Project Plan Presentation
On-Premises ASR Pipeline for Michigan English
The Capstone Experience

Team Michigan State University Linguistics
Eden Seo
Jacob Caurdy
Maria Irimie
Kyle Reinhart
Yichen Ding

Department of Computer Science and Engineering
Michigan State University
Spring 2022
MSU LiLaC

• MSU Linguistic, Language, and Cultures (LiLaC)
  • Offers degree programs and research in linguistics
• MI Diaries Project done by the Sociolinguistics Lab
  • Aims to chronicle language changes over the course of the pandemic
  • Aims to provide primary sources on the pandemic to historians as well
• Takes volunteer audio
Functional Specifications

• Creating an Automatic Speech Recognition (ASR) to fit into the current pipeline for the MSU Linguistics

• Replacing Google's role in the pipeline
  ▪ Saves money
  ▪ Protects data
  ▪ Improves accuracy with dialectal differences

• ASR

• Speaker Diarization (speaker differentiation)
  ▪ Time-aligned transcript

• Model Retraining
  ▪ Handling inaccuracies
Design Specifications

• Audio Upload
• Hand-Correction Interface
• Sensitive Information Flagging
• Diarization View
• Retraining
Screen Mockup: Transcription
Screen Mockup: Transcription (2)
Screen Mockup: Diarization View
Screen Mockup: Retraining
Technical Specifications

• ASR Pipeline consists of two processes
  ▪ Training and Prediction

• Training

• Prediction
  ▪ Pre-processing, prediction, and post-processing
  ▪ Combines both models

• Docker
  ▪ Portability
  ▪ Potential GPU Acceleration
System Architecture

Contributors

MI Diaries App

Back-end

FFmpeg

MI Diaries DB

Google ASR

On-Prem

Feathers

Front-end

Vue.js

Admin Users

Replace

The Capstone Experience

Team Michigan State Linguistics Project Plan Presentation
System Architecture

[Diagram showing system architecture with labeled components such as Training DB, Docker Container, Client, Input Processing, Prediction, Pre-processing (Data Engineering), Model Fine Tuning, Trained Models, Hugging Face Pre-trained Models, Model DB, Transcribed DB, Pre-processing (Language Detection), Prediction (Speech to Text, Speaker Diarization), Post-processing (Private Information Detection, Acoustic Analysis)].
System Components

• Hardware Platforms
  ▪ iMacs
  ▪ MSU EGR GPU Computes

• Software Platforms / Technologies
  ▪ Machine Learning
    ○ HuggingFace
      ❖ PyTorch
      ❖ Wav2Vec2, WavLM
    ○ Natural Language Processing
      ❖ NLTK, spaCy, Gensim
  ▪ Docker
  ▪ GitHub
  ▪ Python and PyCharm
Risks

• **Inadequate Data for Speaker Diarization**
  ▪ Currently not enough labeled data for supervised learning in speaker diarization
  ▪ Mitigation: Self-supervised models or using publicly available labeled datasets

• **Training and Predictions Times without GPUs**
  ▪ Speed for training and prediction models is bottlenecked by the model architecture
  ▪ Mitigation: Possible to use a smaller model which may be less accurate if speed is critical.

• **Extending to different dialects and languages**
  ▪ Extension to different dialects is dependent on data we don't have
  ▪ Mitigation: We will create features for users to upload their own training datasets which can fine-tune ASR models provided by Hugging Face.
Questions?