Video Transcription

Mr. Bhaskar das*, B.Aditi[1], T.Anish[2], B.Maneesha[3], CH. Sowmya[4]
*Assistant Professor, Department of Csd, Hyderabad Institute of Technology and management, JNTUH, Hyderabad, India
[1,2,3,4] Student, Department Csm and cso, Hyderabad Institute of Technology and management, JNTUH, Hyderabad, India

Abstract: **This project presents a web application for automated keyword extraction and analysis from video and audio content, including YouTube videos. Leveraging the Whisper speech recognition model and advanced natural language processing techniques, the application aims to provide efficient and accurate keyword extraction capabilities for various applications.**

The system processes uploaded video files or YouTube URLs, extracting audio content and transcribing it using Whisper. Subsequently, TF-IDF or CountVectorizer algorithms are employed to identify significant keywords from the transcribed text. For YouTube videos, keyword accuracy is evaluated against available transcripts, while for uploaded files, user-provided expected keywords can be used for comparison. Performance metrics, including precision, recall, F1-score, and coverage score, are calculated to assess keyword extraction effectiveness.

The developed web application, built using the Flask framework, offers a user-friendly interface for seamless interaction and facilitates access to extracted keywords, full transcripts, and performance metrics. This tool holds potential applications in content analysis, information retrieval, video indexing, and automated metadata generation. The project demonstrates the feasibility of combining speech recognition and natural language processing for robust keyword extraction from multimedia content, offering valuable insights and efficiency improvements for content management and analysis workflows.

I. INTRODUCTION

This project addresses the challenge of automatic keyword generation and evaluation for video and audio content. The process of manually assigned keywords is very time- consuming and subjective, thus limiting multimedia data discovery and analysis. This project presents a system that uses advanced speech recognition and natural language processing techniques to automate the process. The input into the system can either be an uploaded video file or YouTube URLs. It relies on the Whisper model for audio transcription with high accuracy, then uses TF-IDF and Count Vectorization in the process for keyword extraction. The system compares the generated keywords against the ground truth, which can either be the official YouTube transcript or the one extracted using Whisper. Accuracy metrics like precision, recall, and Fl-score are computed to judge the performance of the system. The results prove the effectiveness of the proposed methodology for generating words which are relevant and accurate. The system shows some promising accuracy scores when using the transcript of YouTube as ground truth for testing purposes. Keywords thus derived provide useful information about the content of video or audio and help in content indexing, search, and analysis. This project helps in the field of multimedia information retrieval in terms of keyword generation and evaluation by providing an automatic and reliable solution. It can thus find applications in any of these various domains: content management, education, or even basic research. The modular design allows for potential future extensions and improvements-the integration of other keyword extraction methods or, as might become relevant, an introduction of some more important metrics for evaluation.

II. LITERATURE SURVEY

Study	Author	Benefits	Limitations	Year
1)Automatic Thumbnail Generation for Videos	Baoquan Zhao	Automates the creation of informative video thumbnails by leveraging salient	May not account for subjective aspects of thumbnail aesthetics or context-specific relevance.	2010

		visual and textual metadata.		
2) Key Term Extraction from Spoken Course Lectures	Yun-Nung (Vivian) Chen, Yu Huang, Sheng-yiKong, Lin-Shan Lee	Enhances understanding of lecture content through unsupervised and supervised term extraction techniques.	Limited to English lectures; performance dependent on the quality of transcription.	2010
3) Applications of Automated Video-to-Text (V2T) for Lip Reading	Ong, Lan, Theobald, Harvey, Bowden	Provides scalable lip-reading solutions for police applications, addressing human limitations.	Automation accuracy may vary under challenging conditions like noise or poor video quality.	2006
4) Lecture2Note: Automatic Generation of Lecture Notes from Slide-Based Educational Videos	Chengpei Xu, Ruomei Wang, ShujinLin, Xia onanLuo	Improves learning efficiency by generating structured notes using visual and speech text integration.	Effectiveness depends on the clarity and structure of the original slides and speech.	2019
5) Text-to-Picture (TTP) and Picture-to-Text (PTT) Synthesis	Yong Xuan Tan, Chin-Poo Lee, Mai Neo, Kian Ming Lim	Introduces innovative methods for generating Arab sentences and improving communication skills.	Limited to specific use cases; existing approaches may struggle with contextual nuances.	2023
6) Automatic Notes Generation from Lecture Videos	D. R. Pratheeksha, R. P. Shreya Reddy, R. Jayashree	Reduces the burden of manual note-takingusing NLP and video summarization technologies.	May not fully capture nuances or non-verbal cues from lectures.	2022
7) Improving Video-Text Retrieval by Multi-Stream Corpus Alignment and Dual SoftmaxLoss	Xing Cheng et al.	Achieves state-of-the-art accuracy in video-text retrieval by aligning video and text features effectively.	Performance relies on training data quality; scalability can be resource-intensive.	2022

III. PROPOSED METHOD

The proposed method introduces an efficient system capable of automatically generating structured notes from video content. It first accepts video input from local files and YouTube URLs, allowing the user to source content freely. For YouTube URLs, the method downloads the video using tools such as pytube or youtube-dl, whereas for a local video, processing can be done directly. To extract the audio from the video, it uses FFmpeg in a manner that will be compatible with all forms of formats.

