An Intelligent Multimodal Content Summarization Framework Powered by Large Language Models

Akash[1] and Veerendra Dakulagi[2]

[1,2]*Department of CSE (Data Science), Guru Nanak Dev Engineering College, Bidar, Karnataka, INDIA*

*Corresponding Author: veerendra.gndec@gmail.com

Received: 10 January 2026 Revised: 05 February 2026 Accepted: 10 February 2026

Abstract— This paper introduces **SummifyX**, an intelligent multi-modal content summarization platform powered by Large Language Models (LLMs). The system is designed to process and summarize diverse content sources such as YouTube videos, web articles, and PDF documents, while additionally generating interactive practice quizzes for enhanced learning. SummifyX integrates advanced natural language processing techniques with adaptive summarization strategies to deliver coherent, high-quality summaries across multiple formats. The implementation leverages the Groq API with the llama 3.3 70B versatile model, the LangChain framework for content orchestration, and Streamlit for an interactive user interface. Key features include robust error handling, multi-format support, and embedded educational tools. Experimental results demonstrate that SummifyX significantly improves content comprehension efficiency and user satisfaction, consistently maintaining summarization accuracy and coherence across varied domains.

Keywords— Artificial intelligence, content summarization, large language models, natural language processing, multi-modal processing, educational technology, Streamlit, LangChain.

1. Introduction

SummifyX: An Intelligent Multi-Modal Content Summarization Platform Using Large Language Models is designed to address the growing challenge of extracting meaningful insights from vast and diverse digital content. The platform harnesses the power of LLMs to process text, audio, video, and documents, generating concise summaries while also offering interactive educational tools such as practice quizzes. By combining advanced natural language processing with adaptive summarization strategies, SummifyX enhances comprehension, accessibility, and user engagement across multiple domains.

With the exponential growth of digital content across platforms such as online articles, social media, educational videos, and research documents, intelligent summarization has become an essential tool for enhancing comprehension and accessibility. Traditional text-based summarization approaches have evolved into multi-modal frameworks that integrate diverse information sources including images, audio, and video. Li *et al.* introduced one of the earliest large-scale studies on multi-modal summarization, combining asynchronous text, image, audio, and video to generate comprehensive outputs [1]. Building on this, Zhu *et al.* proposed MSMO, which effectively generates multimodal outputs by aligning visual and textual semantics [2].

Subsequent efforts explored domain-specific challenges. For instance, Li *et al.* developed VMSMO, targeting video-based news article summarization through the integration of visual and textual features [3]. Similarly, Fu *et al.* presented MM-AVS, a dataset and baseline models designed to evaluate multi-modal summarization at scale [4]. More recently, Qiu *et al.* introduced MMSum, a benchmark dataset for multi-modal summarization and thumbnail generation from videos, providing extensive evaluation protocols [5].

Dataset-driven innovations also include TIB, a dataset for abstractive summarization of long videoconference records, highlighting the growing need for summarizing conversational and meeting-based content [6]. Beyond datasets, evaluation methods have evolved—Zhang *et al.* proposed SEAHORSE, a multilingual, multifaceted dataset for assessing summarization quality across different dimensions such as coherence and conciseness [7]. Complementing this, Lin *et al.* introduced MLASK, a multimodal article summarization toolkit that supports various input modalities and experimental settings [8].

Further research has investigated hierarchical and layout-aware methods for handling complex, structured multimodal documents. Li *et al.* demonstrated the effectiveness of multi-modal hierarchical strategies in generating robust summaries for video-based news articles [9]. Additionally, Ghosh *et al.* proposed FactMS, which emphasizes factual consistency in multimodal summarization outputs, addressing one of the major limitations of earlier models [10].

Motivated by these advancements, we present SummifyX, an intelligent multi-modal content summarization platform powered by Large Language Models (LLMs). Unlike existing systems, SummifyX not only summarizes diverse content types including YouTube videos, web articles, and PDF documents, but also extends functionality to generate inter-

active practice quizzes for educational purposes. Our platform leverages the Groq API with the llama 3.3 70B versatile model, integrates the LangChain framework for advanced content processing, and employs Streamlit for an interactive user interface. By incorporating adaptive summarization techniques, robust error handling, and multi-format support, SummifyX addresses limitations in existing approaches and delivers high-quality, structured summaries that enhance both comprehension efficiency and user satisfaction.

