# Major Examination Report
# Soham Deshmukh
# B21EE067

## 1. Question 1

In this question, the task defined is to analyze a lecture from the Speech Understanding course, which contains a mix of English and Hindi (code-switching). For this, I used the class recording from the class on 'Presentation Attacks'. I attempted to generate the audio of the video in Marathi (a relatively low-resource language).

1. **Preprocessing:**
   - We use the OpenAI Whisper model that works on all forms of audio media and multimedia - .mp3, .mp4, .wav, .mpeg, etc.
   - So, there are no particular preprocessing steps involved in this part.
2. **Transcription:**
   - Loads video files using `OpenAI Whisper`.
   - Transcribes it using a two step process - first, separation of audio from visual component of video, and second, transcription of the audio into corresponding text.
   - It returns a JSON formatted result containing the transcribed 'text'.
   - Then I pre-processed the transcript to remove filler words such as "um", "uh", etc., using *regular expression (re)*.
   - The overall output of this process is a string of transcribed text in English (mostly) obtained from the source video.
3. **Translation:**
   - It includes multiple steps -
     (a) Splitting the text into chunks because of constraints on compute and context window of models
     (b) Translation of English transcript chunks of the video into Marathi using ***Deep Translator*** by ***Google***.
     (c) Recombination of the translated chunks (now in Marathi) to create a single Marathi script
4. **Audio Generation:**
   - Generates the audio of the transcribed text in the low-resource language.
   - It includes multiple steps -
   - Text to Audio conversion using **'Indic Parler TTS'** model by **AI4Bharat**.
   - The base prompt to the model (description) can be customized to get the sound of different languages, types, genders, age, accents, etc.
   - We choose the Marathi output sound and feed the translation chunks into the model. The output audio pieces are then extended and combined together to generate the complete audio file for the entire transcript.

Transcription results containing the following are located in this Google Drive link for reference.
**Folder Link**
It contains the following files in the folder 'Question 1'.
- speech_recording.mp4 - the original input video file
- cleaned_transcript.txt - the transcript obtained using Whisper
- marathi_tts_out.wav - translated text converted to audio in Marathi

### 1.1. Challenges faced

While working on this question, one of the major issues I faced was working in a resource constrained environment and running models with huge memory requirements. The Whisper model is a relatively smaller model and hence for many languages, it's performance is not at par with many other multilingual models that have more parameters and larger sizes. At the same time, I had to resort to chunking/windowing the input to the audio generator because of GPU constraints. I also noticed that the model tends to perform poorly when hard slices of the audio/video are made during chunking. This is evident clearly when the processed. translated chunks in Marathi are recombined into a single audio file and there is a noticeable number of regions where the audios from different chunks don't sync completely.

# 2. Question 2

In this question, we are asked to analyze audio recordings from various events to assess noise levels in each of the scenarios. For this, we are given two types of data - one with pairs of clean audios and their noise-induced versions. The other set is only noisy audio files.

## 2.1. Noise Level Analysis

- In this part, I first extracted the dataset from ZIP files into the current working directory (the instructions for the same are included in the code file (.ipynb) itself).
- The next task was to consider pairs of audio files - the clean audio and its noisy version - and analyse them.
- The function *analyze_paired_data()* loaded these pairs and computed the *Signal-to-Noise Ratio (SNR)* between them. This was done by subtracting the clean audio from the noisy audio to get the noise signal.
- The frequency spectrum of the noise, the clean audio and the noisy audio were all plotted using ***librosa*** and the features such as *low, mid, high frequency energies, spectral centroid, bandwidth, etc.* were calculated.
- Similarly, the waveforms and spectrograms for clean, noisy signals and the noise itself was also plotted.

## 2.2. Denoising Algorithm Design

- **Spectral Subtraction** - estimates the noise from the audio signal and then subtract it.
- **Wiener filtering** - used to minimize the MSE between clean and noisy signals. It assumes both the signal and noise are stationary random processes with known spectral characteristics.

$$\text{SNR}(\omega) = \frac{P_s(\omega)}{P_n(\omega)}$$

- **Adaptive Wiener filtering** - enhances the quality by preserving speech characteristics. By using speech presence probability, it applies less aggressive filtering to speech-dominant regions.

$$\hat{X}(\omega) = \frac{P_s(\omega)}{P_s(\omega) + P_n(\omega)} \cdot Y(\omega)$$

- **Combined Approach** - combination of spectral subtraction and adaptive Wiener filtering. Spectral subtraction operates directly on the spectrum of the noisy signal to reduce noise. Adaptive Wiener Filtering takes the output from spectral subtraction as its input and applies statistical filtering based on estimated SNR at each time-frequency point.

I used SNR between the clean reference and the processed signals as the metric. I also used spectral distance between the frequency domain representations of audios to quantify the audio quality. These are represented as spectrograms and waveforms of signal as well as noise.

## 2.3. Transcription

- For transcription, I used *OpenAI Whisper* model.
- The transcripts had to be processed to remove the filler words from the text.

