Question 1
Automated Video Lecture Transcription with Filler Word Removal

Harishankar Nagar
Github: link

1 Introduction

This report presents an automated approach for transcribing video lectures and implementing a post-processing pipeline to remove filler words. Filler words, such as ”um,” ”uh,” and ”like,” often contribute to transcription noise without adding substantive content. The implementation utilizes state-of-the-art speech recognition technology combined with regular expression-based text processing to generate clean, readable transcripts from educational video content.

2 Implementation

The implementation consists of two main components: audio extraction from video files and speech-to-text transcription with filler word removal. The system is implemented in Python, utilizing several specialized libraries.

2.1 Audio Extraction

The first step involves extracting the audio track from video files. This is accomplished using the FFmpeg multimedia framework, accessed through its Python bindings. The implementation creates a function that uses ffmpeg to convert video files to WAV format, which is optimal for speech recognition processing. The function takes a video file path as input and returns the path to the extracted audio file.

2.2 Transcription and Filler Word Removal

The core functionality is implemented in the `transcribe_audio` function, which utilizes OpenAI's Whisper model for speech recognition and regular expressions for text cleaning. The function performs the following operations:

1. Loads the Whisper ”small” model, which balances accuracy and computational efficiency
2. Transcribes the audio file to raw text
3. Removes common filler words (e.g., ”um,” ”uh,” ”you know”) using regular expressions
4. Normalizes whitespace to ensure the final text is clean and readable

The implementation defines word boundary markers in the regular expressions to ensure only complete filler words are removed from the transcription, preserving the integrity of the content while enhancing readability.

3 Technical Components

The implementation relies on several key libraries:

3.1 OpenAI Whisper

Whisper is an advanced automatic speech recognition (ASR) system developed by OpenAI. The implementation uses the ”small” model variant, offering a good balance between transcription accuracy and computational requirements.

3.2 FFmpeg

FFmpeg is used to extract audio tracks from video lectures, preparing them for the speech recognition process.

3.3 Regular Expressions

Python's `re` module is used to identify and remove filler words from the transcribed text. The approach uses word boundary markers to ensure that only complete words matching the filler patterns are removed.

4 Conclusion

The presented approach offers an efficient solution for transcribing video lectures with automatic removal of common filler words. The implementation leverages state-of-the-art speech recognition technology combined with text processing techniques to generate clean, readable transcripts.

References

1. OpenAI Whisper: https://github.com/openai/whisper
2. FFmpeg: https://ffmpeg.org/
3. FFmpeg-Python: https://github.com/kkroening/ffmpeg-python
4. PyDub: https://github.com/jiaaro/pydub