Coqui-ai TTS library => https://github.com/coqui-ai/TTS

A deep learning toolkit for Text-to-speech conversion.

Requirements: (Windows)

Python version--3.8

pip

espeak-ng

Steps to run the library: (Windows)

Step 1: create a python virtual env

python -m venv .

Step 2: Activate the venv

cd Scripts

./activate

```
C:\Users\ahema\Desktop>cd TTS

C:\Users\ahema\Desktop\TTS>cd Scripts

C:\Users\ahema\Desktop\TTS\Scripts> .\activate
```

```
C:\Users\ahema\Desktop\TTS\Scripts> .\activate

(TTS) C:\Users\ahema\Desktop\TTS\Scripts>python --version
Python 3.8.10
```

Step 3: update the wheel

pip list, pip install pip setuptools wheel -U

Step-4: Install TTS library

pip install TTS==0.8.0

Step-5: Install pyworld

pip install pyworld==0.3.0

Step-6: check TTS installation

tts-server --list_models

```
(TTS) C:\Users\ahema\Desktop\TTS\Scripts>tts-server -h
usage: tts-server.exe [-h] [--list_models [LIST_MODELS]] [--model_name MODEL_NAME] [--vocoder_name VOCODER_NAME] [--config_path CONFIG_PATH] [--model_path MODEL_PATH] [--vocoder_path VOCODER_PATH]
                      [--vocoder_config_path VOCODER_CONFIG_PATH] [--speakers_file_path SPEAKERS_FILE_PATH] [--port PORT] [--use_cuda USE_CUDA] [--debug DEBUG] [--show_details SHOW_DETAILS]

optional arguments:
  -h, --help            show this help message and exit
  --list_models [LIST_MODELS]
                        list available pre-trained tts and vocoder models.
  --model_name MODEL_NAME
                        Name of one of the pre-trained tts models in format <language>/<dataset>/<model_name>
  --vocoder_name VOCODER_NAME
                        name of one of the released vocoder models.
  --config_path CONFIG_PATH
                        Path to model config file.
  --model_path MODEL_PATH
                        Path to model file.
  --vocoder_path VOCODER_PATH
                        Path to vocoder model file. If it is not defined, model uses GL as vocoder. Please make sure that you installed vocoder library before (WaveRNN).
  --vocoder_config_path VOCODER_CONFIG_PATH
                        Path to vocoder model config file.
  --speakers_file_path SPEAKERS_FILE_PATH
                        JSON file for multi-speaker model.
  --port PORT           port to listen on.
  --use_cuda USE_CUDA   true to use CUDA.
  --debug DEBUG         true to enable Flask debug mode.
  --show_details SHOW_DETAILS
                        Generate model detail page.
```

```
(TTS) C:\Users\ahema\Desktop\TTS\Scripts>tts-server --list_models
 Name format: type/language/dataset/model
 1: tts_models/multilingual/multi-dataset/your_tts [already downloaded]
 2: tts_models/en/ek1/tacotron2 [already downloaded]
 3: tts_models/en/ljspeech/tacotron2-DDC [already downloaded]
 4: tts_models/en/ljspeech/tacotron2-DDC_ph
 5: tts_models/en/ljspeech/glow-tts
 6: tts_models/en/ljspeech/speedy-speech
 7: tts_models/en/ljspeech/tacotron2-DCA
 8: tts_models/en/ljspeech/vits [already downloaded]
 9: tts_models/en/ljspeech/fast_pitch
 10: tts_models/en/vctk/vits
 11: tts_models/en/vctk/fast_pitch
 12: tts_models/en/sam/tacotron-DDC
 13: tts_models/en/blizzard2013/capacitron-t2-c50
 14: tts_models/en/blizzard2013/capacitron-t2-c150
 15: tts_models/es/mai/tacotron2-DDC
 16: tts_models/fr/mai/tacotron2-DDC [already downloaded]
 17: tts_models/uk/mai/glow-tts
 18: tts_models/zh-CN/baker/tacotron2-DDC-GST
 19: tts_models/nl/mai/tacotron2-DDC
 20: tts_models/de/thorsten/tacotron2-DCA
 21: tts_models/de/thorsten/vits
 22: tts_models/de/thorsten/tacotron2-DDC [already downloaded]
 23: tts_models/ja/kokoro/tacotron2-DDC [already downloaded]
 24: tts_models/tr/common-voice/glow-tts
 25: tts_models/it/mai_female/glow-tts
 26: tts_models/it/mai_female/vits [already downloaded]
 27: tts_models/it/mai_male/glow-tts
 28: tts_models/it/mai_male/vits
 29: tts_models/ewe/openbible/vits
 30: tts_models/hau/openbible/vits
 31: tts_models/lin/openbible/vits
 32: tts_models/tw_akuapem/openbible/vits
 33: tts_models/tw_asante/openbible/vits
 34: tts_models/yor/openbible/vits
 1: vocoder_models/universal/libri-tts/wavegrad
 2: vocoder_models/universal/libri-tts/fullband-melgan [already downloaded]
 3: vocoder_models/en/ek1/wavegrad [already downloaded]
 4: vocoder_models/en/ljspeech/multiband-melgan
 5: vocoder_models/en/ljspeech/hifigan_v2 [already downloaded]
 6: vocoder_models/en/ljspeech/univnet
 7: vocoder_models/en/blizzard2013/hifigan_v2
 8: vocoder_models/en/vctk/hifigan_v2
 9: vocoder_models/en/sam/hifigan_v2
 10: vocoder_models/nl/mai/parallel-wavegan
 11: vocoder_models/de/thorsten/wavegrad
 12: vocoder_models/de/thorsten/fullband-melgan
```

