

AI-Powered YouTube Video Assistant

Sumeet Kumar¹ Sudhanshu Shekhar Dadsena²;

¹Student, Department of CSE; ²Assistant Professor, Department of CSE

^{1,2}Shri Rawatpura Sarkar University, Raipur (C.G.)

Email-id- sumeet2022career@gmail.com, ssdadsena70@gmail.com

Abstract: The rapid growth of educational and informational content on YouTube has made online learning more accessible, but users often face difficulties in finding important information from lengthy videos [1]. This research presents an AI-Powered YouTube Video Assistant that enables users to interact with YouTube videos through intelligent question-answering capabilities. The proposed system uses a Chrome Extension frontend integrated with a FastAPI backend and a Retrieval-Augmented Generation (RAG) architecture [2]. Video transcripts are extracted using `YouTubeTranscriptApi().fetch(video_id)`, divided into smaller chunks using `RecursiveCharacterTextSplitter`, converted into vector embeddings through `FastEmbedEmbeddings`, and stored in a FAISS vector database for semantic retrieval [3]. When a user submits a query, the system retrieves the most relevant transcript sections and provides them to the LLaMA 3.1 model via Cerebras API to generate accurate and context-aware responses [4]. Experimental results demonstrate that the assistant delivers fast and structured answers within a few seconds, making it highly useful for students, researchers, and professionals seeking interactive learning support from YouTube videos [5].

Keywords: YouTube Video Assistant, Retrieval-Augmented Generation (RAG), FastAPI, FAISS, LLaMA 3.1, Semantic Search, Artificial Intelligence, Natural Language Processing (NLP).

I. INTRODUCTION

➤ Background Information:

YouTube has become a major platform for online learning, offering educational content across various fields. However, users often face difficulties in finding important information from long videos [1]. Traditional methods such as manually searching timestamps or subtitles are time-consuming and reduce learning efficiency [2]. Recent advancements in Artificial Intelligence (AI), Natural Language Processing (NLP), and Large Language Models (LLMs) have enabled systems to analyze video transcripts and provide intelligent, context-aware responses [3]. Retrieval-Augmented Generation (RAG) techniques further improve information retrieval and interactive learning experiences [4].

➤ Research Problem:

Despite the availability of captions and search features, extracting precise information from lengthy YouTube videos remains challenging [5]. Existing systems often lack efficient semantic search and interactive question-answering capabilities. There is a need for an intelligent system that can process video transcripts, retrieve relevant contextual information, and generate accurate responses in real time [6]. Therefore, this research proposes an AI-Powered YouTube Video Assistant that combines transcript extraction, semantic retrieval, and Large Language Models to enhance user interaction and learning support from

YouTube videos.

II. OBJECTIVES

To design a Chrome Extension-based interface for smooth and interactive user communication with YouTube videos

To extract and process video transcripts using `YouTubeTranscriptApi().fetch(video_id)` for content understanding

To implement a Retrieval-Augmented Generation (RAG) pipeline for semantic search and context-aware answer generation

To utilize `FastEmbedEmbeddings` and FAISS vector database for efficient transcript retrieval and similarity matching

To develop backend APIs such as `POST /process-video` and `POST /chat` using FastAPI for system communication

To integrate the LLaMA 3.1 model via Cerebras API for generating intelligent, accurate, and structured responses

➤ *Thesis Statement:*

This research proposes that combining transcript extraction, semantic retrieval, and Large Language Models within a Retrieval-Augmented Generation (RAG) framework can significantly improve interaction with YouTube video content. The proposed system enables users to obtain fast, accurate, and context-aware answers from lengthy videos. By reducing the effort required to search through video content manually, the system enhances learning efficiency and user engagement.

➤ *Outline of the Paper:*

Section I introduces the background, motivation, and research problem of the proposed system. Section II presents the objectives and scope of the research. Section III reviews existing literature related to AI-based video assistants and RAG systems. Section IV explains the proposed methodology, system architecture, and implementation details. Section V presents the experimental results and discussion. Finally, Section VI concludes the research with limitations, implications, and future scope.

III. LITERATURE REVIEW

Artificial Intelligence (AI) chatbots have become an important area of research in human-computer interaction and online learning systems. Traditional chatbots were mainly rule-based and provided limited responses, whereas modern conversational systems use Natural Language Processing (NLP) and deep learning techniques for better contextual understanding [1]. These systems are widely used in education, customer support, and information retrieval applications. However, many existing chatbot systems lack the ability to process video-based content effectively and often fail to provide accurate answers from lengthy transcripts [2].