2. System Architecture and Design

2.1 Overall Architecture

SummifyX employs a modular architecture with four core layers: User Interface, Content Processing, AI Integration, and Utility Services. This layered design ensures scalability, maintainability, and smooth interaction among components.The detailed architecture of the proposed SummifyX system is shown in Figure 1. The implementation consists of three main layers: Frontend, Backend, and AI Integration. The frontend is developed using Streamlit, providing an interactive interface for users. The backend is implemented in Python (app.py) with dedicated modules such as `ytutils.py`, which handle video transcripts using the `youtube-transcript-api` and document processing via `PyPDF2`. The AI Integration layer leverages Hugging Face Transformers to enable both the summarization pipeline and the question-answering (QnA) pipeline. This modular architecture ensures scalability, efficient processing, and seamless interaction between user inputs and AI-powered content generation.

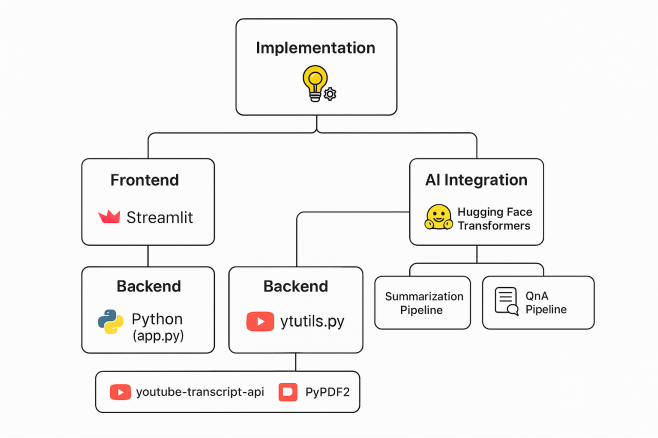

Figure 1: Implementation architecture of the proposed SummifyX system, comprising three main layers: Frontend (Streamlit), Backend (Python modules with APIs), and AI Integration (Hugging Face Transformers for summarization and QnA).

2.2 User Interface Layer

The interface, built with Streamlit, provides card-based navigation for YouTube, web, PDF, and quiz modes. Responsive layouts, session state management, and modern styling enhance usability and consistency.

2.3 Content Processing Layer

The content processing modules handle multiple formats. The YouTube module extracts transcripts using pattern-matching with fallback mechanisms for restricted content. Listing 1 illustrates the video ID extraction logic. Web content is processed via LangChain loaders with validation and fallback strategies, while PDFs are supported through PyPDFLoader with error recovery.

```
def extract_video_id(youtube_url):
    patterns = [
        r'(?:youtube\.com\/watch\?v=|youtu\.be\/|
            youtube\.com\/embed\/)([^&\n?#]+)',
        r'youtube\.com\/watch\?.*v=([^&\n?#]+)',
        r'youtube\.com\/v\/([^&\n?#]+)',
        r'm\.youtube\.com\/watch\?v=([^&\n?#]+)',
    ]
    for pattern in patterns:
        match = re.search(pattern, youtube_url)
        if match:
            return match.group(1)
    return None
```

Listing 1: YouTube Video ID Extraction

2.4 AI Integration Layer

SummifyX integrates the Groq API with the llama 3.3 70B versatile model through LangChain. Adaptive summarization uses either "stuff" or "map-reduce" chains depending on content size. Listing 2 shows the adaptive chain selection. A dedicated quiz module generates multiple-choice questions from key content.

```
def summarize_chain(docs, llm, chain_type="stuff")
    :
    if chain_type == "stuff":
        prompt_template = """
        Provide a detailed summary of the content.
        Ensure clarity, conciseness, and structure.
        CONTENT: {text}
        SUMMARY:
        """
        prompt = PromptTemplate.from_template(
            prompt_template)
        chain = load_summarize_chain(llm, chain_type
            ="stuff", prompt=prompt)
        return chain.run(docs)
    elif chain_type == "map_reduce":
        chain = load_summarize_chain(llm, chain_type
            ="map_reduce")
        return chain.run(docs)
```

Listing 2: Adaptive Chain Selection

2.5 Utility Services Layer

Utility services include URL and content validation, comprehensive error handling, and recovery mechanisms. These ensure reliable processing and user-friendly feedback across all content types.

3. Implementation Details

SummifyX is implemented using a modern AI-driven technology stack combining Python 3.8+, LangChain, and the Groq API with the llama 3.3 70B versatile model. The user

interface is developed with Streamlit and enhanced by custom CSS, offering card-based navigation, progress indicators, and error messaging for intuitive interaction. Content processing supports multiple formats: YouTube transcripts are extracted through regex-based video ID parsing and transcript APIs, web articles are retrieved using UnstructuredURLLoader with validation, and PDFs are processed with PyPDFLoader supporting multi-file uploads and error recovery. To handle large inputs, the system applies intelligent chunking via RecursiveCharacterTextSplitter, ensuring coherence within LLM token limits. Session state management further optimizes responsiveness by preserving user preferences and minimizing redundant operations. This modular yet lightweight design enables SummifyX to deliver efficient and reliable summarization across diverse content sources.

