

## Video Content Classifier using OpenAI CLIP & Whisper

This Colab notebook offers a practical pipeline for analyzing video content safety by combining visual and audio classification. It uses [OpenAI's CLIP](#) for flagging unsafe visuals in frames and [OpenAI Whisper](#) for detecting unsafe language in audio. The tool automatically flags potentially unsafe videos based on customizable prompt lists for both text and vision.

### ► Features

- **Frame-by-frame safety analysis:**  
Extracts one video frame per second and classifies each using CLIP against user-defined textual prompts (e.g., "nudity", "violence", etc.).
- **Audio transcription and review:**  
Extracts and transcribes the video's audio track using Whisper, then scans for profane or unsafe words.
- **Automatic flagging:**  
Any frame or transcript matching unsafe criteria causes the video to be flagged, with evidence and confidence details shown.
- **Safe/unsafe frame previews:**  
Displays top unsafe frames and an example safe frame for manual review.

### 🚀 How It Works

1. **Install dependencies:**  
Installs CLIP, Whisper, and required Python libraries.
2. **Model loading:**  
Loads CLIP (ViT-B/32) and Whisper (base) onto GPU/CPU as available.
3. **Video upload:**  
Upload your video directly to the notebook.
4. **Frame extraction:**  
Extracts frames at 1 FPS for efficient classification.
5. **CLIP classification:**  
Each frame is compared to a set of safety/unsafety prompts; frames exceeding a configurable threshold are flagged.
6. **Whisper transcription:**  
The audio track is transcribed and checked against a list of unsafe/vulgar terms.

## 7. Reporting:

Summarizes findings, displays flagged frames and transcript details, and declares the video as SAFE or UNSAFE.

## Usage

1. Open the Colab notebook.
2. Run all cells.

The notebook will prompt for video upload and proceed through processing automatically.

3. Review outputs:
  - a. Unsafe frames and detected issues (with confidence)
  - b. Transcript and flagged words
  - c. Final safety verdict

## Sample prompts:

- *Safe*: “a safe scene”
- *Unsafe*: “nudity”, “pornographic content”, “graphic violence”, “explicit content”, “a person showing a middle finger”, etc.

You can customize these prompts for your own requirements.

## Dependencies

- clip (OpenAI implementation)
- whisper (OpenAI model)
- moviepy
- opencv-python
- Pillow
- regex
- tqdm
- ffmpeg-python

## Limitations

- Accuracy is threshold-dependent:  
The 0.5 threshold for unsafe content is a heuristic; tune for your use case.

- Prompt engineering matters:  
Expand or refine prompt lists based on desired sensitivity and category.
- Whisper and CLIP are not foolproof:  
Subtle or context-specific content may bypass detection.
- Does not work in real time:  
Designed for offline, batch processing.

## References

- [OpenAI CLIP GitHub](#)
- [OpenAI Whisper GitHub](#)

Instructions above are designed for direct use as a README .md or GitHub project description, clearly explaining the functionality, workflow, and requirements of your code.