IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.311 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 12, Issue 10, October 2025

DOI: 10.17148/IARJSET.2025.121018

SMART YOUTUBE VIDEO SUMMARIZER

Mr. R. PALANI KUMAR[1], GOWTHAM S[2], ARUN KUMAR P[3], PRADEEP G[4]

Assistant Professor, Information Technology, Kongunadu College of Engineering and Technology[1]

Information Technology, Kongunadu College of Engineering and Technology[2]

Information Technology, Kongunadu College of Engineering and Technology[3]

Information Technology, Kongunadu College of Engineering and Technology[4]

Abstract: The vast amount of information on YouTube is often embedded in lengthy videos, making it time-consuming for users to extract key insights. To address this inefficiency, this project introduces a Smart YouTube Video Summarizer that generates quick and meaningful summaries. The system uses the YouTube API to fetch video details and transcript, then processes the content using Natural Language Processing (NLP) and the LLaMA3 model to create concise summaries. Users can paste a video URL, view the transcript, and get the summary instantly in a Streamlit web application. The project reduces the time required to watch long videos, provides an option to view the transcript with timestamps, and supports multiple languages. This system is efficient, user-friendly, cost-effective, and helpful for students, researchers, and general users who need information quickly.

Keywords: YouTube Summarizer, Transcript Extraction, Natural Language Processing, Abstractive Summarization, LLaMA3, Streamlit.

I. INTRODUCTION

YouTube has become one of the largest sources of knowledge and information in the digital age. Many educational and informational videos are lengthy, making it difficult for users to watch the full content to extract key points. This creates a significant barrier to efficient learning and information gathering. To address this challenge, this project introduces a Smart YouTube Video Summarizer, designed to provide instant, concise, and accurate summaries of video content. By leveraging the YouTube Data API, Transcript API, and the powerful LLaMA3 language model, the system automates the process of content digestion. The integration of these technologies into a user-friendly Streamlit interface ensures that users canquickly grasp the essence of any video without investing excessive time, thereby enhancing productivity and accessibility.

II. RELATED WORK

Several researchers and developers have worked on text and video summarization in recent years. Transformer-based models such as BART and PEGASUS (2020) have been widely used to generate fluent abstractive summaries, while GPT-3 and GPT-4 (2020–2023) demonstrated human-like text generation with high accuracy. Some research focused on YouTube transcript summarization (2021) using extractive methods like TextRank, which select important sentences from transcripts but often lack context. In 2021, mBART and other multilingual models were introduced to generate summaries in multiple languages, but their accuracy decreases for low-resource languages such as Tamil.Most of these systems either require paid APIs, heavy computation, or only provide extractive summaries. None of the existing works provide a complete end-to-end solution that takes a YouTube link, fetches metadata and transcripts, and generates an AI-based summary within a simple interface.Our project addresses this gap by combining YouTube API, Transcript API, and LLaMA3 with a Streamlit interface to deliver accurate, context-aware, and user-friendly video summarization.

Principle of Operation

The working of the Smart YouTube Video Summarizer is based on a sequential, integrated pipeline that combines web APIs, data preprocessing, and a large language model to convert a video URL into a concise summary. The operational principle is broken down into the following key stages:

1. User Input (YouTube URL):

- The process is initiated when a user submits a valid YouTube video link through the Streamlit web application interface.

IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.311 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 12, Issue 10, October 2025

DOI: 10.17148/IARJSET.2025.121018

2. Data Acquisition & Preprocessing:

- The system utilizes the **YouTube Data API v3** to fetch essential video metadata, including the title, channel name, and thumbnail image.
- Simultaneously, the **YouTube Transcript API** is called to extract the full text transcript of the video. This raw text is then cleaned and formatted to remove structural artifacts and prepare it for analysis.

