Insights of Code

This documentation outlines the functionalities of the provided Python code, including dependencies, functions for text and audio processing, and the main execution flow. It details how to set up the environment, process YouTube audio, manipulate text and audio data, and integrate with external APIs.

Dependencies Installation

1. Whisper Installation:

- Command: `!pip install git+https://github.com/openai/whisper.git`
- Purpose: Installs Whisper from OpenAI's GitHub repository to enable audio processing

2. FFmpeg Installation:

- Command: `!sudo apt update && sudo apt install ffmpeg`
- Purpose: Installs FFmpeg, a multimedia framework, required for handling audio and video files.

3. Python Packages Installation:

- `pytube`, `transformers`, `gtts`, `pyttsx3`, `pydub`, `torch`, `datasets`
- Purpose: Installs necessary Python packages for YouTube video download, text-to-speech, text processing with transformers, and audio manipulation

Google Drive Mounting

- Command: `from google.colab import drive; drive.mount('/content/drive/')`

- Purpose: Mounts Google Drive to access and read files stored on Google Drive.

Environment Variable Setup

- Purpose: Sets environment variables required for API keys or other configurations from a `.bashrc` file stored in Google Drive.

YouTube Audio Download Function

- Function: `download_youtube_audio(video_url)`

- Purpose: Downloads audio from a YouTube video specified by `video_url` using the `pytube` library.

Text Processing Functions

1. Split Content by Timestamp

- Function: `split_content_by_timestamp(content)`

- Purpose: Splits text content based on timestamp patterns to organize dialogue or subtitles

2. Write to File

- Function: `write_to_file(split_content, output_file)`

- Purpose: Writes processed content to a file.

3. Read from File

- Function: `read_from_file(filename)`

- Purpose: Reads text content from a file specified by `filename`.

Audio Processing Functions

1. Text-to-Speech (TTS) and Audio Generation

- Functions: `generate_audio_files(questions_answers)`, `generate_questions_answers(text)`

- Purpose: Utilizes `gTTS` for text-to-speech conversion to generate audio files from questions and answers.

2. Audio Combination

- Functions: `combine_audio(files, output_file)`

- Purpose: Combines multiple audio files into a single audio file using `pydub`.

Main Function Execution

- Purpose: Executes the entire workflow by reading a text file, generating questions and answers using GPT-3 API, saving them, generating corresponding audio files, and combining them into a final output.

Usage Example

- YouTube video used: `"https://youtu.be/oV74Najm6Nc?si=QMu8_VE_lgzk721L"`

- Text file used: `"/content/What Is Artificial Intelligence? | Artificial Intelligence (AI) In 10 Minutes | Edureka.txt"`

External Tools

- OpenAI API:

- Purpose: Used for generating questions and answers based on input text using the `get_completion` function.