Artificial Intelligence Pres & Future
PROJECT PROPOSAL

Name of the individuals
Name1: SHIVAVARDHAN BANTU
Name2: VINEET REDDY SANVELLI
Name3: NITHEESHA REDDY
Name4: TRIBHUVAN REDDY MIDDELA

Due: October 11th, 2022 @ 11:59 pm (ET)
Prof. REDA NACIF EL ALAOUI

TOPIC OF INTEREST: TEXT - TO - SPEECH RECOGNITION

Introduction & Motivation

TTS is a computer simulation of human speech from a textual representation using machine learning methods. Typically, speech synthesis is used by developers to create voice robots, such as IVR (Interactive Voice Response).

TTS saves a business time and money as it generates sound automatically, thus saving the company from having to manually record (and rewrite) audio files.

You can have any text read aloud in a voice that is as close to natural as possible, thanks to TTS synthesis. To make TTS synthesized speech sound natural, the painstaking process of honing its timbre, smoothness, placement of accents and pauses, intonation, and other areas is a long and unavoidable burden.

There are two ways developers can go about getting it done:

Concatenative - gluing together fragments of recorded audio. This synthesized speech is of high quality but requires a lot of data for machine learning.

Parametric - building a probabilistic model that selects the acoustic properties of a sound signal for a given text. Using this approach, one can synthesize a speech that is virtually indistinguishable from a real human.

Related Work

The Machine Learning domain of Audio is definitely at the cutting edge right now. A majority of the applications that products offer today are proprietary. There are many audio-specific open-source frameworks and algorithms that the community is developing. My goal over the next few articles is to give a deeper dive into some practical end-to-end uses of Audio. From Speech to Text and Text To Speech to Voice Cloning. As the title mentioned, this post will focus on a simple implementation of Text To Speech (Speech synthesis)

Text To Speech Architecture Types

We must understand the different types of architectures that we can utilize to synthesize speech along with the current evolution.

Concatenative — Old School

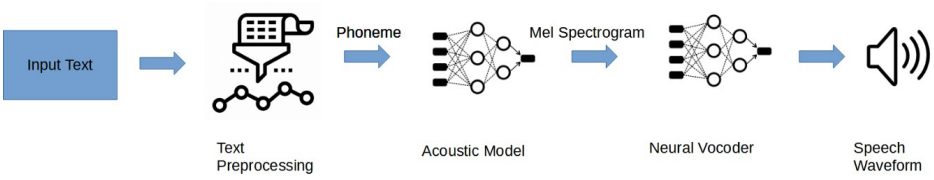
Traditional old school technique that uses a stored speech database where speech is mapped to specific words. While for certain mapped words, you can produce understandable Audio, the output speech will not include the natural sounds of voice, "prosody, emotion, etc.."

Mainstream 2-Stage:

A hybrid parametric TTS approach that relies on a Deep Neural Network consisting of an acoustic model and neural vocoder to approximate the parameters and relationship between input text and the waveform that make up speech.

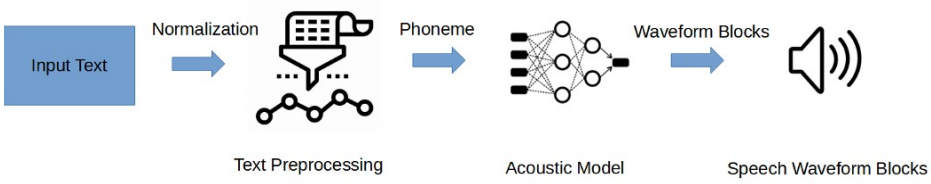
A basic high-level overview of mainstream 2-Stage TTS System

Next Generation End-to-End Text to



Wave Model:

The recent papers in Audio TTS are heading in this direction. Utilizing a single acoustic model that doesn't output Mel-spectrograms that feed a neural vocoder.



Approach

Step 1: Install from terminal or through Jupyter notebook with the prefix (!)

Step 2: Download a Pre-Trained Acoustic Model and Neural Vocoder

Step 3: Model Setup

Initialize your ESPnet Model with the selected pretrained acoustic model and neural vocoder (if selected). There are some hyperparameters to tune for some acoustic algorithms, but we'll get more into that in the next post. For now use the default.

Step 4: Speech Synthesis

Hopefully, this part speaks for itself, but simply place whatever text you wish to transform into beautiful Audio!

Source code will be using

Language chosen: Python
Algorithms Used: ML Algorithms, viterbi search
Libraries Used: pyttsx3

Evaluation

For objective evaluations the most popular test is simple MCD test (mel cepstral distortion), but there are more advanced ones.

Milestones/Timelines

The following table shows the milestones with their corresponding expected dates for achievement:

Dates (Weeks)	Describe the Objectives and roadmap actions
October 11th	Topic selection and proposal submission
October 11th	Finish the literature review
October 11th	Data acquisition and pre-processing
November 15th	Report project progress
December 13th	The final presentation of project results & discussion

References

1. Authors and Contributors of ESPnet
2. End-to-End Speech Processing Toolkit,
3. https://www.youtube.com/watch?v=2mRz3wH1vd0&ab_channel=WAVLab
4. https://theaisummer.com/text-to-speech/
5. https://www.cs.mcgill.ca/~rwest/wikispeedia/wpcd/wp/s/Speech_synthesis.htm#:~:text=The%20first%20computer%2Dbased%20speech,the%20history%20of%20Bell%20Labs.