play with the library:

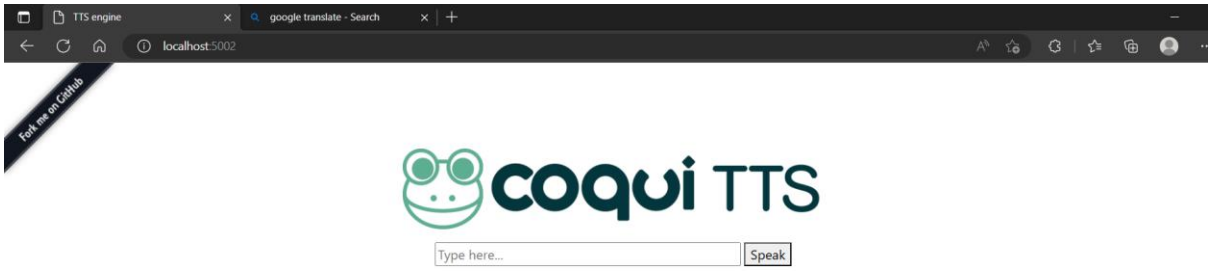
converts the text to audio using the default model

tts --text "Hello world"

creates a local web interface to interact and convert the text to audio

tts-server --model_name tts_models/en/ljspeech/tacotron2-DDC

```
 | > trim_db:60
 | > do_sound_norm:False
 | > do_amp_to_db_linear:True
 | > do_amp_to_db_mel:True
 | > do_rms_norm:False
 | > db_level:None
 | > stats_path:None
 | > base:2.718281828459045
 | > hop_length:256
 | > win_length:1024
 > Generator Model: hifigan_generator
 > Discriminator Model: hifigan_discriminator
Removing weight norm...
 * Serving Flask app 'TTS.server.server'
 * Debug mode: off
INFO:werkzeug:WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
 * Running on all addresses (::)
 * Running on http://[::1]:5002
 * Running on http://[::1]:5002
INFO:werkzeug:Press CTRL+C to quit
```




```
 * Serving Flask app 'TTS.server.server'
 * Debug mode: off
INFO:werkzeug:WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI
 * Running on all addresses (::)
 * Running on http://[::1]:5002
 * Running on http://[::1]:5002
INFO:werkzeug:Press CTRL+C to quit
INFO:werkzeug:::1 - - [02/Apr/2023 22:08:45] "GET / HTTP/1.1" 200 -
INFO:werkzeug:::1 - - [02/Apr/2023 22:08:46] "GET /static/coqui-log-green-TTS.png HTTP/1.1" 304 -
 > Model input: Hello
 > Speaker Idx:
 > Text splitted to sentences.
['Hello']
   > Decoder stopped with `max_decoder_steps` 10000
 > Processing time: 69.00391721725464
 > Real-time factor: 0.5914483528626989
INFO:werkzeug:::1 - - [02/Apr/2023 22:10:19] "GET /api/tts?text=Hello&speaker_id=&style_wav= HTTP/1.1" 200 -
INFO:werkzeug:::1 - - [02/Apr/2023 22:10:19] "GET /favicon.ico HTTP/1.1" 404 -
```