

Efficient Video Summarization and Analysis with LLMs and RAG Architecture

Mrunal Malekar, Anusha Kalbande, Harshada Sasturkar

Abstract

In an era inundated with an abundance of video content spanning tutorials, product reviews, news updates, and fitness clips, the challenge of efficiently extracting relevant information persists. Often, time constraints or limited bandwidth hinder our ability to sift through lengthy videos for specific insights, leading to frustration and inefficiency. To address this issue, we present an innovative solution leveraging state-of-the-art Language and Vision models (LLMs) such as Llama-2-7b-Chat-GPTQ, FLAN T5, and BART. Our approach centers on the development of a robust video summarization system, designed to distill key concepts and highlights from diverse video sources. By harnessing the power of LLMs, we generate concise summaries of videos that encapsulate the essence of the original content. Additionally, we introduce a Multi-Modal Retriever-Generator (RAG) framework, enabling comprehensive analysis and retrieval of video content across modalities. Through rigorous experimentation and evaluation, our methodology demonstrates significant advancements in video understanding and summarization.

Introduction

In our modern era inundated with a deluge of video content spanning tutorials, product reviews, news segments, fitness clips, and more, the challenge lies not in scarcity but in abundance. With countless hours of footage at our fingertips, extracting pertinent information becomes akin to finding a needle in a haystack. The time constraints and limited bandwidth of our daily lives often render it impractical to sift through lengthy videos, only to realize halfway through that they do not serve our intended purpose.

To address this dilemma, we developed an efficient video summarization tool harnessing the cutting-edge capabilities of Language Model Architectures (LLMs). This ambitious task involves distilling meaningful insights from the inherently complex nature of video content, which amalgamates audio, visual, and textual modalities.

Our proposed integration seeks help in automating the extraction of pivotal scenes, dialogues, and concepts, thereby offering a transformative approach to textual summarization through video transcription. We use tools to load and process video content, converting audio to transcripts. We utilize advanced Large Language Model Architectures (LLMs) such as Llama-2-7b-Chat-GPTQ, BART, and Flan T5 for zero-shot learning to distill transcripts into concise abstracts, capturing the video's essence. Moreover, we propose the utilization of the Retriever-Generator (RAG) architecture using multi modal vector databases to effectively analyze and interpret video content, extracting salient information while preserving context. Responses are generated using vision models like GPT-4 Vision or Gemini Pro Vision. This innovative approach

not only streamlines the summarization process but also enhances comprehension and condensation of video data.

The potential applications of this technology are manifold, spanning domains such as security, education, and media. By providing users with concise summaries encapsulating the essence of lengthy videos in just a few lines, our tool promises to revolutionize information consumption, saving valuable time and resources.

Background

The tools and technologies used in this project constitute Python libraries for video processing, LLMs for video summarization and vector database for RAG implementation. To load a video from a given link we used pytube, yt-dlp and moviepy, commonly used for tasks such as downloading videos from online platforms, processing and manipulating video and audio files. The audio and visual components of the video were segregated with ffmpeg-python which provides a user-friendly interface for controlling ffmpeg, a powerful multimedia framework for handling multimedia files and streams. With the speech recognition library, we performed audio-to-text conversion. Speech recognition allows developers to incorporate speech-to-text capabilities into their Python applications, enabling features such as voice commands, transcription, and more. The LLMs - Llama-2-7b-Chat-GPTQ, FLAN T5, and BART were imported from the Hugging Face Model Hub platform that hosts thousands of pre-trained models for natural language processing tasks. We also made use of the CLIP model developed by OpenAI for generating embeddings as part of RAG architecture. For efficient semantic search with these embeddings, LanceDB vector database was used along with LlamaIndex which creates vector indices over large documents and corpora, enabling fast question-answering and information retrieval using language models like GPT. Additionally, a novel evaluation framework called G-Eval was incorporated that aims to provide a more comprehensive and task-agnostic way to measure the capabilities of LLMs with diverse benchmarks and metrics. See the Project Description section for more details about the implementation.

Related Work

Video summarization techniques have been a significant area of research within the field of computer vision and video analytics, aiming to distill lengthy videos into more manageable, concise formats. Here, we explore some of the primary techniques and methodologies that have been developed, alongside identifying potential gaps and opportunities for innovation.