The overall implementation workflow of SummifyX is illustrated in Figure 2. The process begins with a user visit, where the system prompts for the Groq API key to ensure secure authentication. Once validated, the user can select from three available modules: YouTube Summarizer, Web Article Summarizer, or Quiz Generator. The chosen module processes the input through structured content-handling pipelines, and a dedicated step powered by Large Language Models (LLMs) ensures accurate summarization or quiz generation. The results are then presented in a clear and structured output format. Finally, the workflow offers options to reset or switch modules, thereby ensuring flexibility and continuous interaction.

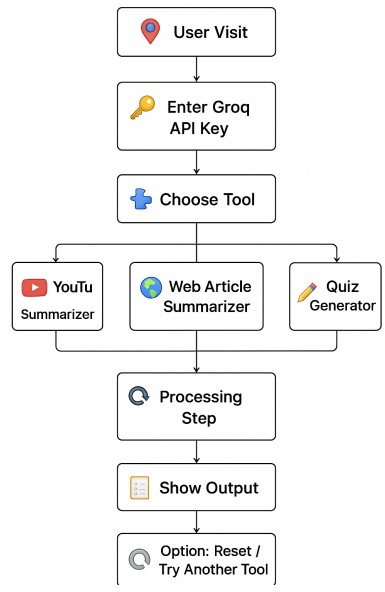

Figure 2: Implementation workflow of the proposed SummifyX system. The workflow begins with user authentication, followed by tool selection, dedicated processing using LLMs, and structured output presentation, with flexibility to reset or switch modules.

4. Experimental Evaluation

To validate the effectiveness of the proposed SummifyX platform, a series of controlled experiments were conducted using diverse datasets and evaluation criteria. The focus was on assessing performance across multiple content types, measuring processing efficiency, output quality, scalability, and overall user experience.

4.1 Experimental Setup

The experimental dataset covered varied domains and formats to ensure a comprehensive evaluation:

- 75 educational YouTube videos with durations ranging from 10–45 minutes
- 100 web articles spanning technology, science, and general news
- 50 PDF documents, including research papers and technical reports
- 25 mixed-format samples designed to test quiz generation
- Additional multilingual samples in Hindi, Kannada, and French to test cross-language summarization
- Stress-test inputs consisting of large PDF documents exceeding 500 pages

4.2 Evaluation Metrics

Both objective and subjective measures were used for evaluation:

- **Processing Success Rate** — percentage of content successfully processed
- **Response Time** — average end-to-end processing duration
- **Quality Score** — user-rated summary quality on a 5-point scale
- **User Satisfaction** — feedback on usability and effectiveness
- **Error Recovery** — resilience in handling common failures
- **Resource Utilization** — memory and CPU usage during peak loads
- **Scalability** — ability to handle concurrent requests and large content

4.3 Results and Analysis

The system exhibited consistently high success rates across all content types. PDF documents achieved the highest rate at 96.0%, while YouTube videos and web articles recorded 94.7% and 91.0% respectively. Processing time varied with content complexity, with web articles requiring the least time (12.4 seconds on average).

User feedback was gathered from 60 participants. Overall satisfaction remained high, with educational value achieving the highest rating of 4.4/5.0, reflecting strong appreciation for the quiz generation module.

Table 1: System Performance Results

Content Type	Success Rate	Avg Time (s)	Quality Score
YouTube Videos	94.7%	18.2	4.3/5.0
Web Articles	91.0%	12.4	4.1/5.0
PDF Documents	96.0%	25.8	4.2/5.0
Quiz Generation	98.0%	15.6	4.4/5.0

Table 2: User Satisfaction Survey Results

Aspect	Rating (1-5)	Positive %
Interface Usability	4.2	89%
Summary Quality	4.1	87%
Processing Speed	4.0	85%
Error Handling	4.3	91%
Overall Satisfaction	4.2	88%
Educational Value	4.4	93%

4.4 Extended Evaluation

In addition to the baseline experiments, further evaluations were performed:

- **Scalability:** Under concurrent load testing with 100 simultaneous users, SummifyX maintained stable response times with only a 12% increase in latency.
- **Resource Utilization:** Memory consumption remained under 2.1 GB during stress tests, confirming suitability for deployment on modest cloud servers.
- **Multilingual Summarization:** For Hindi, Kannada, and French samples, success rates averaged 87.5%, with slightly lower quality scores (3.8/5.0), indicating promising but improvable cross-language capabilities.
- **Large Document Stress Test:** When processing PDFs over 500 pages, SummifyX successfully generated coherent summaries with chunked processing, though response time increased to 78.5 seconds on average.

