Road Scene Understanding for the Visually Impaired (RSU-VI)

Debanjan Chakraborty

Pattern Recognition Lab

FAU

Outline

- Introduction to the *Voice Assistance System*
- Overview of Speech Recognition
- Speech to Text and Text to Speech
- Challenges
- **FINALLY, WE FIGURED IT!!!**
- Text To Speech
- Conclusion
- Future Works
- Demo

Introduction

What is a voice assistant?

Examples: Siri, Alexa

What is Voice Assistant doing in our project :

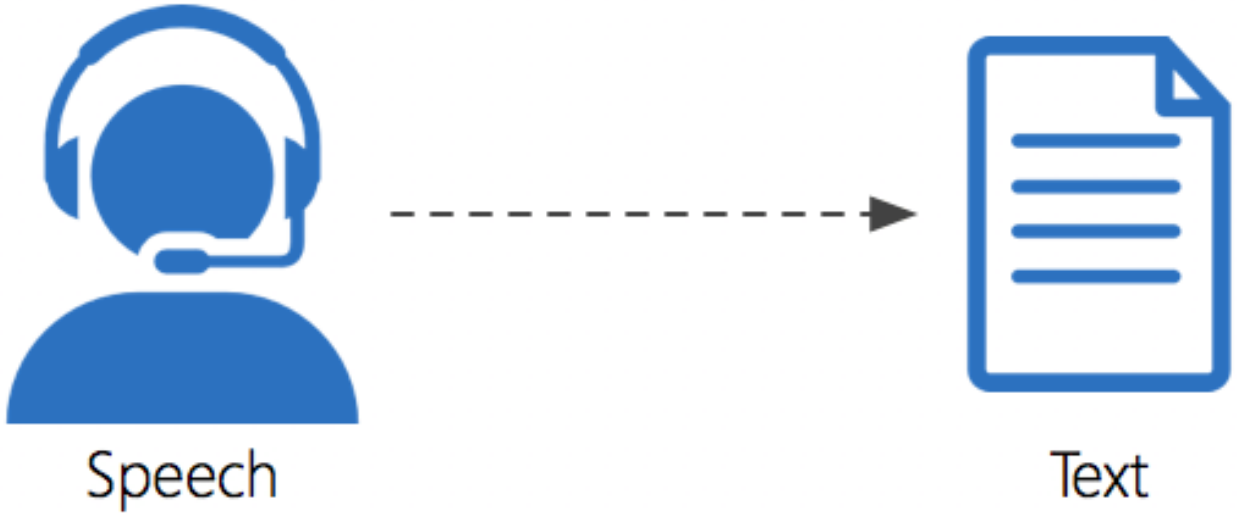
- In earlier Implementation
- Enhancing Natural Language Processing (NLP)

Advantages of Voice Assistance in BVIP(Blind or Visually impaired person)

- Voice-Activated Commands
- Real-Time Guidance
- Points of Interest Identification
- Accessibility and Safety
- Customization and Personalization
- Integration with Other Accessibility Features

Speech to Text

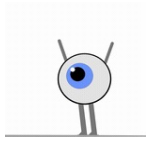
- Numerous technologies for STT.
- How does it function ?
- Where do we have its uses.
- What will this help us in.



Text to Speech

- TTS converts written text into spoken words
- It uses natural language processing to understand the text and generate natural-sounding speech.
- TTS systems employ voice synthesis techniques to produce the spoken output.
- TTS has applications in accessibility, navigation systems, virtual assistants, and more.

Challenges



Operating Systems

Needed VM setup as none of our systems were Linux and CIP pools gives only 2GB data limit

MacOS:

- Silicon Processor of MacOS
- VM – Qemu with UTM

Windows:

- Dependencies for different ASR tools were not same as linux.
- VM setup and Microphone issue resolved

ASR Tools

Kaldi:

- One of the important dependencies – Python 2.7
- About 40 GB of space is required
- Huge setup time
- Not accurate speech recognition

Speech recognition:

- Background noise
- Performance based on accent, dialect and speaking speed
- Homophones - ‘two’ and ‘to’

ASR Tools

Sphinx:

