28/02/2022

Created an Ubuntu based virtual machine to attempt to install and run MyCroft AI Speech Recognition

- https://mycroft-ai.gitbook.io/docs/using-mycroft-ai/get-mycroft/macos-and-windows-with-virtualbox

Implemented ArgParse into the script to assist in creating and parsing a list of arguments that are checked at runtime

- https://docs.python.org/3/library/argparse.html

While attempting to find out what mycroft is and how it can be implemented effectively, I found out that it runs on DeepSpeech. DeepSpeech is an AI based speach to text service that can be installed and run on a local machine, or deployed onto a server.

I found a tutorial at 'https://www.slanglabs.in/blog/how-to-build-python-transcriber-using-mozilla-deepspeech' which covers setting up and cofiguring DeepSpeak, in various ways including:

passing an audio file to a local instance to process.

integration into a python script which allows streaming files and microphone input

Sending information to a server instance that processes the sound data and returns the result as text.

It should be noted that I couldn't get the implementation in this tutorial to work correctly. But I did further research and found the git mentioned below, which is for an instance of a server that waits for a file to process.

The following link (https://pypi.org/project/deepspeech-server/) is a python based DeepSpeech server that is configurable to run on any http port and recieves requests via HTTP post commands. I found the a git repo that hosts it (listed below). Using this version allows me to change the version of DeepSpeak it uses.

I downloaded an instance of a DeepSpeak server from https://github.com/MainRo/deepspeech-server

After running the 'setup.py' script, the config needed to be set. There were template configs, that were copied and modified to set DeepSpeak instance configuration and server information.

the server could be run with deepspeech-server --config config.json

Imported DeepSpeech into a test python script and created tests to compare the speed of the following DeepSpeech Communication Method:

-Passing a sound file to a local instance of DeepSpeak installed through the 'apt-get' command

-Stream a sound file to DeepSpeech to simulate how a microphone should function.

-Run a local server that can process DeepSpeech data communicated over the HTTP protocol.

After running these tests, it was concluded that running a local linux server (or through WSL on Windows 10 or 11) was the fastest way of processing sound data.

All results can be found in the file 'results.xls'