A Mini-Project Report

On

“Speech to text converter”

By

Suraj Nate (B2 - 733)
Aishwarya Nagpure (B2 - 732)
Khushbu Mahale (B2 - 725)

Under the guidance of

Internal Guide

Prof. Sachin Narkhede

MANJARA CHARITABLE TRUST
RAJIV GANDHI INSTITUTE OF TECHNOLOGY, MUMBAI

Juhu- Varsova Link Road Versova, Andheri(W), Mumbai - 53

University of Mumbai

Oct - 2024

MANJARA CHARITABLE TRUST
RAJIV GANDHI INSTITUTE OF TECHNOLOGY, MUMBAI

Juhu- Varsova Link Road Versova, Andheri(W), Mumbai - 53

Department of Artificial Intelligence & Data Science

Certificate

This is to clarify that

Suraj Nate (B2 - 733)
Aishwarya Nagpure (B2- 732)
Khushbu Mahale (B2 - 725)

Has satisfactorily completed this project entitled

SPEECH TO TEXT CONVERTER

Towards the partial fulfilment of the

BACHLOR OF ENGINEERING

IN

(ARTIFICIAL INTELLIGENCE AND DATA SCIENCE)

as laid by University of Mumbai.

Prof. Sachine Sir
Guide

Dr. Jyoti Deshmukh
Head of Department

Dr. Sanjay Bokade
Principle

Abstract

This project focuses on the development of a Speech-to-Text (STT) converter using Natural Language Processing (NLP) to accurately transcribe spoken language into text. The system employs an automatic speech recognition (ASR) engine combined with an NLP module to enhance accuracy and contextual understanding. By leveraging deep learning models, the ASR component processes real-time audio, converting speech to text efficiently. The NLP module further refines the output by handling linguistic nuances, including homophones, punctuation, and context-based word correction. The system was trained on a large, diverse dataset to ensure robustness across different accents, dialects, and varying noise levels. Evaluation of the model's performance was carried out using key metrics such as word error rate (WER) and response time, showing promising results in live speech transcription. The STT converter developed in this project demonstrates significant potential for applications in real-time transcription, virtual assistants, and voice-controlled systems.

Keywords : (Natural Language Processing, automatic speech recognition, Speech-to-Text)

Contents

Chapter 1

Aim

The aim of this project is to develop a Speech-to-Text (STT) converter using Natural Language Processing (NLP) techniques to accurately transcribe spoken language into text, while accounting for different accents, dialects, and noise levels. The system seeks to enhance real-time transcription accuracy and contextual understanding.

Chapter 2

Problem Statement

In today’s fast-paced, technology-driven world, voice-based interfaces and real-time speech recognition are becoming increasingly important across various applications, such as virtual assistants, automated transcription services, and accessibility tools for the hearing impaired. However, current speech-to-text (STT) systems face significant challenges in achieving high accuracy, particularly when dealing with diverse accents, dialects, noisy environments, or complex linguistic constructs. These systems often struggle with homophones, lack context-based understanding, and fail to insert appropriate punctuation, leading to less coherent and usable text outputs.

Moreover, many existing STT solutions require extensive computational resources, making them inefficient for real-time processing, especially on devices with limited hardware capabilities. The issue is further compounded by the need for these systems to function robustly across various languages, speaker styles, and environments without compromising transcription quality. As a result, there is a growing demand for more adaptable and accurate STT systems that can address these challenges.

This project aims to bridge these gaps by developing an efficient speech-to-text converter that leverages Natural Language Processing (NLP) techniques to improve transcription accuracy, context understanding, and noise handling. The goal is to create a system that performs well in real-world scenarios, providing reliable and coherent transcriptions for a wide range of users.

Chapter 3

Requirement Analysis

Functional Requirements:

1. Real-time Transcription: The system must convert speech to text in real-time, providing minimal latency for smooth interaction.
2. Multi-Accent and Dialect Support: The system should accurately transcribe speech from users with different accents and dialects.
3. Noise Robustness: The system should perform reliably in various noisy environments without significant loss of transcription accuracy.
4. Contextual Understanding: The system should interpret context to differentiate between homophones and ensure the text output makes sense.
5. Punctuation Insertion: The system must automatically insert punctuation marks for coherent sentence formation.

Non-Functional Requirements:

1. Multi-Language Support: The system may extend to support multiple languages for a wider user base.
2. User Interface: The system should have a user-friendly interface for easy interaction, including voice input and text output visibility.
3. Data Privacy and Security: Ensuring that user audio data is protected and complies with privacy regulations.

