
</tr>
<tr>
<td>Tacotron 2 (this paper)</td>
<td>4.526 ± 0.066</td>
</tr>
</tbody>
</table>

Version 1
[Wang, et al., INTERSPEECH’17]

Version 2
[Shen, et al., ICASSP’18]
How good is Tacotron?

<table>
<thead>
<tr>
<th>System</th>
<th>MOS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tacotron 2 (Linear + G-L)</td>
<td>3.944 ± 0.091</td>
</tr>
<tr>
<td>Tacotron 2 (Linear + WaveNet)</td>
<td>4.510 ± 0.054</td>
</tr>
<tr>
<td>Tacotron 2 (Mel + WaveNet)</td>
<td>4.526 ± 0.066</td>
</tr>
</tbody>
</table>

WaveNet is much better than Griffin-Lim

<table>
<thead>
<tr>
<th>Training</th>
<th>Synthesis</th>
</tr>
</thead>
<tbody>
<tr>
<td>Predicted</td>
<td>Predicted</td>
</tr>
<tr>
<td>Predicted</td>
<td>4.526 ± 0.066</td>
</tr>
<tr>
<td>Ground truth</td>
<td>4.362 ± 0.066</td>
</tr>
</tbody>
</table>

WaveNet needs to be trained
Tip at Inference Phase

• You need dropout!

At inference time!

RNN

RNN

RNN

Pre-net

Pre-net

Pre-net

Attention

dropout

with dropout

without dropout

感謝杜濤同學提供實驗結果
用 Tacotron 做閩南語語音合成

中文 ➔ Translation? ➔ 台羅拼音

https://i3thuan5.github.io/tai5-uan5_gian5-gi2_kang1-ku7/index.html

感謝張凱為同學提供實驗結果

台羅拼音 ➔ Tacotron ➔

Source of training data: https://suisiann-dataset.ithuan.tw/

台灣媠聲2.0
Outline

TTS before End-to-end

Tacotron: End-to-end TTS

Beyond Tacotron

Controllable TTS
The raters considered ground truth is better than Tacotron 2 because …

“… occasional mispronunciation by our system is the primary reason …”

(Mispronunciation)

Mispronunciation

• Using a lexicon to transform word to phoneme, and using phoneme as Tacotron input
  • But lots of OOV words ...

“What is nCoV”

not in lexicon???

Input of Tacotron

• Character and phoneme hybrid input [Ping, et al., ICLR’18]

If the pronunciation of machine is incorrect, one can add the word into the lexicon to fix the problem.
More information for Encoder

• Syntactic information
  [Guo, et al., INTERSPEECH’19]

[Figure 1: An example of syntactically parsed tree]

一日，小龍女對楊過說：
「我也想過過過兒過過的生活」

Source of example:
https://youtu.be/kptTHjBi_ak

• BERT embedding as input
  [Hayashi, et al., INTERSPEECH’19]
Attention

• **Guided Attention** [Tachibana, et al., ICASSP’18]

Penalizing the non-diagonal attention matrix during training
Attention

• Monotonic Attention
  [Raffel, et al., ICML’17]

• Location-aware attention
  (Have been mentioned when we talked about ASR)
More Attention ...
More ...

Attention matrix

Only attend in a fixed window

(constraint at inference)

Constraint attention by positional encoding

[Ping, et al., ICLR’18]
Fast Speech

[Ren, et al., NeurIPS’19]

How to train this model?

Duration Informed Attention Network (DurIAN) [Yu, et al, arXiv’19]
Fast Speech

[Ren, et al., NeurIPS’19]

During the **training** phase:

Using ground truth (alignment from another model?)

Duration Informed Attention Network (DurIAN)  [Yu, et al, arXiv’19]
Fast Speech

In 50 sentences:

<table>
<thead>
<tr>
<th>Method</th>
<th>Repeats</th>
<th>Skips</th>
<th>Error Sentences</th>
<th>Error Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tacotron 2</td>
<td>4</td>
<td>11</td>
<td>12</td>
<td>24%</td>
</tr>
<tr>
<td>Transformer TTS</td>
<td>7</td>
<td>15</td>
<td>17</td>
<td>34%</td>
</tr>
<tr>
<td>FastSpeech</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0%</td>
</tr>
</tbody>
</table>

zero zero zero zero zero zero zero zero zero zero zero two seven nine eight F three forty zero zero zero zero zero zero six four two eight zero one eight
c five eight zero three three nine a zero bf eight FALSE zero zero zero bba3add2 - c229 - 4cdb -
Calendar agent failed with error code 0x80070005 while saving appointment.
Exit process - break ld - Load module - output ud - Unload module - ignore ser - System error -
ignore ibp - Initial breakpoint -
h t t p colon slash slash teams slash sites slash T A G slash default dot aspx As always , any
feedback , comments ,
two thousand and five h t t p colon slash slash news dot com dot com slash i slash n e slash f d slash two zero zero three slash f d
Dual Learning: ASR & TTS

ASR & TTS form a cycle.

Speech Chain
[Tjandra et al., ASRU 2017]
Dual Learning: TTS v.s. ASR

- Given pretrained TTS and ASR system

![Diagram showing the flow of speech and text between TTS and ASR systems.](Diagram.png)
Dual Learning: TTS v.s. ASR

• Experiments

1600 utterance-sentence pairs
7200 unpaired utterances and sentences

Table 2: Experiment result for multi-speaker test set.