- Reduced accuracy in complex acoustic environment, noisy recordings, accents etc
- Adaptation to speaker variability
- Difficult to handle complex sentences
- Complex Fine-tuning
- Bad output speech quality

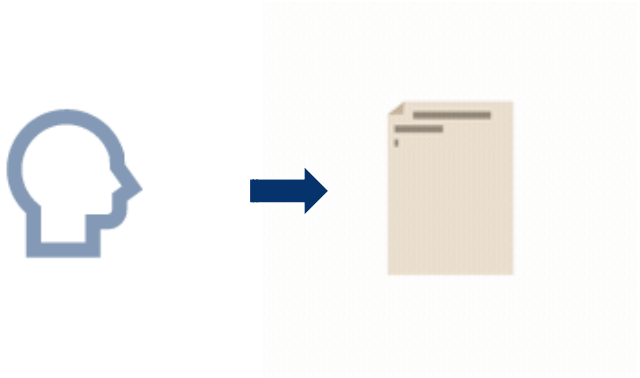
HuggingFace :

- Limited language selection
- The data capturing was not accurate
- Download different dataset

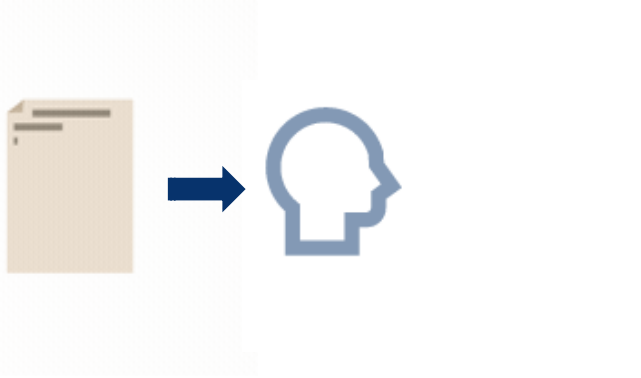
When ASR became a battle of giants

Finally, we figured it out !!