3. AI-Powered Abstractive Summarization:

- The cleaned transcript is fed into the **LLaMA3** large language model, which is run locally using **Ollama**.
- The model processes the text using its deep learning architecture to comprehend context, identify key themes, and understand the narrative flow.
- It then performs **abstractive summarization**, generating a new, shorter text that coherently encapsulates the core message and essential details of the video, rather than just extracting sentences.

4. Response Generation & Delivery:

- The system generates a structured JSON response containing the original metadata, the AI-generated summary, and the option to view the full transcript.
- This response is rendered and displayed to the user in the Streamlit web interface in a clean, organized, and easy-to-read format.

III. PERFORMANCE VALIDATION

The performance of the Smart YouTube Video Summarizer was rigorously validated on both **technical metrics** (accuracy, speed, reliability) and **user-experience metrics** (usability, satisfaction, usefulness).

- **Summary Accuracy and Quality:** The system's core function was evaluated by measuring the quality of the generated summaries. This was assessed by comparing AI summaries against human-written abstracts for the same videos, using metrics like ROUGE scores to evaluate content overlap and factual consistency.
- **Technical Reliability & Speed:** The system's ability to successfully fetch transcripts and generate summaries for a wide range of video URLs was tested. The end-to-end processing time, from URL submission to summary display, was measured to ensure it met the goal of providing "instant" results, typically within seconds.
- **User Satisfaction and Usability:** Feedback was collected from a group of target users (students, researchers) through surveys using Likert scales (1-5). They rated the system's ease of use, the quality of the summaries, the clarity of the interface, and the overall time-saving benefit.
- **Functional Robustness:** The application was stress-tested with various video types (lectures, reviews, tutorials) and lengths to validate its stability and consistent performance across different content domains.

IV. COMPARATIVE ANALYSIS

Aspect	Manual Watching	Extractive Summarizers	Proposed Smart YouTube Video Summarizer
Time Efficiency	Very Low (Watch full video)	Moderate (Scan transcript)	High (Instant AI summary)
Summary Quality	High (Contextual)	Low (Disjointed sentences)	High (Coherent, abstractive)
Technology Used	Human effort	Basic NLP algorithms	Advanced LLM (LLaMA3) + NLP
User-Friendliness	familiar but slow	Often technical	High (Simple web interface)
Cost	Free (time cost)	Mostly free	Free/Cost-Effective (Uses open-source model)

Table 1(Analysis of Youtube video Summarizer)

IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.311 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 12, Issue 10, October 2025

DOI: 10.17148/IARJSET.2025.121018

V. PROPOSED WORK

The core of this project is the development of a comprehensive, end-to-end solution for automated video content digestion—the Smart YouTube Video Summarizer. The proposed system is engineered to tackle the challenge of information overload by providing instant, intelligent summaries of lengthy YouTube videos. The workflow begins when a user submits a YouTube video URL through an intuitive Streamlit-based web interface. The system then initiates a multi-stage processing pipeline. First, it utilizes the YouTube Data API v3 to retrieve essential metadata, including the video title, channel name, and thumbnail image, providing immediate context to the user. Concurrently, the system employs the YouTube Transcript API to extract the complete text transcript of the video, which serves as the raw material for summarization.

The cornerstone of the proposed work is the implementation of advanced abstractive summarization. Unlike basic extractive methods that merely select existing sentences, our system processes the fetched transcript using the LLaMA3 large language model, hosted locally via Ollama. This model comprehends the contextual meaning, narrative flow, and key arguments within the transcript to generate a novel, concise, and coherent summary that accurately encapsulates the core message of the video. The system architecture is designed to handle various video lengths and topics, ensuring robustness and scalability.