## 2.4. Performance Evaluation

- The performance of the denoising algorithm is calculated using the Signal-to-noise Ratio (SNR), Perceptual Evaluation of Speech Quality (PESQ),Short-Time Objective Intelligibility (STOI) and Spectral Distance.
- The metrics are computed between the clean samples and the denoised samples to check entent of noise removal.

| Metric | Values |
|---|---|
| SNR | 14.8749 |
| PESQ | 2.19 |
| STOI | 0.914 |
| Spectral Distance | 9.823 |

Table 1. Results. Ours is better.

- Results for the same can be observed in the table below.

### 2.5. Result Analysis

- We can observe that the average **SNR** value is **14.8749** which means that the audio signal strength is much higher than the noise.
- This tells us that the filtering process has subdued the noise in the samples to a great extent.
- The **PESQ** value of **2.19** tells us that most samples are of fair quality (since 4 is maximum) but contain traces of noise.
- **STOI** value of **0.914** (between 0-1) indicates that the speech is still highly informative and clearly understandable, inspite of traces of noise being present.
- Average **Spectral Distance** is around **9.823** which shows slight mismatch between the spectrograms of the pura and poisy signals.

You can find the link to all the denoised audio and transcriptions here - **Drive Link**.

Some of the images of noisy audio waveforms, denoised waveform and their respective spectrograms are given in Figure 3 below.

## 3. Question 3

**Problem Statement** - Detecting Early-Onset Neurodevelopmental Disorders (e.g., Autism Spectrum Disorder - ASD) in Children through Speech Understanding

Although speech understanding has been effectively utilized to identify cognitive impairment in elderly patients (such as Alzheimer's disease), early-onset neurodevelopmental conditions in children—such as ASD—continue to be very under-researched within speech-based detection work.

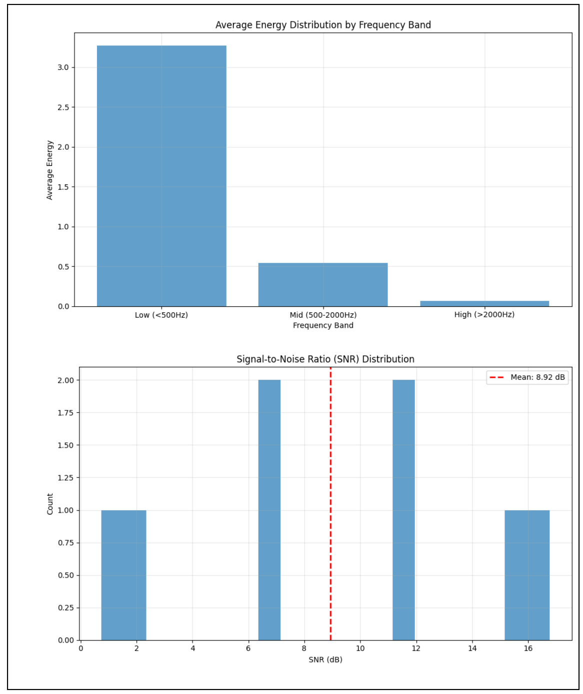


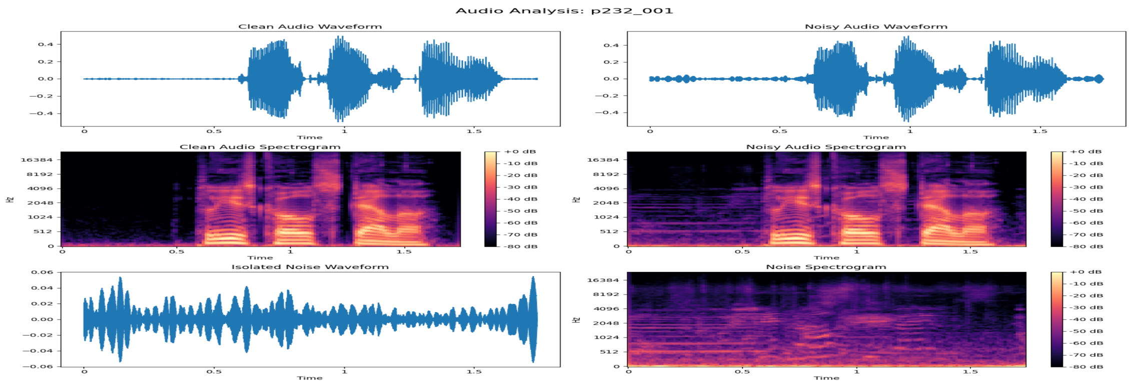

Figure 1. Top: Energy Distribution by Frequency Bands. Bottom: Signal-to-Noise Ratio (SNR) Distribution in Ques2.

Such conditions tend to exhibit mild prosodic, phonetic, and pragmatic speech irregularities many years prior to official diagnosis (typically around age 3-5).

