Transcription, Translation and Personalized TTS of Code-Switched Lectures for Speech Understanding.

Lt Col Ritesh Lamba
MTech (AI) Executive, Indian Institute of Technology, Jodhpur
Roll No: M23CSA544
M23CSA544@iitj.ac.in

Abstract

This report presents an end-to-end pipeline for transcribing code-switched lectures, translating them into Hindi and synthesizing speech using a cloned voice. The goal is to evaluate transcription and TTS quality for code-switched speech, leveraging Whisper, Google Translate and YourTTS. Results show high transcription accuracy (WER = 4%) and strong subjective audio quality (MOS = 4.2).

1. Introduction

Code-switching in academic speech introduces challenges in automatic transcription, translation and speech synthesis. This work aims to create a pipeline that takes a lecture in English with Hindi elements, generates a clean transcript, translates it into Hindi and synthesizes it in the user's voice.

2. Pipeline Overview

Step 1. **Transcription:** Whisper (medium model) was used to transcribe a lecture video Titled: ”Speech Enhancement” Dated: 25 Jan 2025. Filler words were removed using regex.

Step 2. **Translation:** Google Translate API (via deep-translator) was used to convert English text to Hindi with chunked input.

Step 3. **TTS with Voice Cloning:** Tortoise TTS was used to synthesize speech from the translated Hindi text, using a short English voice sample recorded by the user.

Step 4. **Evaluation:** WER was calculated against manually transcribed reference; MOS was gathered from 5 human evaluators.

3. Tools and Models Used

- **ASR:** OpenAI Whisper (medium)
- **Translation:** deep-translator (GoogleTranslate backend)
- **TTS:** Tortoise TTS (Text-to-Speech with zero-shot voice cloning)
- **Voice Sample:** 40 Sec English recording of a pre decided script (16kHz mono WAV)

4. Evaluation Results

Metric	Value	Notes
Word Error Rate (WER)	**4.00%**	Whisper vs manual segment
Mean Opinion Score (MOS)	**4.2 / 5**	Ratings: [5, 4, 5, 3, 4]

Table 1. Transcription and TTS evaluation metrics

5. Discussion

WER of 4% indicates excellent transcription quality despite code-switching. Whisper effectively handled lecture speech and filler removal. **MOS** of 4.2 shows YourTTS produced intelligible and pleasant audio even from a short English-only voice sample. Chunked Hindi translation using deep-translator helped avoid size limits and ensured continuity.

6. Explanation of Metrics

- **Word Error Rate (WER)** measures transcription accuracy by comparing the predicted transcript to a ground truth. It captures substitution, insertion and deletion errors.
- **Mean Opinion Score (MOS)** is a subjective evaluation metric where human listeners rate audio quality from 1 (bad) to 5 (excellent). It reflects perceived naturalness and clarity.

7. Challenges Faced

- **Googletrans Failures:** Frequent NoneType errors and other errors prompted switching to deep-translator after failure to work with ai4bharat/indictrans2-en-indic-1B and googletrans.

- **Translation Size Limits:** The initial input size was too big and Google Translate supports up to 5,000 characters. Input was chunked using textwrap to overcome this.
- **Python Incompatibility:** Coqui TTS and YourTTS failed under Python 3.11+. Environment was rebuilt using Python 3.10.
- **Unpickling Errors:** PyTorch 2.6 introduced stricter defaults. These were handled using `add_safe_globals()`.
- **Voice Cloning from WAV:** Bark was unreliable. YourTTS was used locally to synthesize Hindi using an English voice sample.
- **Voice cloning stability:** YourTTS was initially considered but had compatibility issues. Tortoise TTS was used instead for more robust inference in Colab and better Hindi support.

8. Conclusion

This pipeline demonstrates accurate transcription, effective Hindi translation and personalized voice synthesis for code-switched academic lectures. It highlights how open-source tools can enable accessible TTS workflows in multilingual and low-resource contexts.

References

1. Radford, A., et al. ”Robust Speech Recognition via Large-Scale Weak Supervision.” *Whisper by OpenAI.* GitHub: https://github.com/openai/whisper
2. Google Translate API used via deep-translator: https://github.com/nidhaloff/deep-translator
3. Edresson Casanova et al. ”YourTTS: Towards Zero-Shot Multi-Speaker TTS and Zero-Shot Voice Conversion for everyone.” *Coqui TTS + YourTTS model.* https://github.com/coqui-ai/TTS
4. JiWER: Evaluation library for WER/CER. https://github.com/jitsi/jiwer
5. AI4Bharat, IndicTrans2 (Attempted): https://github.com/AI4Bharat/IndicTrans2
6. PyTorch Serialization Compatibility Patch: https://pytorch.org/docs/stable/generated/torch.load.html
7. Googletrans (Deprecated): https://github.com/ssut/py-googletrans

Deliverables & Supplementary Resources

1. **Colab File** with code. Google Colab: https://colab.research.google.com/drive/1zHGGi3J3nheok2ylV64x9ZLxnqaBk2R1
2. **Lecture Video** (25 Jan 2025 – Speech Enhancement). Google Drive: https://drive.google.com/file/d/1C5NxR7o-fY4JViuFD72Y99lVKA9lbBeR
3. Input **Transcript Text** File (filler-removed). Google Drive: https://drive.google.com/file/d/1qsJqkAvb9hiWowclHFtBKMV-3FhiS99G
4. Translated **Hindi Text** File. Google Drive: https://drive.google.com/file/d/1x-AYvELvfmI_3h5bajLbECTKuPgBOBIB
5. **Voice Sample** File. Google Drive: https://drive.google.com/file/d/1ahUfB4E4j3eJvZ5bR3G02ilm36lJwD5r
6. **Generated Audio** Output. Google Drive: https://drive.google.com/file/d/1P0yz-XMs3AK3cGq2AAdk_0M9VEc0kc8a
7. **Program for Voice Modelling** Google Colab: https://colab.research.google.com/drive/1NKGjfqKZkWQSNs8wsCTpMn9qUuAv3Ndq
8. Manually Corrected Reference for WER: Included inline in code for comparison