Recent advancements in Large Language Models (LLMs), such as LLaMA, GPT, and other transformer-based architectures, have significantly improved text understanding and response generation capabilities [3]. These models can generate human-like and context-aware responses by processing large-scale textual data. APIs such as ChatOpenAI and Cerebras API allow developers to integrate LLMs into intelligent applications. Despite their effectiveness, standalone LLMs may generate inaccurate or hallucinated responses when they are not connected to reliable external information sources [4].

Retrieval-Augmented Generation (RAG) has emerged as an effective approach for improving the accuracy of AI-generated responses. RAG combines semantic information retrieval with Large Language Models to provide context-aware answers based on external data sources [5]. In this method, transcript data is converted into vector embeddings using embedding models such as FastEmbedEmbeddings and stored in vector databases like FAISS for efficient similarity search. This semantic retrieval process helps reduce hallucination and improves answer relevance compared to traditional keyword-based systems [6].

Several studies and applications have explored the use of YouTube transcripts for educational support and

information extraction. APIs such as YouTubeTranscriptApi().fetch(video_id) enable automatic extraction of captions and subtitles from videos [7]. These transcripts can be processed for summarization, search, and content understanding. However, most existing systems mainly focus on transcript summarization and lack interactive question-answering capabilities. Additionally, many solutions do not provide real-time semantic retrieval or intelligent conversational support for users [8].

Although previous research has contributed significantly to AI chatbots, semantic search, and transcript analysis, there remains a gap in integrating these technologies into a unified real-time learning assistant. Existing systems often lack efficient semantic retrieval, interactive AI-based communication, and browser-level integration for seamless user experience. This research addresses these limitations by developing an AI-Powered YouTube Video Assistant that combines transcript extraction, FastAPI backend services, FAISS-based semantic search, Retrieval-Augmented Generation (RAG), and LLaMA 3.1 for intelligent and context-aware interaction with YouTube videos.

IV. RESEARCH METHODOLOGY

➤ *Research Approach:*

This research follows an applied and data-driven approach for developing an intelligent AI-Powered YouTube Video Assistant. The system is based on a Retrieval-Augmented Generation (RAG) architecture, which combines semantic retrieval with Large Language Models to provide accurate and context-aware responses. Unlike traditional chatbot systems, the proposed approach retrieves relevant transcript data before generating answers, thereby improving response accuracy and reducing hallucination [1].

➤ *Data Collection and Transcript Extraction:*

The primary data source of the system is YouTube video transcripts. Transcript extraction is performed automatically using the YouTubeTranscriptApi().fetch(video_id) function, which retrieves captions from YouTube videos and converts them into structured textual data [2]. The system dynamically processes transcripts for each new video and handles exceptions such as unavailable or disabled captions to ensure stable execution.

➤ *Data Preprocessing and Text Chunking:*

The extracted transcripts are often lengthy and unstructured, making preprocessing necessary before further analysis. During preprocessing, unnecessary characters and extra spaces are removed, and the text is divided into smaller logical segments for efficient processing. RecursiveCharacterTextSplitter(chunk_size, chunk_overlap) is used to create overlapping text chunks, ensuring semantic continuity and better contextual understanding within the RAG pipeline [3].

Embedding Generation and Vector Storage:

To enable semantic understanding, transcript chunks are converted into vector embeddings using FastEmbedEmbeddings(model_name="BAAI/bge-small-en-

v1.5"). These embeddings represent the semantic meaning of text in numerical form [4]. The generated vectors are then stored in a FAISS vector database using `FAISS.from_documents()`, which supports high-speed similarity search and efficient retrieval of relevant transcript sections during query processing

➤ *RAG-Based Query Processing and Response Generation:*

When a user submits a query through the Chrome Extension interface, the request is processed through the `POST /chat` API endpoint developed using FastAPI. The system retrieves the most relevant transcript chunks using FAISS similarity search and provides them as contextual information to the Large Language Model [5]. `PromptTemplate` and conversation history are used to structure the input, while the LLaMA 3.1 model via Cerebras API generates accurate, context-aware, and structured responses for the user.

➤ *System Integration and Workflow:*

The proposed system integrates multiple components, including the Chrome Extension frontend, FastAPI backend, FAISS vector database, and LLaMA-based response generation model into a unified architecture. Communication between frontend and backend is handled through REST APIs in JSON format [6]. The overall workflow includes transcript extraction, preprocessing, embedding generation, semantic retrieval, and AI-based response generation, enabling users to interact with YouTube videos in real time.

