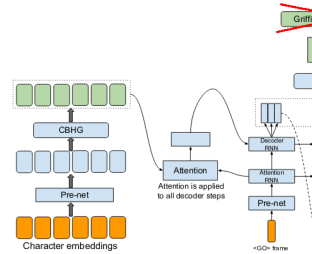

WaveRNN



(Update: Vanilla Tacotron One TTS system just implemented - more coming soon!)

Pytorch implementation of Deepmind's WaveRNN model from Efficient Neural Audio Synthesis

Installation

Ensure you have:

- Python ≥ 3.6
- Pytorch 1 with CUDA

Then install the rest with pip:

```
pip install -r requirements.txt
```

How to Use

Quick Start

If you want to use TTS functionality immediately you can simply use:

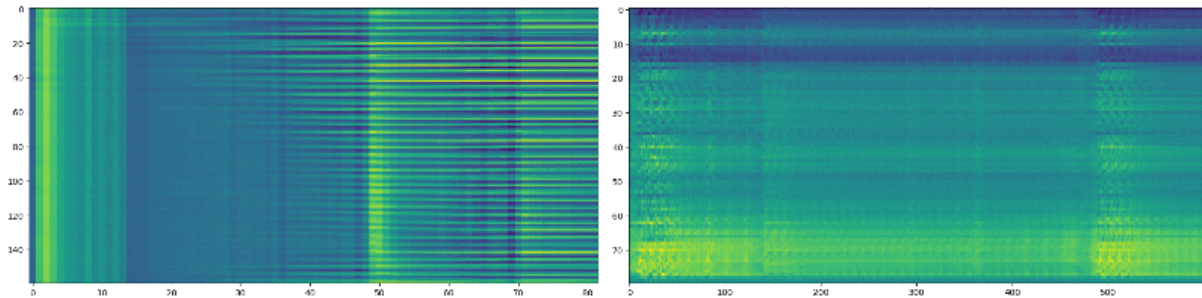
```
python quick_start.py
```

This will generate everything in the default sentences.txt file and output to a new 'quick_start' folder where you can playback the wav files and take a look at the attention plots

You can also use that script to generate custom tts sentences and/or use '-u' to generate unbatched (better audio quality):

```
python quick_start.py -u -input_text "What will happen if I run this command?"
```

Training your own Models



Download the LJSpeech Dataset.

Edit **hparams.py**, point **wav_path** to your dataset and run:

```
python preprocess.py
```

or use `preprocess.py -path` to point directly to the dataset ____

Here's my recommendation on what order to run things:

1 - Train Tacotron with:

```
python train_tacotron.py
```

2 - You can leave that finish training or at any point you can use:

```
python train_tacotron.py -force_gta
```

this will force tacotron to create a GTA dataset even if it hasn't finish training.

3 - Train WaveRNN with:

```
python train_wavernn.py -gta
```

NB: You can always just run `train_wavernn.py` without `-gta` if you're not interested in TTS.

4 - Generate Sentences with both models using:

```
python gen_tacotron.py wavernn
```

this will generate default sentences. If you want generate custom sentences you can use

```
python gen_tacotron.py -input_text "this is whatever you want it to be" wavernn
```

And finally, you can always use `-help` on any of those scripts to see what options are available :)

Samples

Can be found here.

Pretrained Models

Currently there are two pretrained models available in the `/pretrained/` folder':

Both are trained on LJSpeech

- WaveRNN (Mixture of Logistics output) trained to 800k steps
- Tacotron trained to 180k steps

References

- Efficient Neural Audio Synthesis
- Tacotron: Towards End-to-End Speech Synthesis
- Natural TTS Synthesis by Conditioning WaveNet on Mel Spectrogram Predictions

Acknowledgements

- <https://github.com/keithito/tacotron>
- https://github.com/r9y9/wavenet_vocoder
- Special thanks to github users G-Wang, geneing & erogol