Keyframe Extraction and Shot Boundary Detection: This technique simplifies videos into keyframes or short clips, highlighting essential moments through algorithms that detect transitions, like cuts and fades. However, it often neglects audio, a crucial part of content comprehension. **Action Recognition and Temporal Subsampling:** Focusing on identifying actions within videos, this method uses pattern recognition and deep learning to analyze video segments. While effective for specific applications like surveillance, it may overlook the video's broader narrative. **Single and Multi-modal Approaches:** Video summarization varies from focusing solely on one data type, like audio for textual summaries, to integrating multiple modalities. Single-modal methods might miss important visual details, suggesting the need for a comprehensive multi-modal approach.

Video summarization and multi-modal retrieval are vital areas of research in the field of multimedia analysis, with numerous studies exploring various techniques to efficiently extract and summarize key information from videos. Traditional methods often relied on manual annotations or rule-based approaches, which were limited in scalability and accuracy. However, recent advancements in deep learning and natural language processing (NLP) have led to significant progress in automatic video summarization.

One notable study by **Gygli et al. (2015)** introduced the concept of "SumMe," a large-scale benchmark dataset for video summarization evaluation. Their work highlighted the importance of objective evaluation metrics and provided a standardized benchmark for assessing summarization algorithms.

Moreover, approaches leveraging deep learning models have shown promising results in video summarization. For instance, **Zhang et al. (2016)** proposed a method based on Long Short-Term Memory (LSTM) networks to generate video summaries by modeling the temporal dynamics of video sequences. Similarly, **Mahasseni et al. (2017)** introduced a deep reinforcement learning framework for abstractive video summarization, demonstrating superior performance compared to traditional methods.

Multi-Modal RAG architecture combines text and visual information to enable effective content retrieval and generation. While traditional retrieval methods often relied on text-based queries or visual features separately, Multi-Modal RAG offers a holistic approach by integrating both modalities. One seminal work in this area is the RAG model proposed by **Lewis et al. (2020)**, which introduced a unified framework for text-based question answering and content generation. By pre-training on large-scale text and image datasets, RAG demonstrated remarkable capabilities in understanding and generating contextually relevant responses.

Furthermore, recent studies have explored the application of Multi-Modal RAG to video content retrieval. For instance, **Liu et al. (2021)** proposed a Multi-Modal Transformer for video question answering, where the model leveraged both textual and visual information to generate accurate responses to user queries. While our approach leverages Multi-Modal RAG for content retrieval, alternative methods exist in the literature. One such approach is the fusion of textual and visual features using traditional fusion techniques like concatenation or late fusion. However, these methods often suffer from information loss or

lack of contextual understanding compared to Multi-Modal RAG.

Another approach involves leveraging graph-based representations to model relationships between video segments as a graph and using graph algorithms to identify important nodes (segments) and edges (relationships) to generate summaries. Although effective in capturing temporal dependencies, these methods may struggle with scalability and require manual tuning of parameters. Our approach of utilizing zero-shot learning and prompt engineering with Language Model Architectures (LLMs) offers several advantages over graph-based representations for video summarization. Unlike graph-based methods, LLMs are highly scalable and can handle large volumes of data efficiently. They also automate prompt design, eliminating the need for manual parameter tuning. LLMs capture complex relationships in language, enabling them to generate more informative summaries.

Project Description

The project aim was to utilize LLMs and RAG architecture to create a user-friendly application which can take a video as an input and output either its summary or answer a video-related user query as needed. The complete technical flow can be seen in **Figure A1 (Appendix)**.

Our dataset for this project consisted of TVSum - 50 videos of around 10 MB size and SumMe - 25 videos of around 50 MB; each video having duration of about 5 min. Additionally, we included 10 YouTube videos - each of 100 MB size and 15 to 30 min long. The video topics had a broad range from new clips, movie reviews to educational tutorials. The project implementation can be broadly divided into three components.