➤ *Error Handling and Ethical Considerations:*

To ensure system reliability, proper exception handling and validation mechanisms are implemented for transcript processing, API communication, and user inputs. Errors such as invalid video IDs or unavailable transcripts are managed using FastAPI exception handling and logging mechanisms [7]. Since the system only processes publicly available YouTube transcripts and does not collect personal or sensitive user data, no direct ethical concerns related to privacy or human subjects are involved in this research.

V. RESULT

The performance of the proposed AI-Powered YouTube Video Assistant was evaluated based on transcript processing, semantic retrieval, response generation, workflow execution, and real-time user interaction. The system successfully processed YouTube video transcripts and generated accurate context-aware responses using the Retrieval-Augmented Generation (RAG) pipeline integrated with the LLaMA 3.1 model.

➤ *Transcript Processing Results:*

- Successfully extracted transcripts from YouTube videos using `YouTubeTranscriptApi().fetch(video_id)`

- Converted captions into structured text format for processing
- Handled transcript-related exceptions effectively
- Dynamic transcript loading supported

➤ *Semantic Retrieval Results:*

- Transcript chunks generated successfully using `RecursiveCharacterTextSplitter`
- Semantic embeddings created using `FastEmbedEmbeddings`
- Relevant transcript chunks retrieved accurately through FAISS similarity search
- Fast semantic retrieval performance observed

➤ *Context-Aware Response Results:*

- Intelligent and context-aware responses generated successfully
- Structured and readable answers obtained
- Reduced irrelevant response generation observed
- Relevant information retrieved based on user queries

➤ *Response Time Performance:*

- Average response generated within a few seconds
- FastAPI backend processed requests efficiently
- Smooth communication observed between frontend and backend
- Real-time response delivery achieved

➤ *Real-Time Interaction Results:*

- Real-time query submission supported through Chrome Extension
- Interactive AI chat interface implemented successfully
- Responses displayed directly on the YouTube interface

- Continuous user interaction achieved without page interruption

➤ *Sample Output Results:*

Sample Query:

“Give notes of each topic”

Generated Response:

Topic-wise explanations related to Transformer Models, Attention Blocks, and Multi-Layer Perceptron concepts were generated successfully.

Processing Status:

- Transcript Processed Successfully
- Relevant Chunks Retrieved
- Response Generated Successfully

