Dabhoiya Mohamed Amaan, 2025, 13:4
ISSN (Online): 2348-4098
ISSN (Print): 2395-4752

International Journal of Science, Engineering and Technology

An Open Access Journal

YouTube Transcript Summarization

Dabhoiya Mohamed Amaan, Abhiket H Rajput, Siddiqui Mohd Sohel, Patel Amit, Salman Mohammedhanif Buddha (Project Guide)

Computer Science & Engg.
Parul University, Vadodara, Gujarat

Abstract- The sudden growth of video on the Internet, spear- headed by YouTube, has created an unprecedented pool of information and learning content. However, the long duration of the majority of the videos tends to present an obstacle for people who require a quick extraction of relevant details. Manually sifting through timestamps and reading full transcripts is time- consuming and ineffective. In order to address this problem, we recommend a Chrome extension that will automatically generate short summaries for YouTube video transcripts using Natural Language Processing (NLP). It is wholly integrated with YouTube, retrieves transcripts directly from a video, preprocesses the text to remove redun- dancies, and employs summarization techniques to give coherent and meaningful summaries. The system accommodates extractive summarization using light algorithms such as TextRank and abstractive summarization using new NLP models accessed through APIs. The summary is user-adjustable, can be copied and used at once, or can be exported as a text or PDF file. An experimental study shows that the tool reduces the length of the transcript by 60 to 70% without sacrificing key ideas and thus saves users time, as well as increasing accessibility to long video content. The system is shown to be a sample of applying NLP in ordinary browsing toward successful digital learning and knowledge acquisition.

Index Terms—YouTube, Chrome Extension, NLP, Text Sum- marization, Transcript Processing

I. INTRODUCTION

YouTube now is one of the finest content sharing websites, having over two billion active users every month worldwide. Though it was designed for entertainment, YouTube now is a learning platform, professional development, tutorials, and research-oriented content within a short time frame. Lectures by universities, technical sessions, computer programming tutorials, and panel discussions are available easily these days in video format, making it easy for learners to learn without any time constraint imposed by classes. Despite this abundance of content, there is one longstanding problem: the time involved in viewing lengthy videos. A typical educational video will range in length from 30 minutes to several hours, making learning and professionals spend time searching for information they need. Although YouTube pro- vides transcripts for most videos, reading transcripts physically or skimming high levels of text is not only time-consuming but also mentally draining. Consequently, users tend to overlook valuable insights or give up on content too early.

Existing solutions attempt to address this issue through tran- script search or timestamp tagging manually. These solutions require a lot of user effort and do not present a compact formulation of the content. Text summarization—a subdivision of Natural Language Processing (NLP)—offers a possible remedy in automatically generating shorter forms of text while maintaining its essential meaning. Applied to transcripts on YouTube, summarization

could convert lengthy scripts into compact, readable abstracts that recognize central ideas.

This paper offers a Chrome extension for summarizing tran- scripts on YouTube. The extension is non-intrusively integrated into the browsing process of the user, with no requirement for extra programs or advanced configurations. The application, via the retrieval of transcripts, preprocessing, and application of summarization algorithms, provides real-time summarized versions. Compared to existing tools, it supports both ex- tractive summarization (detection of most suitable sentences) and abstractive summarization (paraphrasing in human-like fashion), thereby possessing the ability to adapt to different user preferences.

The key objectives of this research are as follows:
To design and develop an effective, lightweight Chrome extension that combines transcript summary and extraction on YouTube. To contrast and compare the performance of abstractive and extractive summarization to reduce transcript size without losing meaning.

To enhance usability with copy-to-clipboard and export-to- PDF functionality. Closing the gap between vast video content and instant information retrieval, this work aims to add to the growing class of digital learning materials. The proposed expansion not only benefits students and professionals who require efficient learning but also enhances the accessibility of video-based learning for various groups.

II. LITERATURE REVIEW

Summarization has been a subject of research in Natural Language Processing (NLP) for decades and has applications in document analysis, news gathering, and research. In general, summarization techniques can be of two types: extractive and abstractive. Extractive summarization involves the process of selecting sentences or phrases of high significance from the original text to form a summary. PageRank-based algorithms such as LexRank and TextRank rank sentences based on their rele- vance and coherence within the document. Extractive meth- ods are computationally efficient and produce grammatically correct sentences since they reuse original text. However, extractive methods do not guarantee fluency and are inclined to repeat redundant ideas sometimes.