<table>
<thead>
<tr>
<th>Data</th>
<th>Hyperparameters</th>
<th>ASR</th>
<th>TTS</th>
<th>mce</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>α</td>
<td>β</td>
<td>gen. mode</td>
<td>CER (%)</td>
</tr>
<tr>
<td>Paired (80 utt/spk)</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>26.47</td>
</tr>
<tr>
<td>+ Unpaired (remaining)</td>
<td>0.25</td>
<td>1</td>
<td>greedy</td>
<td>23.03</td>
</tr>
<tr>
<td></td>
<td>0.5</td>
<td>1</td>
<td>greedy</td>
<td>20.91</td>
</tr>
<tr>
<td></td>
<td>0.25</td>
<td>1</td>
<td>beam 5</td>
<td>22.55</td>
</tr>
<tr>
<td></td>
<td>0.5</td>
<td>1</td>
<td>beam 5</td>
<td>19.99</td>
</tr>
</tbody>
</table>

Mel: mel-spectrogram
Raw: raw waveform

[Tjandra et al., ASRU 2017]
Outline

TTS before End-to-end

Tacotron: End-to-end TTS

Beyond Tacotron

Controllable TTS
Can we control these factors?

Control by input text

Who is speaking

What is being said

How is it said

Speech
Controllable TTS

• 誰在說？
  • Synthesize speech for a specific person (voice cloning)
  • Lack of high quality single speaker data to train a speech synthesis system

• 怎麼說？
  • Intonation (語調), stress (重音), rhythm (韻律) ...
  • Prosody (抑揚頓挫)

How to describe prosody?

**Definition.** Prosody is the variation in speech signals that remains after accounting for variation due to phonetics, speaker identity, and channel effects (i.e. the recording environment).

[Skerry-Ryan, et al., ICML’18]
Controllable TTS

Controllable TTS

Audio to be converted (provide content)

Reference audio ("say it like this")

VC Model

Reference audio ("say it like this")

Voice Conversion (VC)
Controllable TTS

**Training**

- Input: Text
- Output: Reference Audio

**Inference**

- Input: Text
- Output: Generated Audio

---

TTS Model

Reference audio

I love you

(with the style of “I love you”)
Controllable TTS

**Training**

- **Ignore text**
- **Minimize reconstruction error**

**Inference**

- **I love you (Reference audio)**

**TTS Model**

- **hello**
- **I love you**

- **L1 copy**
Speaker Embedding

[jia, et al., NeurIPS’18]

Training

Pre-trained network (fix)

Feature Extractor

Minimize reconstruction error

TTS Model

speaker embedding
GST-Tacotron

GST = global style tokens
[Wang, et al., ICML’18]
GST-Tacotron

output only one vector

Reference audio

Feature Extractor

vector set
(learned)

attention
weight

A

B

C

D

0.2

0.1

0.3

0.4

Style Tokens

Reference audio

Encoder
• What does the tokens effect?
  • One token corresponds to a lower pitch voice
  • One token for a decreasing pitch
  • One token for a faster speaking rate
  • ……

Two-stage Training

Training

2nd stage training

[Liu, et al., SLT’18]
Two-stage Training

minimize recognition error

TTS

ASR

attention consistency

reference audio

[Liu, et al., SLT’18]
Concluding Remarks

- TTS before End-to-end
- Tacotron: End-to-end TTS
- Beyond Tacotron
- Controllable TTS
Reference


• [Shen, et al., ICASSP’18] Jonathan Shen, Ruoming Pang, Ron J. Weiss, Mike Schuster, Navdeep Jaitly, Zongheng Yang, Zhifeng Chen, Yu Zhang, Yuxuan Wang, RJ Skerry-Ryan, Rif A. Saurous, Yannis Agiomyrgiannakis, Yonghui Wu, Natural TTS Synthesis by Conditioning WaveNet on Mel Spectrogram Predictions, ICASSP, 2018


Reference


• [Zhang, et al., arXiv’19] Liqiang Zhang, Chengzhu Yu, Heng Lu, Chao Weng, Yusong Wu, Xiang Xie, Zijin Li, Dong Yu, Learning Singing From Speech, arXiv, 2019

• [Guo, et al., INTERSPEECH’19] Haohan Guo, Frank K. Soong, Lei He, Lei Xie, Exploiting Syntactic Features in a Parsed Tree to Improve End-to-End TTS, INTERSPEECH, 2019


• [Skerry-Ryan, et al., ICML’18] RJ Skerry-Ryan, Eric Battenberg, Ying Xiao, Yuxuan Wang, Daisy Stanton, Joel Shor, Ron J. Weiss, Rob Clark, Rif A. Saurous, Towards End-to-End Prosody Transfer for Expressive Speech Synthesis with Tacotron, ICML, 2018
Reference


• [Ren, et al., NeurIPS’19] Yi Ren, Yangjun Ruan, Xu Tan, Tao Qin, Sheng Zhao, Zhou Zhao, Tie-Yan Liu, FastSpeech: Fast, Robust and Controllable Text to Speech, NeurIPS, 2019


• [Raffel, et al., ICML’17] Colin Raffel, Minh-Thang Luong, Peter J. Liu, Ron J. Weiss, Douglas Eck, Online and Linear-Time Attention by Enforcing Monotonic Alignments, ICML, 2017
Reference

• [Tachibana, et al., ICASSP’18] Hideyuki Tachibana, Katsuya Uenoyama, Shunsuke Aihara, Efficiently Trainable Text-to-Speech System Based on Deep Convolutional Networks with Guided Attention, ICASSP, 2018


• [Liu, et al., SLT’18] Da-Rong Liu, Chi-Yu Yang, Szu-Lin Wu, Hung-Yi Lee, "Improving Unsupervised Style Transfer in End-to-End Speech Synthesis with End-to-End Speech Recognition", SLT, 2018