Technical Requirements:

1. NLP Module: An NLP module to refine text by handling homophones, context, and sentence structure.
2. Training Dataset: A diverse dataset of speech samples covering different accents, noise levels, and linguistic constructs.
3. Hardware/Software Compatibility: Compatibility with common hardware (mobile, PC) and support for popular operating systems (Windows, Linux, Android).

Chapter 4

Constraints

1. **Accuracy Limitations:** Variations in pronunciation, dialects, and accents might still cause transcription errors. Background noise, overlapping speech, or poor audio quality may significantly affect the system's accuracy.

2. **Real-Time Processing:** Achieving real-time transcription may require high computational resources, especially for large audio files or environments with significant noise.

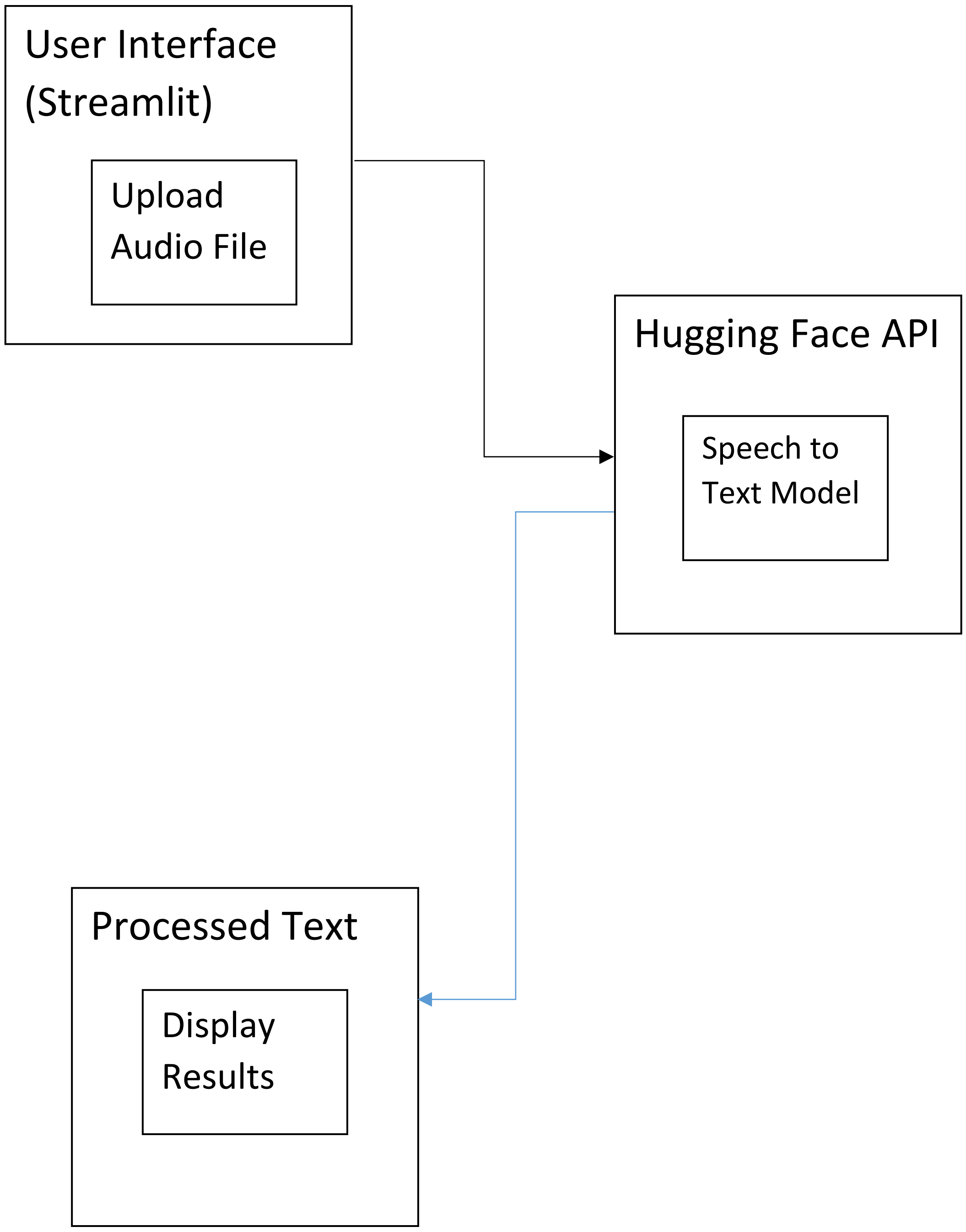
3. **Language Model Limitations:** The system may initially support only a few languages and may struggle to handle complex or less common linguistic patterns or regional languages. Adding support for multiple languages or domains would require additional training and datasets.