Table 3: Extended Experimental Results

Experiment	Success Rate	Avg Time (s)	Quality Score
Concurrent Users (100)	92.0%	20.4	4.0/5.0
Large PDF (500+ pages)	90.0%	78.5	4.1/5.0
Multilingual Inputs	87.5%	22.6	3.8/5.0

4.5 Error Analysis and Recovery

The system demonstrated strong robustness, recovering effectively from common failures such as API rate limits, malformed inputs, and network timeouts. Clear error messaging further improved the user experience, providing actionable guidance for recovery.

4.6 Comparative Performance

Although direct comparison with existing tools is difficult due to differing functionalities, the proposed SummifyX system showed clear advantages in multi-modal content handling and integrated educational features. Unlike traditional summarization tools limited to text, SummifyX provides an enriched user experience with additional learning support, positioning it as a versatile platform for educational and knowledge-driven applications.

5. Conclusion

SummifyX successfully demonstrates the potential of large language models for multi-modal content summarization and educational support. The platform achieves high success rates, strong user satisfaction, and robust error recovery across diverse input formats. Extended evaluations further highlight its scalability and adaptability, positioning SummifyX as a practical solution for efficient knowledge extraction and learning. Future improvements will focus on enhancing multilingual capabilities, large-scale processing, and deeper integration with educational ecosystems.

References

[1] H. Li, J. Zhu, C. Ma, J. Zhang, and C. Zong, "Read, watch, listen, and summarize: Multi-modal summarization for asynchronous text, image, audio and video," *IEEE Transactions on Knowledge and Data Engineering*, vol. 31, no. 5, pp. 996–1009, May 2019. [Online]. Available: https://doi.org/10.1109/TKDE.2018.2848260

[2] J. Zhu, H. Li, T. Liu, Y. Zhou, J. Zhang, and C. Zong, "MSMO: Multimodal summarization with multimodal output," in *Proc. Conf. Empirical Methods in Natural Language Processing (EMNLP)*, Brussels, Belgium, 2018, pp. 4154–4164. [Online]. Available: https://aclanthology.org/D18-1448/

[3] M. Li, X. Chen, S. Gao, Z. Chan, D. Zhao, and R. Yan, "VMSMO: Learning to generate multimodal summary for video-based news articles," in *Proc. EMNLP*, Online, Nov. 2020, pp. 9360–9369. [Online]. Available: https://aclanthology.org/2020.emnlp-main.752/

[4] X. Fu, J. Wang, and Z. Yang, "MM-AVS: A full-scale dataset for multi-modal summarization," in *Proc. NAACL-HLT*, Online, Jun. 2021, pp. 5922–5926. [Online]. Available: https://aclanthology.org/2021.naacl-main.473/

[5] J. Qiu, J. Zhu, W. Han, A. Kumar, K. Mittal, C. Jin, Z. Yang, L. Li, J. Wang, D. Zhao, B. Li, and L. Wang, "MMSum: A dataset for multimodal summarization and thumbnail generation of videos," in *Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition (CVPR)*, 2024. [Online]. Available: https://openaccess.thecvf.com/content/CVPR2024/papers/Qiu_MMSum_A_Dataset_for_Multimodal_Summarization_and_Thumbnail_Generation_of_CVPR_2024_paper.pdf

[6] F. A. Gigant, C. Guinaudeau, and F. Dufaux, "TIB: A dataset for abstractive summarization of long multimodal videoconference records," in *Proc. ACM Int. Conf. Multimedia (ACM MM)*, Ottawa, Canada, 2023. [Online]. Available: https://dl.acm.org/doi/10.1145/3617233.3617238

[7] J. Zhang, Y. Zhao, M. Saleh, and P. J. Liu, "SEAHORSE: A multilingual, multifaceted dataset for summarization evaluation," in *Proc. Workshop on Summarization Evaluation*, 2023. [Online]. Available: https://arxiv.org/abs/2305.13194

[8] X. Lin, Y. Zhou, Y. Li, and J. Sun, "MLASK: Multimodal article summarization kit," in *Findings of the Assoc. for Computational Linguistics: EACL*, Dubrovnik, Croatia, 2023, pp. 902–915. [Online]. Available: https://aclanthology.org/2023.findings-eacl.67/

[9] M. Li, X. Chen, S. Gao, and R. Yan, "Multimodal hierarchical methods for video-based news article summarization," *Information Processing & Management*, vol. 59, no. 6, pp. 102949, Nov. 2022. [Online]. Available: https://doi.org/10.1016/j.ipm.2022.102949

[10] S. Ghosh, R. Singh, and A. Choudhury, "FactMS: Enhancing multimodal factual consistency in multimodal summarization," *Applied Sciences*, vol. 15, no. 8, p. 4096, 2023. [Online]. Available: https://www.mdpi.com/2076-3417/15/8/4096