Exercise 1 - Chatbot for Youtube video

1. Overview

This project implements a chatbot capable of answering natural language questions about video content. It supports input in the form of MP4 files or YouTube links, and optionally, image prompts to locate appearances of specific persons or objects.

2. Pipeline Summary

The chatbot system processes video input using the following steps:

- Download video from YouTube using yt-dlp
- Extract audio using ffmpeg
- Transcribe audio with whisper.cpp (with timestamps)
- Split video based on transcript timestamps
- Analyze visual content with InternVL3-8B (captioning, OCR, objects, scenes)
- Combine visual and text data into contextual chunks
- Use InternVL3-8B for LLM-based chat response with multimodal context

3. Features

- Input: YouTube links or MP4 files
- Natural language Q&A from video content
- Optional image input to find timestamp of a person/object
- Web-based chatbot interface

4. Source Code & Demo

Chatbot-video

Exercise 2 - LLM Deployment

1. Overview

This project demonstrates how to deploy the open-source BLOOMZ-1b1 language model using llama.cpp. The deployment is containerized with Docker and optimized for GPU inference with quantized model formats.

2. Features

- On-premise deployment
- REST API for chat completions
- Model quantized (Q8_0) to reduce VRAM usage
- Docker-based deployment for reproducibility

3. Model & Benchmark

- Model: bigscience/bloomz-1b1 (Q8_0 GGUF format)
- VRAM usage: ~1.4 GB
- Prompt speed: 25 tokens / 235.35 ms (≈106.22 tokens/s)
- Generation speed: 10 tokens / 324.63 ms (≈30.8 tokens/s)
- Hardware: RTX 3060 GPU

4. Source code

Llm-server

Report by

Name: Bui Thanh Liem

Email: liemkg1234@gmail.com

Github: liemkg1234