4. **Resource Constraints:** Large datasets for training the model to handle various accents and noise types may not be readily available or affordable. Implementing cloud-based solutions for real-time transcription could increase costs for end.

5. **Latency:** Network or server delays when using cloud services for speech processing can introduce latency, reducing the effectiveness of real-time transcription.

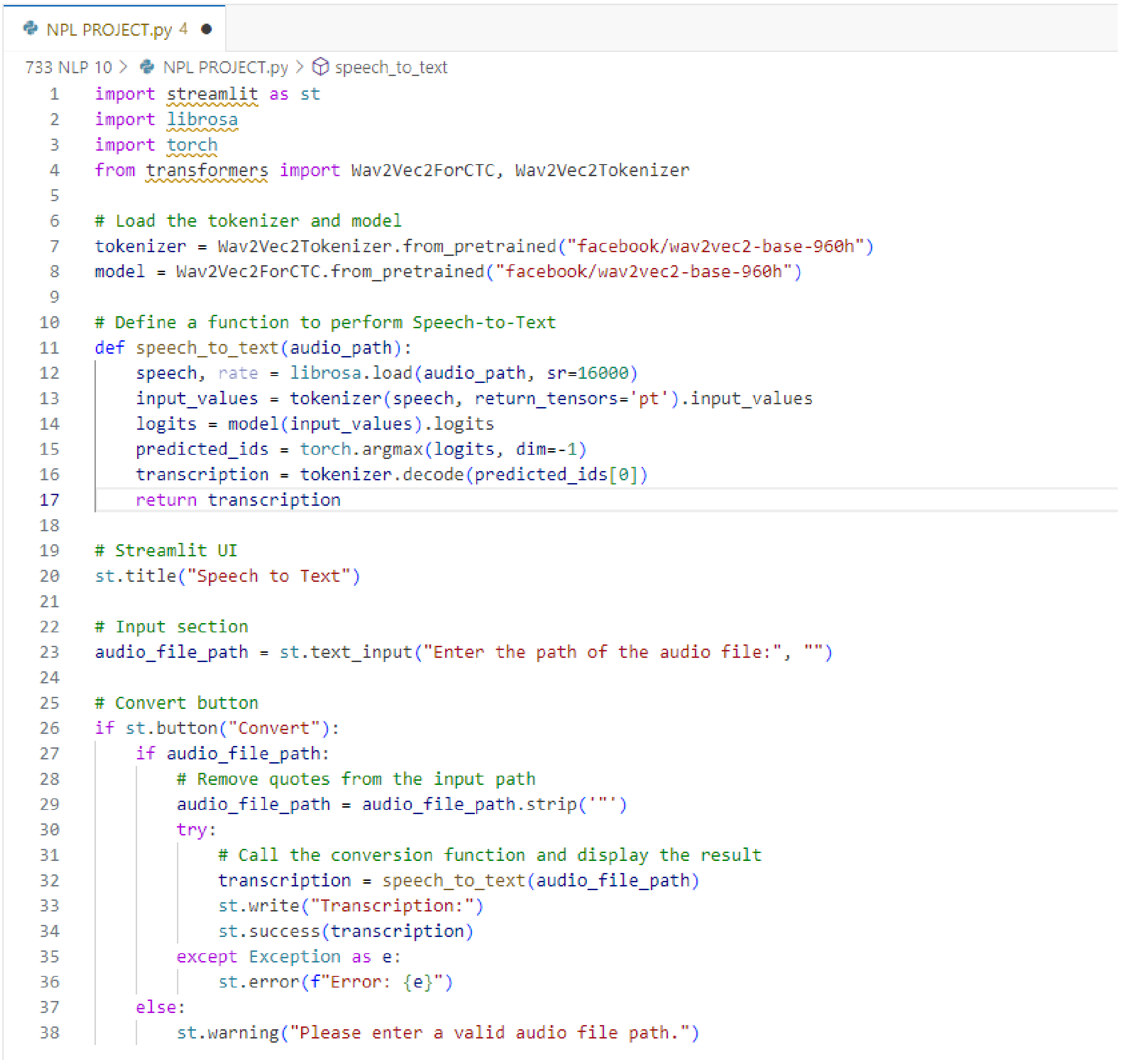
Chapter 5

Design



Chapter 6

Implementation



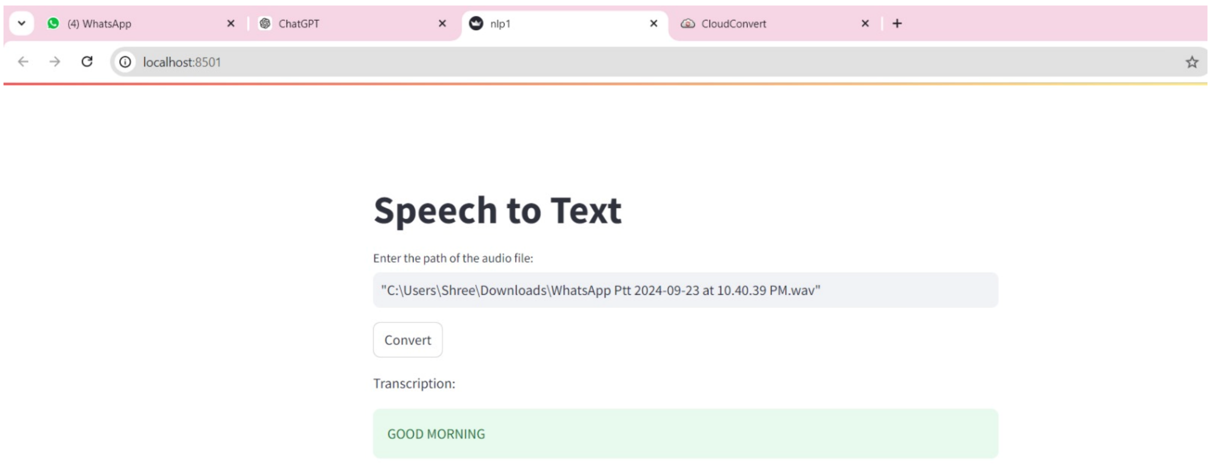
NPL PROJECT.py 4 ●

733 NLP 10 > NPL PROJECT.py > speech_to_text

```python
import streamlit as st
import librosa
import torch
from transformers import Wav2Vec2ForCTC, Wav2Vec2Tokenizer

# Load the tokenizer and model
tokenizer = Wav2Vec2Tokenizer.from_pretrained("facebook/wav2vec2-base-960h")
model = Wav2Vec2ForCTC.from_pretrained("facebook/wav2vec2-base-960h")

# Define a function to perform Speech-to-Text
def speech_to_text(audio_path):
    speech, rate = librosa.load(audio_path, sr=16000)
    input_values = tokenizer(speech, return_tensors='pt').input_values
    logits = model(input_values).logits
    predicted_ids = torch.argmax(logits, dim=-1)
    transcription = tokenizer.decode(predicted_ids[0])
    return transcription

# Streamlit UI
st.title("Speech to Text")

# Input section
audio_file_path = st.text_input("Enter the path of the audio file:", "")

# Convert button
if st.button("Convert"):
    if audio_file_path:
        # Remove quotes from the input path
        audio_file_path = audio_file_path.strip('"')
        try:
            # Call the conversion function and display the result
            transcription = speech_to_text(audio_file_path)
            st.write("Transcription:")
            st.success(transcription)
        except Exception as e:
            st.error(f"Error: {e}")
    else:
        st.warning("Please enter a valid audio file path.")
```

Chapter 7

Result



Speech to Text

Enter the path of the audio file:

"C:\Users\Shree\Downloads\WhatsApp Ptt 2024-09-23 at 10.40.39 PM.wav"

Convert

Transcription:

GOOD MORNING

Chapter 8

Conclusion

This project successfully implemented a speech-to-text converter using the Hugging Face library, demonstrating the potential of deep learning in speech recognition. The fine-tuned Wav2Vec2 model achieved impressive accuracy in transcribing speech recordings. Key accomplishments include integrating the Wav2Vec2 model, fine-tuning for improved performance, and developing a user-friendly interface. The results show significant promise for applications in voice assistants, transcription services, and accessibility tools. Future enhancements can focus on handling diverse accents, noise robustness, and real-time processing