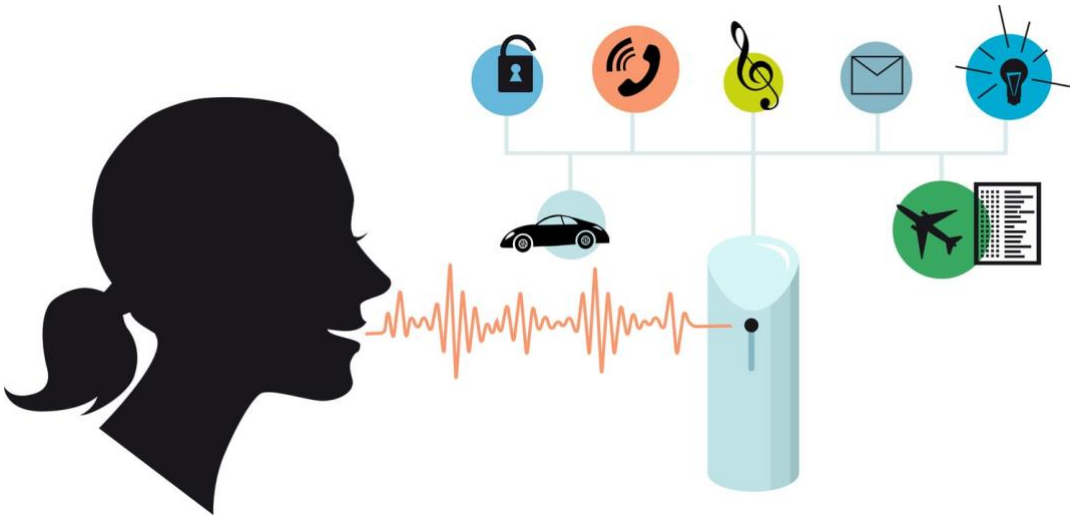
Capstone Proposal

Speech Recognition & Speech-To-Text



Proposed by: Tristin Johnson

tristinjohnson@gwu.edu

Advisor: Dr. Amir Jafari

The George Washington University, Washington, DC

M.S. in Data Science

1. Objective

The goal of this project is to build and develop a pretrained speech recognition engine. The focus of this speech engine is to have the ability to transcribe any given audio file to its pertained text. When it comes to speech-to-text engines, these types of systems are extremely efficient, cheap, and convenient to multiple different environments including students in the classroom, business meetings, personal use, and more. Furthermore, speech-to-text software can boost productivity by providing the ability to simple record any voice and have it written to a document compared to recording notes by hand, which can be time consuming. The speech engine we want to develop will have the ability to do the following:

- Transcriptions that will work for all gender, ages, ethnicity, etc.
- Pre-train a transformer model from scratch using two of the following: UniSpeech, UniSpeechSat, WavLM, Wav2Vec, Wav2VecPhoneme, Speech2Text
- Use the above transformer to transcribe any given audio file from different problem sets

2. Dataset

There are multiple different open-source datasets that can be used for this project:

- Common Voice - English (2015 hours of voice audio, 75,879 different voices, 65 GB)
- TED-LIUM (452 hours of voice audio, 2351 different TED Talks, 54 GB)
- LibriSpeech (~1000 hours of audiobooks, 57.2 GB)

Initially, we have decided to use the LibriSpeech dataset to pretrain a model and build the speech engine. Once we have an accurate and reliable model, the Common Voice and TED-LIUM datasets may be used to test accuracy and diversity of the model, depending on the allocated resources.

3. Rationale

This project has multiple different real-world applications in which a generated speech engine could be applied to. Developing this type of engine can lead to multiple other in-depth research projects for any researchers out there trying to create a speech-to-text system. For this project specifically, the focus is to create an engine to support students in the classroom, especially ones with learning disabilities.

Speech-to-text technology allows student to easily transfer their ideas onto a page, having the ability to talk ideas through and have all the ideas show up on a document, rather than struggling to get any ideas written down, which allows students to focus on the content rather than the act of reading, which results in a better understanding of the material. Furthermore, this also helps with students who have ADHD and other processing-related disabilities. Speech-to-text tools also save massive amounts of time. This is helpful for students who might forget their ideas once they try to write or students who struggle with getting any words on the page at all. For some students, there is an intimidation factor of writing academically, such as spelling and grammar anxieties preventing them from the start. This is what we set out to achieve, an automatic speech recognition engine to help students focus more on the content of courses rather than the stress of having to write everything down by hand.

4. Approach

I plan on approaching the capstone project through several different steps:

- Gathering the data and speech data wrangling
- Speech data preprocessing
- Modeling with different transformers
- Transcriber heads
- Analysis

5. Timeline

This is a rough timeline for the capstone project:

- (2 weeks) – Gathering the data and speech data wrangling
- (2 weeks) – Speech data preprocessing
- (4 to 6 weeks) – Modeling with different transformers mentioned above
- (1 to 2 weeks) – Transcriber heads
- (1 week) – Analysis
- (1 week) – Write research paper and submission
- (1 week) – Final Presentation
- (Bonus if enough time) – User Interface (Website)

6. Possible Issues

One of the main possible issues with this project is the data preprocessing. All the datasets are large (all over 50 GBs with hundreds of hours of audio), so preprocessing may take more time than expected. Furthermore, working with big data to train an accurate model could also be a potential issue, especially if certain resources aren't available (Cloud Services, GPUs, etc.).