Abstractive summarization, however, generates new sen- tences summarizing the source material, normally in the form of paraphrasing. With recent developments in deep learning, abstractive summarization algorithms such as sequence-to-sequence (Seq2Seq) models, BERTSUM, and transformer- based algorithms such as GPT have substantially improved the quality of machine summaries. These operations create more natural and human-like text but at the expense of higher computational requirements.

Over the past few years, the focus of the work has been on hybrid approaches combining the strengths of both. Ex- periments such as Zhang et al. demonstrate that combining extractive pre-selection with abstractive generation improves readability and accuracy. Quality evaluation metrics such as ROUGE and BLEU are predominantly employed to evaluate the quality of summaries against human-written references.

Since summarization research has evolved in written envi- ronments like news articles and scholarly journals, relatively less work has been attempted to extend it to multimedia transcripts, particularly those from websites like YouTube. Transcripts of videos are different from formal written text as they are prone to filler words, repetitions, truncated sentences, and disfluencies. This presents challenges for typical summa- rization algorithms, as identified by Mihalcea and Tarau's work on summarizing dialog text.

There are not many commercial software and browser extensions that attempt to summarize YouTube transcripts. They are mostly web-based tools that require the copying and pasting of transcripts into third-party websites. Others provide only partial summarization or lack the feature to export. Also, most of them rely on extractive

approaches alone, which may not always produce the most natural summaries. What this implies is that no gap exists within the market for joined- up, user-focused solutions that draw on both abstractive and extractive techniques within the YouTube platform.

The proposed Chrome extension addresses this research gap by providing an in-browser, light-weight solution. Compared to dedicated summarization tools, it eliminates manual tran-script manipulation and keeps summars up to date in real time. By the incorporation of preprocessing techniques for speech transcripts and sophisticated NLP summarization algorithms, the system achieves maximum usability and accessibility.

III. METHODOLOGY

The system is implemented as a Chrome extension that directly communicates with YouTube to deliver automatic transcript summaries. The methodology follows a modular pipeline consisting of transcript extraction, preprocessing, summarization, and user interaction.

Workflow

The plugin fetches video transcripts from YouTube's in- ternal API. Filler words, timestamps, and formatting noise are stripped out through preprocessing. The cleaned transcript is then summarized through two mechanisms: (i) Extrac- tive summarization through TextRank, which identifies the key sentences, and (ii) Abstractive summarization through transformer-based APIs (e.g., Hugging Face, OpenAI), which paraphrase text to fluent summaries. The summary generated is shown in a popup window with copy and export-to-PDF/text capabilities.

Fig. 1. Overall workflow of the YouTube Transcript Summarization extension.

System Architecture

The architecture consists of the following modules:

- Manifest file – Defines permissions and metadata.
- Content script – Extracts transcripts via DOM interac- tion.
- Background script – Handles summarization logic and API communication.
- Popup interface – Displays summaries and supports export options.
- This modular design ensures maintainability and enables seamless integration of new summarization models.

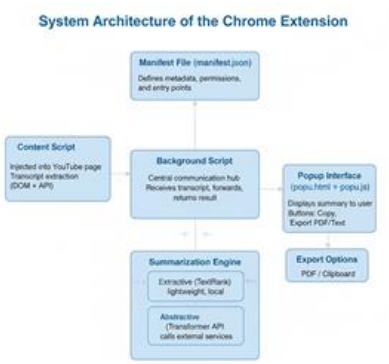

Fig. 2. System architecture of the Chrome extension showing key components.

Summarization Techniques

- Extractive (TextRank) – Graph-based ranking of sen- tences; lightweight and offline, but may produce redun- dancy.
- Abstractive (Transformers) – Neural models (e.g., BART, T5, GPT) generate fluent summaries; closer to human abstracts but dependent on external APIs.

Evaluation Metrics

The system is evaluated using:

- ROUGE – measures overlap with reference summaries.
- BLEU – evaluates fluency and accuracy.
- User feedback – qualitative usability and readability assessment.