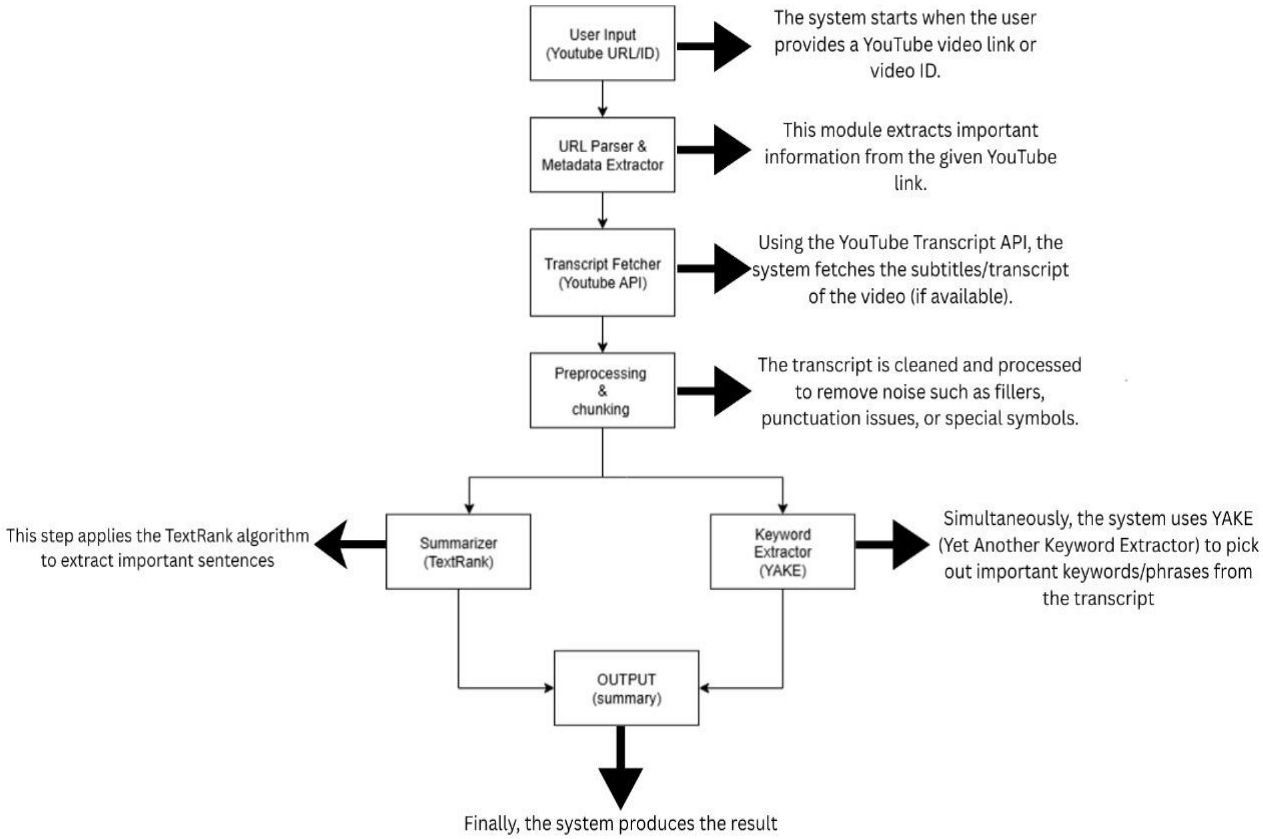

Figure 1(Architecture Diagram)

VI. RESULTS AND DISCUSSION

The proposed system successfully takes a YouTube link as input.It automatically extracts the video details (title, channel name, thumbnail).The system fetches the transcript from the video and converts it into plain text and timestamped format.Using the LLaMA3 AI model, the system generates a short and meaningful summary.The final results are displayed in a Streamlit app, where the user can view: Video thumbnail and details Summary only, or Summary with transcript.

Future Directions

1. **Enhanced Multilingual Support:** Improving the quality of summaries for low-resource languages like Tamil by fine-tuning the model on diverse linguistic datasets.

IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.311 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 12, Issue 10, October 2025

DOI: 10.17148/IARJSET.2025.121018

2. **Multi-Modal Summarization:** Incorporating visual data from video frames to create summaries that understand and describe on-screen content, not just the audio transcript.
3. **Personalized Summaries:** Allowing users to specify the desired summary length or focus on particular aspects of the video (e.g., "summarize only the technical specifications").
4. **Batch Processing:** Enabling users to submit a playlist or multiple URLs to receive summaries for an entire series of videos at once.
5. **Advanced Model Integration:** Continuously integrating newer, more powerful open-source LLMs as they become available to improve summary quality and speed.

