# Approaches Tried and Experiments Conducted for Novelty Assessment

Team Name: **NLPeeps**
**Aryash Srivastava (22B1506), Dhruv Garg (22B1529)**
*Guided by: Prof. Balamurugan*
*TA's: Mr. Rahul, Mr. Bheeshm*
Course: IE 643-2024-1 — Deep Learning - Theory and Practice
Department of Industrial Engineering and Operations Research, IIT Bombay

## Introduction

This document outlines the novelty assessment of our project, *Feeble Audio-Based Transcript Generation*, which focuses on transcribing noisy and feeble audio signals while addressing transcription challenges. The project utilizes pre-trained models, audio distortion handling, and predictive modeling techniques to handle muted sections. Below, we detail the approaches tried, experiments conducted, and solutions implemented.

## Problem Statement

The primary issues identified were:

1. Only the first pause was handled in audio segmentation.
2. Predictions included incorrect languages.
3. No built-in denoising mechanism, leading to errors in noisy audio.

## Approaches and Solutions

### 1. Multiple Pauses Handling

**Problem:** Only the first pause was being detected, limiting the transcription's completeness. **Solution:**

- Spectrogram analysis was used to identify multiple pauses by analyzing low-amplitude regions.
- Each pause was processed independently using OpenAI Whisper for transcription.
- Predictions for missing words in each pause were generated separately to ensure accuracy.

**Methodology:**

$$S(f,t) = |\text{STFT}(x(t))|^2 \tag{1}$$

where $S(f,t)$ is the spectrogram, and STFT represents the Short-Time Fourier Transform.

### 2. Language Mismatch in Predictions

**Problem:** The model sometimes predicted non-English words due to pre-trained weights. **Solution:**

- A customized dataset based on Mozilla's *Common Voice* was used for fine-tuning.
- The training focused on English-only recordings, ensuring consistent language predictions.

**Dataset Details:**

- **Dataset:** Mozilla's Common Voice (200,000 samples, 3-7 seconds each).
- **Preprocessing:** Clipping and noise augmentation using *librosa* and *numpy*.

### 3. Noise Resilience

**Problem:** Excessive noise caused transcription errors due to missing denoising capabilities. **Solution:**

- Gaussian noise ($\eta(t) \sim N(0, \sigma^2)$) and white noise were added to simulate distortions.
- A noise addition function was implemented to create distorted audio for training and testing:

$$y(t) = x(t) + \eta(t) \tag{2}$$

  where $x(t)$ is the original audio, $y(t)$ is the distorted audio, and $\eta(t)$ is Gaussian noise.

- Spectrogram decomposition was used to separate noise from useful signals.

## Experimental Results

**Dataset Testing:**

- **Input:** Then I got a hold of some dough and went goofy.
- **Predicted:** Then got a hold of some joe and went goofy.

**Real-Time Testing:**

- **Input:** Sun from East.
- **Predicted:** Sun rises from East.

**Metrics:**

| Metric | Baseline Model | Proposed Model |
|---|---|---|
| Accuracy (%) | 65% | 85% |
| Noise Resilience (%) | 50% | 78% |

## Conclusion

The proposed solutions successfully addressed the identified inefficiencies:

1. Spectrogram analysis and pause-wise processing improved transcription accuracy across multiple pauses.
2. Fine-tuning on an English-only dataset eliminated language mismatches.
3. Spectrogram decomposition enhanced noise resilience, improving predictions under distorted conditions.

The integration of these solutions represents a significant advancement in handling feeble and noisy audio transcription challenges.