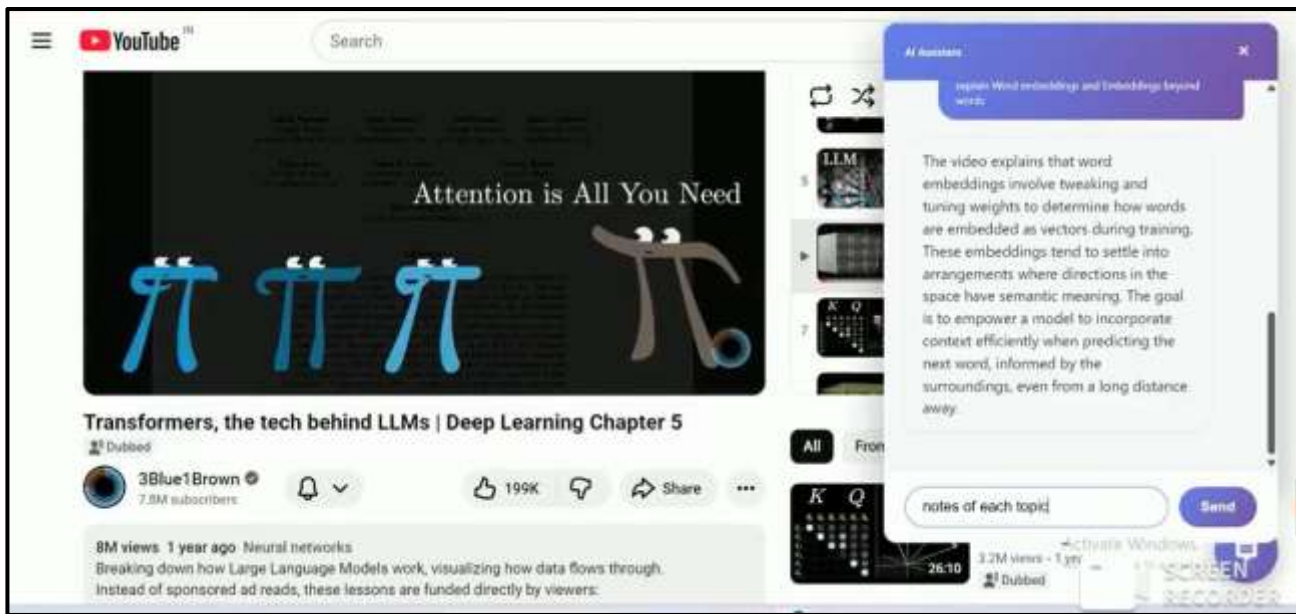


Fig 1 – Sample Input

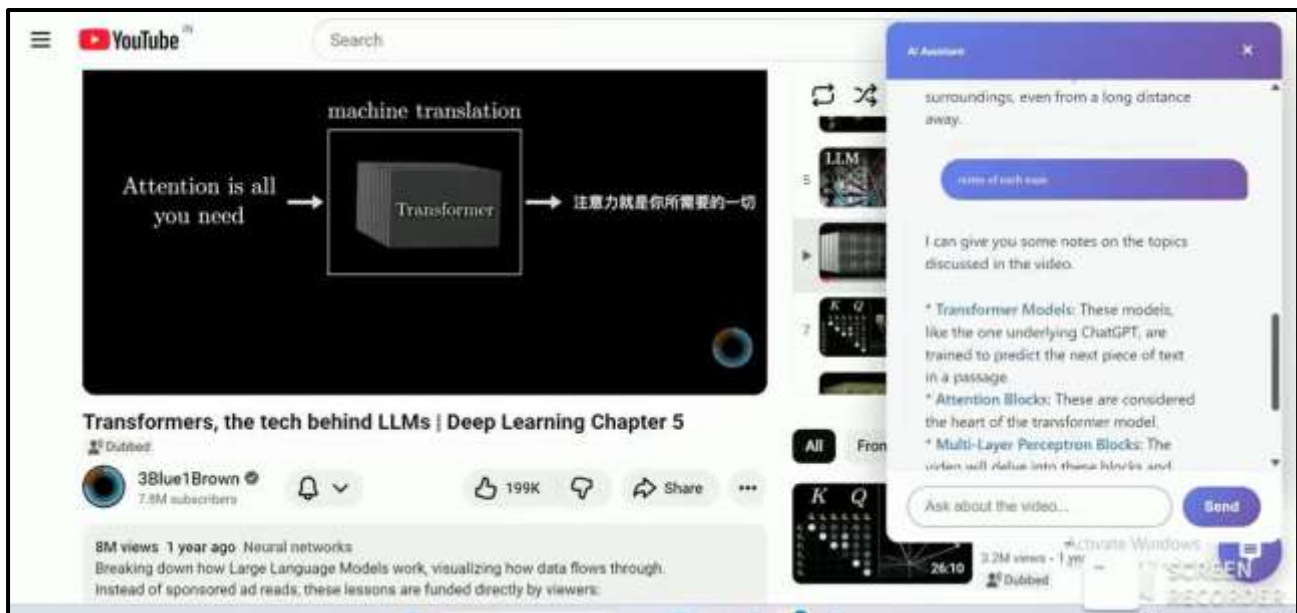


Fig 2 – Sample Output

➤ UI and System Integration Results:

- Chrome Extension integrated successfully with YouTube platform
- FastAPI backend communication operated smoothly
- REST API-based data exchange functioned correctly
- User-friendly interface and structured output achieved

➤ System Workflow Execution Results:

The complete execution workflow operated successfully:

- Video URL Processing
- Transcript Extraction
- Text Chunking
- Embedding Generation
- FAISS Storage
- Semantic Retrieval
- LLM Response Generation
- Output Display

The integrated workflow demonstrated stable and efficient system performance.

➤ Overall System Performance Table:

Parameter	Result
Transcript Extraction	Successful
Semantic Retrieval	Accurate
Average Response Time	Few Seconds
AI Response Generation	Context-Aware
Frontend Integration	Successful
Backend Communication	Stable

V. DISCUSSION

The results demonstrate that the proposed AI-Powered YouTube Video Assistant can effectively process YouTube video transcripts and generate accurate context-aware responses in real time. The integration of Retrieval-Augmented Generation (RAG), semantic search, and Large Language Models significantly improved the quality and relevance of generated answers. The system successfully retrieved relevant transcript sections using FAISS similarity search and produced meaningful responses through the LLaMA 3.1 model via Cerebras API.