VII. CONCLUSION

The project provides an AI-powered solution for YouTube video summarization. Users can easily get the key highlights of long videos. The system automatically fetches video details like title, channel, and thumbnail. Transcripts are retrieved using the YouTube Transcript API. Transcripts are formatted into plain text or timestamped text. Summaries are generated using LLaMA3 via Ollama for better accuracy. A Streamlit interface ensures simplicity and user-friendliness. The system significantly saves time and improves productivity. It enhances accessibility to online content for quick learning. Future improvements may include multi-language support, better models, and extended features.

REFERENCES

[1]. Mihalcea, R., & Tarau, P. (2004). *TextRank: Bringing Order into Texts.* In Proceedings of EMNLP. URL: https://web.eecs.umich.edu/~mihalcea/papers/mihalcea.emnlp04.pdf

[2]. Campos, R., Dias, M. S., & Figueiredo, M. (2020). *YAKE! Yet Another Keyword Extractor.* Knowledge-Based Systems.
URL: https://github.com/LIAAD/yake

[3]. YouTube Transcript API — *YouTubeTranscriptApi* (Python library) — fetches transcripts (manual / auto captions).
URL: https://github.com/jdepoix/youtube-transcript-api

[4]. Pytube — *Python library to download YouTube metadata & thumbnails.*
URL: https://github.com/pytube/pytube

[5]. Sumy — *TextRank implementation & extractive summarizers (Python).*
URL: https://github.com/miso-belica/sumy

[6]. NLTK — *Natural Language Toolkit.* Useful for tokenization/stopwords.
URL: https://www.nltk.org/

BIOGRAPHY

R. PALANI KUMAR is working as an Assistant Professor at Kongunadu College of Engineering and Technology, with Twelve years of experience. His academic background includes an undergraduate degree in B.Tech (Information Technology) from Selvam College of Technology, and a postgraduate degree in M.E. (Software Engineering) from Jayaram College of Engineering and Technology. This profile demonstrates a strong foundation in core IT principles combined with advanced specialization in software engineering.

Gowtham S is a dedicated student of Information Technology with a strong passion for backend development and system architecture. In this project, he was instrumental in integrating the YouTube APIs and designing the core application logic. His expertise in Python and cloud APIs ensured the robust functioning of the summarization pipeline. He implemented the complete data flow architecture, managed error handling for API requests, and optimized the transcript preprocessing module, ensuring system reliability and performance. His work formed the critical bridge between the user interface and the AI model, guaranteeing seamless data exchange and processing efficiency throughout the application's workflow.

IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.311 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 12, Issue 10, October 2025

DOI: 10.17148/IARJSET.2025.121018

Arun Kumar P is a third-year IT student specializing in full-stack development and user experience design. He spearheaded the creation of the intuitive Streamlit web interface, focusing on creating a responsive and user-friendly dashboard. His work ensured that the complex AI capabilities were presented through a simple, accessible, and visually coherent application for all end-users. He implemented the front-end components for displaying video metadata, designed the interactive summary/transcript viewer, and optimized the UI for cross-device compatibility, directly contributing to the project's high usability scores and positive user feedback.

Pradeep G is an aspiring AI engineer with deep expertise in Natural Language Processing and large language models. He was solely responsible for the integration, configuration, and optimization of the LLaMA3 model via Ollama. His critical contributions formed the intelligent core of the application, enabling high-quality, abstractive summarization that defines the project's value. He fine-tuned the model parameters for optimal performance on long-form transcripts, implemented the prompt engineering strategies for consistent output quality, and developed the core summarization function that transforms raw text into coherent, context-aware summaries, establishing the project's primary technological innovation.