- **Video Summarization Module:** The objective of this module is to extract audio and textual components from the video to get a brief abstract using LLM models. After taking a video URL as input, we load it with pytube, yt-dlp and moviepy. Audio and visual components (video frames) are extracted using ffmpeg using parallel processing. Speech recognition is used for audio to text conversion, essentially giving us the video transcripts. The transcripts are summarized using pretrained LLMs like Llama-2-7b-Chat-GPTQ, BART and Flan T5. Due to lack of ground truth summaries in the dataset we couldn't finetune the models. Instead, we employed zero shot learning along with prompt engineering (see **Figure A2, Appendix**) to improve accuracy.
- **Multi Modal RAG Module :** Multimodal retrieval augmented generation (RAG) is a technique that involves retrieving relevant information from a multimodal knowledge base and using it to augment the input for a generative model, which then generates the desired output (e.g., text, image, or audio) incorporating the retrieved information. We incorporated this into our project to answer video-specific user questions. The video's textual and visual data are stored as embeddings generated by OpenAI's CLIP model in LanceDB vector database and LlamaIndex framework. LanceDB uses similarity scores to enable efficient retrieval of most relevant text/image embeddings. The final responses are generated from the embeddings using GPT-4 Vision or Gemini Pro Vision.

- **Streamlit Deployment:** We integrated the above modules into a Streamlit application where users can request a video summary, transcripts or ask a particular question about the video content in general through the application.

The model summaries (inferences) obtained for the videos present in the dataset were then evaluated using G-Eval which helps to evaluate the quality of generated text using LLM. Our scoring LLM was GPT-4. There are 4 evaluation criteria used by G-Eval: Relevance - evaluates if the summary includes only important information and excludes redundancies, Coherence - assesses the logical flow and organization of the summary, Consistency - checks if the summary aligns with the facts in the source document and Fluency - rates the grammar and readability of the summary. We performed Chain-of-thought Prompting to guide the model to output a numeric score from 1-5 for each criterion. Finally, a Direct Scoring Function allows GPT - 4 to generate a discrete score (1-5) for each metric which is later normalized and averaged for the entire dataset.

Empirical Results

We did a comparative analysis of all 3 LLMs on our dataset using the G-Eval metric discussed above. The results are shown in **Figure 1**.



Figure 1: Model Evaluation Metrics

From the figure, we can see that LLAMA 2-7b-Chat-GPTQ excelled in all 4 metrics, achieving scores above 0.8. FLAN T5 BASE also performed well in coherence compared to BART. Both BART and FLAN T5 BASE exhibited significant issues with fluency. FLAN T5 BASE had a moderate performance in consistency, while BART was the lowest. BART's relevance was slightly higher than that of FLAN T5 BASE. The moderate performance of BART and FLAN T5 BASE could be due to - Inadequate model complexity, lack of finetuning and issues with the model's understanding of context. The complete Streamlit application output is shown in **Figure A3 to A8 (Appendix)**.

Broader Implications

The development of a robust video summarization tool leveraging advanced Language Model Architectures (LLMs) and multimodal vector databases holds immense potential for

democratizing knowledge and fostering inclusivity across various domains. By distilling complex video content into concise summaries, this technology can streamline educational processes, enabling efficient learning for students and focused lesson planning for educators. Moreover, it has the capacity to bridge gaps for learners with diverse backgrounds, learning styles, or language proficiencies, promoting equitable access to educational resources.

Beyond the academic sphere, the impact of this innovative tool extends to fields like journalism, security, and healthcare. Journalists could leverage video summarization for timely reporting of critical events, while security agencies could rapidly analyze surveillance footage to identify potential threats. In the healthcare sector, medical professionals could utilize summarized video content to stay up to date with the latest research, techniques, and best practices, ultimately driving advancements in patient care and medical knowledge. By harnessing the power of LLMs and multimodal vector databases, this technology promises to revolutionize the way we consume and comprehend information, fostering efficiency, accessibility, and inclusivity across diverse domains.

Conclusion and Future Directions

In conclusion, we were able to leverage LLMs and Multi Modal RAG architecture to efficiently perform video summarization and query answering. Employing prompt engineering along with zero shot learning resulted in some improvement in the model performance and accuracy. From the model evaluation, LLAMA 2-7b-Chat-GPTQ was determined to have the best performance, achieving scores above 0.8 in all four of the metrics. We created a user-friendly Streamlit application combining video processing, summarization, and RAG to generate relevant responses. The application can be used for quick comprehension of a diverse range of videos.

In future, we will be trying out larger video datasets preferably with ground truth summaries which could allow us to finetune our models, further enhancing model accuracy. We plan to work on enhanced personalization and customization to Develop adaptive learning algorithms that can tailor summaries based on user preferences, prior interactions, and specific interests. We will also be exploring real-time video summarization, which would be invaluable for live broadcasts and streaming content, allowing users to catch up or get summaries on-the-fly.