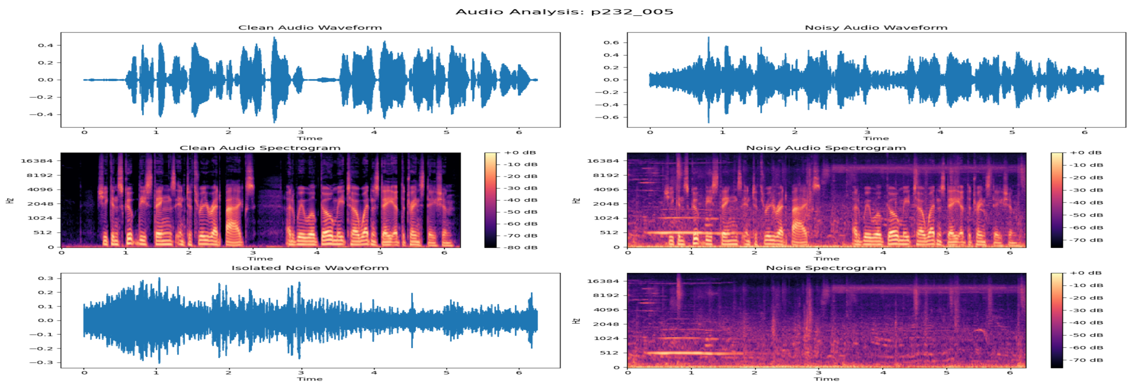
### 3.1. Importance of this solution

-

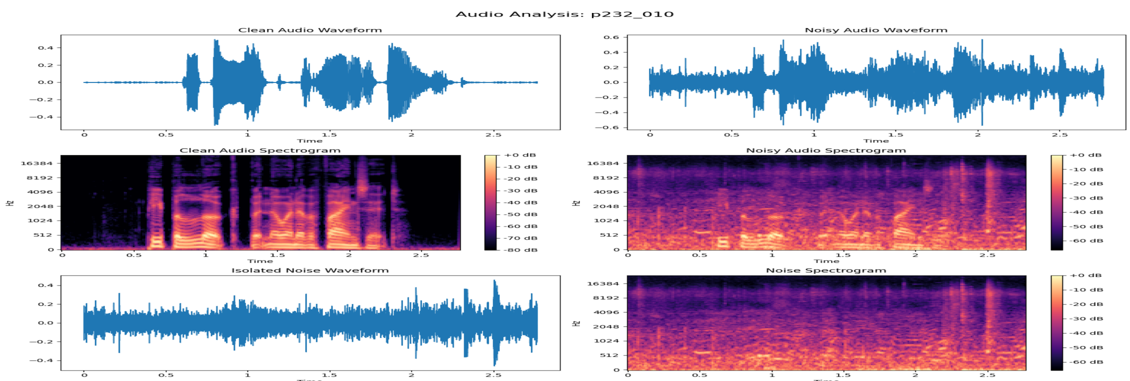
1. Critical Gap: Existing diagnosis technologies for conditions such as ASD are subjective, manual, and often delayed.
2. Scientific Impact: Early intervention significantly enhances cognitive and social development in children, yet we do not have scalable, speech-based, non-invasive early diagnosis technologies.
3. Commercial/Societal Impact: Low-cost, widespread screening technologies can be integrated into pediatric telehealth infrastructures, opening up early diagnostics to the broader




(a) p232_001




(b) p232_005




(c) p232_010

Figure 2. Spectral and waveform properties of some audio samples (for example purposes) from Question 2

population.

### 3.2. Proposed methodology

A hybrid model combining speech signal processing, transformer-based sequence modeling, and behavioral context embeddings - Multimodal Speech-Behavior Transformer (MSBT) - can be a good solution for this.

Its components can include:

- Input Features (Multiscale):
  - Low-Level Acoustic Features: MFCCs (prosody and tone), Log-mel spectrograms (frequency variation), Zero-crossing rate (speech fluency)
  - High level Embeddings: Pretrained models like Wav2Vec2.0, Whisper, or HuBERT for contextual speech understanding.
- Behavioral Embedding Module:
  - Textual cues from speech (semantic errors, repetitive phrases)
  - Pauses, echoing, and unusual intonations captured through speech diarization
- Core Model:
  - A dual-encoder architecture: one Transformer Encoder for speech features, one for behavioral context embeddings
  - Attention (wieghted) Layer to combine both streams
- Objectives/Tasks:
  - Classification (e.g., ASD vs control)
  - Anomaly detection score (unsupervised pretraining)
  - Speech rhythm prediction (self-supervised)

### 3.3. Usecases

- Integration into telehealth apps or speech therapy bots
- Integration into digital pediatric care
- Speech-driven IQ, EQ screenings

## 4. Links

Question 1 Notebook - Question 1.ipynb
Question 2 Notebook - Question 2.ipynb

## 5. References


1. A. Radford, J. W. Kim, T. Xu, G. Brockman, C. McLeavey, and I. Sutskever, ”Whisper: Robust Speech Recognition via Large-Scale Weak Supervision,” in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2022, pp. 1-10.
2. AI4Bharat, Indic-Parler-TTS, https://huggingface.co/ai4bharat/indic-parler-tts
3. Google Deep Translator https://pypi.org/project/deep-translator/
4. Librosa Official Documentation https://librosa.org/doc/latest/index.html
5. Wiener filter - Scipy documentation https://docs.scipy.org/doc/scipy/reference/generated /scipy.signal.wiener.html
6. Adaptive Wiener Filters https://github.com/rishiraj824/adaptive_wiener_filters