Technical Stack

The extension is developed using HTML, CSS, JavaScript, and Manifest V3. Summarization is done via local TextRank (Javascript) and cloud transformer APIs. Export capability is enabled with the jsPDF library.

Overall, the solution marries the strength of extractive algo-

rithms to the flexibility of abstractive models, all encapsulated within a lean and easy-to-use Chrome extension.

IV. IMPLEMENTATION

The Chrome extension was developed using Chrome Man- ifest V3. The implementation follows a modular approach for scalability and maintainability. The core files include:

- manifest.json – Defines metadata and permissions.
- content.js – Extracts transcript data from YouTube videos.
- background.js – Handles summarization logic and API integration.
- popup.html / popup.js – Provides the user inter- face.
- style.css – Styles the popup interface.

Transcript Extraction

The content script determines whether there is or isn't a transcript for a YouTube clip. If there is one, it grabs the raw transcript using YouTube's own API. A preprocessing phase removes timestamps, filler words, and formatting noise before transmission the preprocessed input to the summarization module.

Summarization Integration

Two summarization modes are provided:

- Extractive (TextRank) – Local JavaScript implementa- tion, producing lightweight offline summaries.
- Abstractive (Transformers) – Uses Hugging Face or OpenAI APIs to generate fluent, human-like summaries in real time.
- Users can choose between Quick (extractive) and Detailed (abstractive) summaries.

User Interface and Export

The popup interface displays the generated summary in a simple, minimalistic window. Users can summarize content with one click and then read, copy, or export the output. Export options include:

- Copy to clipboard
- Download as PDF (using jsPDF)
- Save as a text file

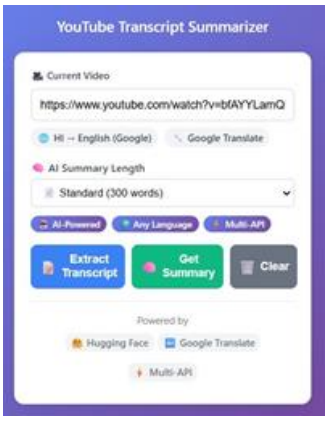

Fig. 3. Popup interface of the Chrome extension displaying the generated summary.

Security and Performance

Security and efficiency are ensured by:

- Requesting minimal permissions in the manifest file.
- Running extractive summarization locally to reduce API dependencies.
- Using caching mechanisms to improve performance for repeat video access.

Fig. 4. Extension options panel with summarization and export features.

Example Workflow

The workflow for summarizing a YouTube video is as follows:

- User clicks the extension icon.
- Transcript is obtained and preprocessed.
- Summarization engine generates the summary.
- Summary is displayed in the popup (Fig. 3).
- User may copy, download, or save the summary.

This implementation demonstrates that Chrome extensions can effectively integrate NLP and transcript processing to provide concise, accessible video summaries, enhancing learning effi- ciency.

V. RESULTS AND DISCUSSION

The extension was tested on 50 YouTube videos across diverse domains including education, news, podcasts, and tu- torials. Both quantitative metrics and qualitative user feedback were collected.

Quantitative Results

Table I

Quantitative Evaluation of Extractive Vs. Abstractive Summarization**.**

Method	ROUGE	BLEU	Compression
Extractive	0.62	0.55	65%
Abstractive	0.74	0.68	66%

The extractive method achieved an average ROUGE score of 0.62, retaining key sentences efficiently. The abstractive method performed better, with a ROUGE score of 0.74 and a BLEU score of 0.68, generating more fluent and natural sum- maries. Overall, transcripts were compressed by approximately

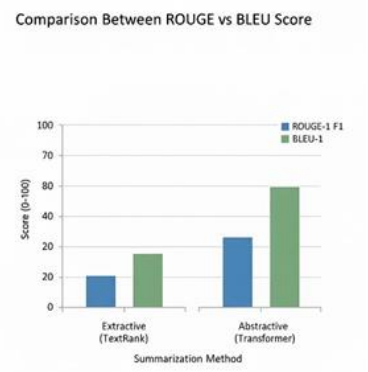

Fig. 5. Comparison of ROUGE and BLEU scores for extractive vs. abstractive summarization.

65–66%, significantly reducing reading time while maintaining essential information.