References

Gygli, M., Grabner, H., Van Gool, L. (2015). "Video Summarization by Learning Submodular Mixtures of Objectives." CVPR.

Lewis, P., et al. (2020). "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks." NeurIPS.

Liu, H., et al. (2021). "Multi-Modal Transformer for Video Question Answering." arXiv preprint arXiv:2108.01002.

Mahasseni, B., Lam, M., Todorovic, S. (2017). "Unsupervised Video Summarization with Adversarial LSTM Networks." CVPR.

Zhang, K., Chao, W. L., Sha, F., Grauman, K. (2016). "Summary Transfer: Exemplar-based Subset Selection for Video Summarization." CVPR.

GitHub Repository
https://github.com/mrunal-create/LLM_Project

Appendix

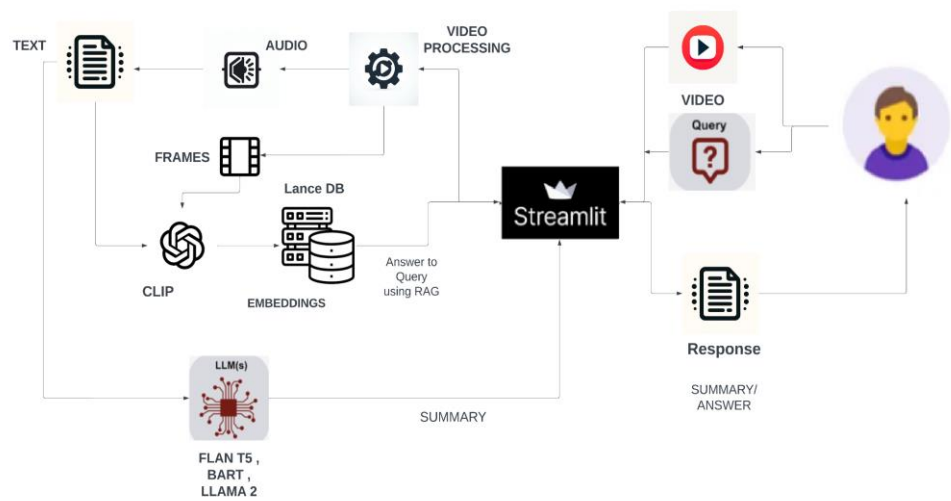


Figure A1: Technical Flow

```
template = """
    Write a summary of the following text delimited by triple backticks.
    Return your response which covers the key points of the text.
    ```{text}```
 SUMMARY:
 """
```

Figure A2: Prompt used for Zero Shot Learning

Video Summarizer and RAG

Enter the URL of the Video

Extract Audio for this video

Extract Text for this video

Summarize the video using:

☒ BART

☐ Flan-T5

☐ LLama-2

Summarize using the above selected model

Try asking a query based on the video

Find the answer

Figure A3: Home Page of the Application

# Video Summarizer and RAG

Enter the URL of the Video

[https://www.youtube.com/watch?v=d\\_qvLDhkg00&ab\\_channel=3Blue1Brown](https://www.youtube.com/watch?v=d_qvLDhkg00&ab_channel=3Blue1Brown)

Extract Audio for this video

Audio extraction completed!

Figure A4: Audio Extraction

# Video Summarizer and RAG

Enter the URL of the Video

[https://www.youtube.com/watch?v=d\\_qvLDhkg00&ab\\_channel=3Blue1Brown](https://www.youtube.com/watch?v=d_qvLDhkg00&ab_channel=3Blue1Brown)

Extract Audio for this video

Extract Text for this video

Text extraction completed!

Figure A5: Text Extraction

Try asking a query based on the video

what is this video about

Find the answer


 Finding the answer...