OpenAI / Whisper



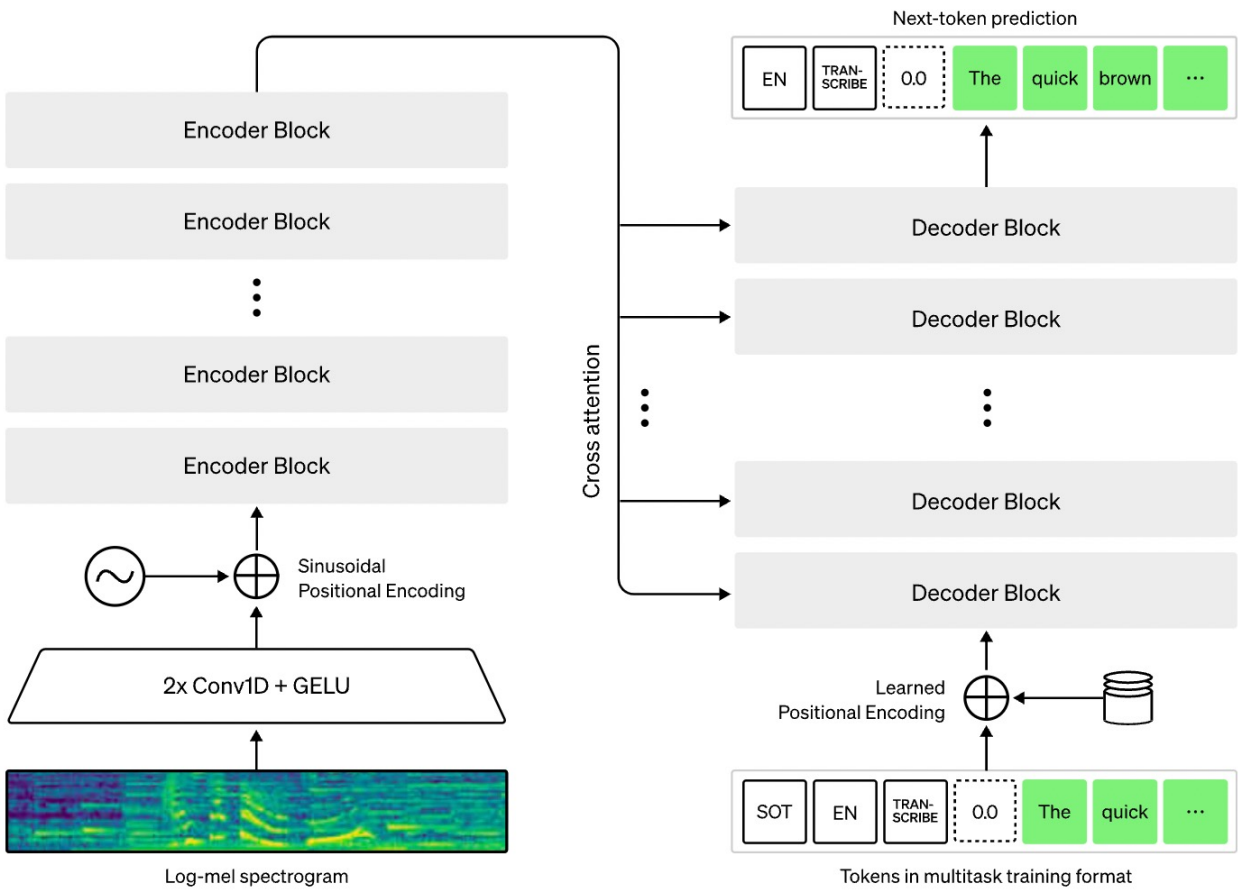
PYTTSX3



Open AI Whisper – an Overview

- According to OpenAI, Whisper is a general-purpose speech(**automatic voice recognition system)** recognition model
- It is trained on a large dataset of diverse audio
- It is also a multitasking model that can perform multilingual speech recognition, speech translation, and language identification
- OpenAI Whisper is very fast.

Architecture of Whisper



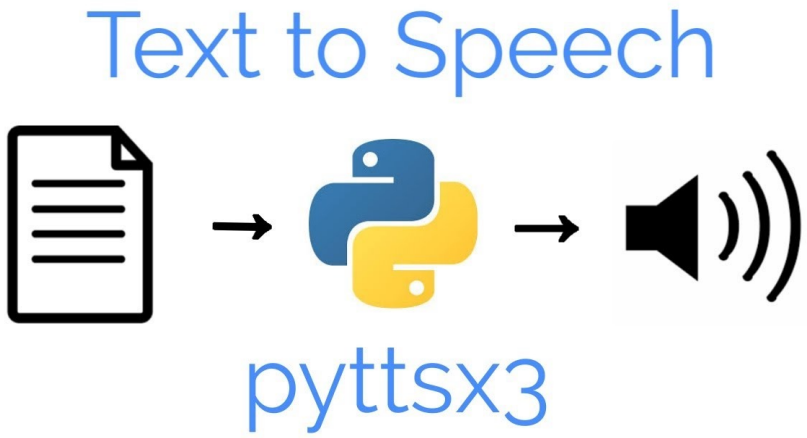
- Source: https://openai.com/research/whisper

Pyttsx3

Pyttsx3 is a Python library that provides a simple yet powerful interface for performing text-to-speech conversion.

Here's an overview of pyttsx3's key features:

- Easy installation
- Platform independence
- Multiple speech engines
- Voice customization
- Compatibility with Python versions



Conclusion

- Improved usability and functionality of the system for blind or visually impaired persons (BVIPs).
- Hands-free interaction to start , stop or shutdown the system using speech input
- Can be integrated with Jetson Nano as this is supported in all 3 OS – Windows, MacOS, Linux
- Space required is around 30 MB which is very less compared to Kaldi which required about 40 GB

Future scope

1. Future work in Sensation

    a. Integration with sensation/ chest box.
    b. Further implementation of code to handle different def according to requirement.
    c. Training the system to understand natural language - few things are hard coded right now.

2. Fine tuning.
3. Auto –detection of language.
4. Multi model implementation of Whisper.

Thank You

Team 1

Pattern Recognition Lab

FAU