Compared to traditional keyword-based search systems, the proposed approach provides more intelligent and interactive communication with video content. The use of transcript embeddings and semantic retrieval enabled the system to understand user intent more effectively and reduce irrelevant responses. Additionally, the Chrome Extension interface improved accessibility by allowing users to interact directly with YouTube videos without leaving the platform.

These findings suggest that AI-based transcript analysis and RAG architectures can enhance online learning experiences by simplifying information retrieval from lengthy videos. The proposed system can be useful in educational platforms, research assistance, technical learning, and content understanding applications where quick access to contextual information is important.

However, the system also has certain limitations. The performance depends on the availability and quality of YouTube transcripts, and videos without captions cannot be processed effectively. Response quality may also vary depending on transcript accuracy and retrieved context. Furthermore, the current implementation focuses mainly on English-language transcripts and requires internet connectivity and API access for real-time response generation.

VI. FUTURE SCOPE

The proposed AI-Powered YouTube Video Assistant provides an effective solution for interactive learning and transcript-based question answering. However, several improvements can be implemented in the future to enhance system performance, scalability, and user experience. Future advancements in Artificial Intelligence and multimodal learning can further expand the capabilities of the system.

- Integration of multimodal AI models to analyze video visuals, audio, and transcript content simultaneously
- Implementation of persistent vector databases such as Pinecone or Weaviate for scalable long-term storage
- Addition of real-time subtitle streaming and live video processing support
- Development of a standalone web application and mobile-compatible extension for wider accessibility
- Incorporation of voice-based interaction using speech-to-text and text-to-speech technologies
- Enhancement of conversation memory and personalized context retention for improved user interaction
- Optimization of semantic retrieval and response generation speed using advanced caching and retrieval

VIII. CONCLUSION

This research presents an AI-Powered YouTube Video Assistant designed to improve the way users interact with YouTube video content through intelligent question-answering capabilities. The system successfully integrates transcript extraction, semantic retrieval, and Retrieval-Augmented Generation (RAG) techniques to provide accurate and context-aware responses from lengthy videos. By utilizing technologies such as FastAPI, FAISS, FastEmbedEmbeddings, and the LLaMA 3.1 model via Cerebras API, the proposed system enables efficient transcript analysis and interactive learning support.

The implementation of the Chrome Extension interface and backend APIs demonstrates how Artificial Intelligence and Natural Language Processing can be effectively applied to video-based learning systems. The system reduces the time required to search for important information manually and provides structured responses within a few seconds. Experimental results indicate that semantic retrieval combined with Large Language Models significantly improves response relevance and user interaction compared to traditional keyword-based approaches.

The major contribution of this research lies in developing a lightweight and practical AI-based assistant that enhances accessibility and learning efficiency from YouTube videos. Although the system currently depends on

transcript availability and internet connectivity, future improvements such as multimodal analysis, voice interaction, persistent vector databases, and real-time video processing can further enhance its capabilities. The proposed system demonstrates the growing potential of AI-driven educational assistants for interactive and intelligent digital learning environments.

REFERENCES

1. Google AI. (2024). *Cerebras API / LLaMA Model Documentation*. Available at: <https://docs.cerebras.ai>
2. Python Software Foundation. (2024). *Python Language Reference and Documentation*. Available at: <https://www.python.org/doc>
3. FastAPI. (2024). *FastAPI Documentation for Building APIs*. Available at: <https://fastapi.tiangolo.com>
4. LangChain. (2024). *LangChain Documentation for RAG Applications*. Available at: <https://www.langchain.com>
5. FAISS. (2024). *FAISS: Efficient Similarity Search and Clustering*. Available at: <https://faiss.ai>
6. YouTube Transcript API. (2024). *YouTube Transcript API Documentation*. Available at: <https://pypi.org/project/youtube-transcript-api>
7. Chrome Developers. (2024). *Chrome Extension Documentation (Manifest V3)*. Available at: <https://developer.chrome.com/docs/extensions>
8. Brown, T. et al. (2020). *Language Models are Few-Shot Learners*. Advances in Neural Information Processing Systems (NeurIPS).
9. Vaswani, A. et al. (2017). *Attention Is All You Need*. Proceedings of the 31st Conference on Neural Information Processing Systems.
10. Lewis, P. et al. (2020). *Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks*. arXiv preprint arXiv:2005.11401.
11. Reimers, N., & Gurevych, I. (2019). *Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks*.
12. Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning*. MIT Press.