Figure A6: Query Answering

# Video Summarizer and RAG

Enter the URL of the Video

[https://www.youtube.com/watch?v=d\\_qvLDhkg00&ab\\_channel=3Blue1Brown](https://www.youtube.com/watch?v=d_qvLDhkg00&ab_channel=3Blue1Brown)

Extract Audio for this video

Extract Text for this video

Summarize the video using:

- ☐ BART  
☒ Flan-T5  
☐ LLama-2

Summarize using the above selected model

Here is the summary of the video using flan\_t5:

normal distribution AKA a gaussian is  $e$  to the negative  $x$  squared but you might wonder why this function of all the Expressions we could dream up that give you some symmetric smooth graph with mass concentrated towards the middle  $e$  to the negative  $x$  squared function is a convolution of two gaussian functions that has to be computed between the two original functions to compute the distribution describing the sum of those two functions. I'd like  $e$  to the negative  $x$  squared the exponent is typically written as  $-1/2 * x / \text{Sigma squared}$  where Sigma describes the spread of the distribution specifically the standard deviation all of this needs to be multiplied by a convolution is an integral expression that we build up last video and then to plug again for each one of the functions involved the formula for a gaussian is kind of a lot of symbols when you throw it all together  $x + y = s$  and  $y + x = y$  and  $s = x - y$  if  $x > y$  or  $y > x$  and  $x + y = s$  divide by the square root of  $y$  you still this area is the key feature to focus on you can think of it as a way to combine together all the probability densities for all of the outcomes corresponding to a given sum in the specific  $y = s = y$  for some value of  $s$  and a little bit of Pythagoras will show you that the straight-line distance from the origin to this line is sooooo divided by the square rallel to the  $y$ -axis is much much easier than slices in other directions because it only involves taking an integral with respect to why the value of  $x$  on the slice is a constant specifically it would be the constant Gaussian convolution for a normal distribution with a mean zero and an arbitrary standard deviation square root of  $2 * \text{sigma}$  if you haven't computed a lot of convolutions before it's worth normal distributions are often a little exaggerated but to the extent that they do come up it is because of the central limit theorem but it would be cheating to say that the central limits theorem implies this t of the central limit theorem is a gaussian in the first place and not some other function we've talked all about the central limit t "of the central limite theor" is p is actually pretty technical it goes a little beyond what I want to talk about here you often use these objects called moment generating functions that gives you a very abstract argument that there mustbe some Universal shape but it doesn't make ic argument that leverages the rotational symmetry of this graph is it directly connected to a few things that we've talked about on this channel before for example the Herschel Maxwell derivation of a gaus I'm not a big fan of a lot of things but if you are interested in following my work then this is probably one of the most enduring ways to do so - I try not to be too promotional at the end

Try asking a query based on the video

Find the answer

Figure A7: A Short Summary of the Video by Flan-T5



# Video Summarizer and RAG

Enter the URL of the Video

[https://www.youtube.com/watch?v=d\\_qvLDhkg00&ab\\_channel=3Blue1Brown](https://www.youtube.com/watch?v=d_qvLDhkg00&ab_channel=3Blue1Brown)

Extract Audio for this video

Extract Text for this video

Summarize the video using:

- ☒ BART  
☐ Flan-T5  
☐ LLama-2

Summarize using the above selected model

Try asking a query based on the video

what is this video about?

Find the answer

Here is the answer to your query

The video is about a mathematical concept related to probability distributions, specifically the Gaussian or normal distribution. The author discusses a two-step approach to proving a theorem, presumably the Central Limit Theorem, which states that the distribution of the sum of a large number of independent, identically distributed variables with finite variance tends towards a Gaussian distribution, regardless of the original distribution of the variables.

The first step mentioned is to show that all different finite variance distributions tend towards a single universal shape through a process of repeated convolutions. This involves technical concepts such as moment generating functions and is more abstract.

The second step, which is the focus of the video, is to demonstrate that the convolution of two Gaussian distributions results in another Gaussian distribution. This means that a Gaussian distribution is a fixed point in the process of repeated convolutions and, therefore, must be the universal shape that other distributions are tending towards.

The video also connects this mathematical proof to the rotational symmetry of the Gaussian distribution graph, the Herschel Maxwell derivation of a Gaussian, and the appearance of the mathematical constant  $\pi$  in the formula for the Gaussian distribution. Additionally, the author mentions an alternative approach involving entropy, brought to their attention by a channel supporter, Dr. Vide Quinter.

The images provided show graphs related to the Gaussian distribution and its convolution, illustrating the mathematical concepts discussed in the video. The last image appears to be a credits or acknowledgments section, thanking patrons for their support of the channel.

The metadata indicates that the video is titled "A pretty reason why Gaussian + Gaussian = Gaussian" and is created by the YouTube channel 3Blue1Brown, which is known for explaining complex mathematical concepts with visualizations. The video has 736,740 views at the time of the metadata snapshot.

Figure A8: Query Response based on the Video by RAG