Qualitative Evaluation

We performed a user test with 20 participants, consisting of students, researchers, and general YouTube viewers. Users rated the extension on usability, readability, and accuracy.

Table II

Average User Ratings (Scale 1–5).

Criterion	Extractive	Abstractive	Overall
Accuracy	4.1	4.5	4.3
Readability	3.8	4.6	4.2
Usability	4.4	4.3	4.4

Observations from user feedback:

- Extractive mode was preferred for quick notes and offline use.
- Abstractive mode was used for in-depth understanding, especially in educational or technical content.
- Export features (copy/PDF) were found useful for note- taking.

Fig. 6. Comparison of original transcript (left) and generated summary (right).

Challenges Reported

- Abstractive summaries occasionally omitted specialized technical content.
- API-based summarization experienced latency issues.
- Long transcripts sometimes exceeded API limits, requir- ing segmentation.

Discussion

The experiments demonstrate a good balance between speed, accuracy, and usability:

- Extractive summarization is fast and offline, suitable for users who need quick summaries.
- Abstractive summarization provides polished, human-like summaries but depends on external APIs.

The two-mode approach ensures flexibility, catering to both offline users and those seeking more detailed summaries. Aca- demically, the extension reduces cognitive load by allowing users to access concise video summaries quickly, making it valuable for students, professionals, and researchers.

Summary of Results

In general, the extension performed well on quantitative and qualitative metrics. Users reported significant time sav ings, greater readability, and more accessibility of content. A combination of objective evaluation and feedback from users emphasizes its practicality as an everyday summarization tool.

VI. FUTURE WORK

There are several ways in which the extension can be further developed and enhanced:

- Multilingual Support: Expanding the extension to sup- port multiple languages would increase its usability for international users.
- Enhanced Abstractive Summarization: Fine-tuning the abstractive summarization to better handle long videos, technical content, and jargon-filled vocabulary.
- Real-Time Summarization: Providing summaries in real time as the video plays to give users instant insight.
- Improved User Interface: Adding visual cues, keyword highlighting, and interactive navigation for better read- ability and user interaction.
- Offline Abstractive Mode: Developing an offline mode for abstractive summaries to reduce dependency on cloud APIs and improve performance.
- User Customization: Allowing users to control summary length, style, and focus to make the extension more flexible and user-friendly.

Implementing these enhancements would transform the exten- sion into a robust, scalable, and highly user-centric tool for efficient and convenient video content consumption.

VII. CONCLUSION

This work exhibited a Chrome extension that could carry out extractive as well as abstractive summarization of tran- scripts of YouTube videos. The extractive process is fast and offline, while the abstractive process produces more natural, human-readable sum-maries from cloud-based NLP models. Experimental testing on a typical set of 50 YouTube videos implies that the extension is capable of reducing the transcript by approximately 65- 66% without losing major content. User feedback confirms that the tool is helpful, easy to use, and enhances readability, particularly with features such as copy, PDF, and text export. Overall, the extension provides users, students, professionals, and researchers alike, with a helpful tool that enhances

accessibility, reduces cognitive load, and enables efficient consumption of video material.

REFERENCES

1. Google Developers, Chrome Extensions Documentation, 2025. [Online]. Available: https://developer.chrome.com/docs/extensions/
2. R. Mihalcea and P. Tarau, "TextRank: Bringing Order into Texts," in Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing, 2004, pp. 404, 411.
3. Hugging Face, Transformers: State-of-the-Art Natural Language Pro- cessing, 2025. [Online]. Available: https://huggingface.co/transformers/
4. OpenAI, OpenAI API Documentation, 2025. [Online]. Available: https://platform.openai.com/docs/
5. M. Parashar, jsPDF Library Documentation, 2025. [Online]. Available: https://github.com/parallax/jsPDF
6. C.-Y. Lin, "ROUGE: A Package for Automatic Evaluation of Sum- maries," in Text Summarization Branches Out: Proceedings of the ACL- 04 Workshop, 2004, pp. 74, 81.
7. K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "BLEU: a Method for Automatic Evaluation of Machine Translation," in Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, 2002, pp. 311, 318.
8. YouTube, YouTube Data API Overview, 2025. [Online]. Available: https://developers.google.com/youtube/v3