The text is then obtained by utilizing Google's Speech Recognition API after audio is extracted. This API is very fast and highly accurate, thus suitable for the processing of large or complex video files. The transcription forms the raw textual content that is full of unnecessary words and phrases. Filtering out common words and extracting significant keywords is therefore applied by the system through scikit-learn's CountVectorizer. This step involves keeping only the most significant information and concentrating on the core themes of the video. The transcription is formatted in a way that enhances its readability. Using an external API for text processing, the text is linearized into coherent paragraphs or sections identifiable by alphabetical markers or other logical segmentation. Thus, the text content becomes more accessible, so the user can easily apprehend the basic ideas without needing to wade through a block of unstructured text.

Performance evaluation of keyword extraction using precision, recall, and F1-score metrics is performed by the system. To ensure the relevance and accuracy of the extracted information, it compares the output with a manually curated benchmark. Processing time for each step-audio extraction, transcription, and keyword filtering-is also determined to make sure the system works efficient even while dealing with large-scale inputs.

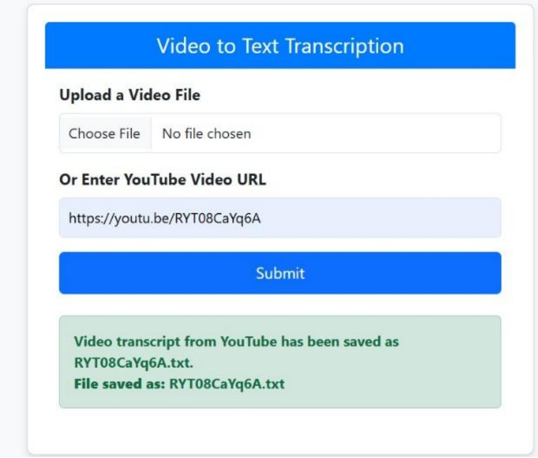
The final output would come in the form of two formats: a human-readable summary that can be immediately consumed, and a structured JSON file for easy integration with other applications. This would allow for the support of the widest range of use cases, from personal note-taking to large-scale data analysis. By integrating advanced tools and methodologies, the proposed method provides a robust and scalable solution for transforming video content into actionable insights.

IV. METHODOLOGY

This project employs a systematic approach to process video and audio files, extracting relevant information using modern tools and techniques. The methodology is broken down into several key steps. First, the system accepts input in the form of a local video file (supported formats include .mp4, .avi, .mov, and .mkv) or a valid YouTube URL. File uploads and URL submissions are handled through a Flask route (/upload), with input validation modules ensuring only allowed file types or valid YouTube URLs are processed. Once the input is received, audio extraction takes place. For uploaded videos, FFmpeg is used to extract and encode the audio as a .wav file, which is then stored in a designated directory. For YouTube videos, yt_dlp retrieves the best available audio quality, which is subsequently converted into .wav format using a postprocessor.

The next step involves transcribing the extracted audio into text. This is achieved using the Whisper AI model, a robust transcription tool capable of handling diverse audio formats. The transcribed text is saved as a .txt file in a specified folder for further evaluation or retrieval. Finally, textual analysis is performed to extract meaningful keywords from the transcription, ensuring that critical information is highlighted for subsequent use. This methodology ensures a streamlined and efficient process for handling and analyzing multimedia inputs.

V. IMPLEMENTATION



Video to Text Transcription

Upload a Video File

Choose File No file chosen

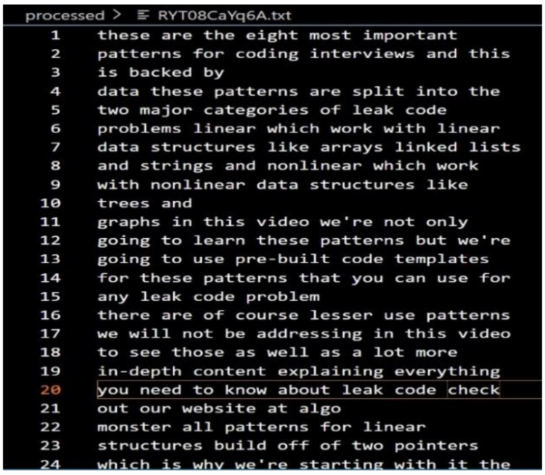
Or Enter YouTube Video URL

https://youtu.be/RYT08CaYq6A

Video transcript from YouTube has been saved as RYT08CaYq6A.txt.
File saved as: RYT08CaYq6A.txt

Fig 1: Web Ui

VI. RESULT



```
these are the eight most important
patterns for coding interviews and this
is backed by
data these patterns are split into the
two major categories of leak code
problems linear which work with linear
data structures like arrays linked lists
and strings and nonlinear which work
with nonlinear data structures like
trees and
graphs in this video we're not only
going to learn these patterns but we're
going to use pre-built code templates
for these patterns that you can use for
any leak code problem
there are of course lesser use patterns
we will not be addressing in this video
to see those as well as a lot more
in-depth content explaining everything
you need to know about leak code check
out our website at algo
monster all patterns for linear
structures build off of two pointers
which is why we're starting with it the
```

Fig 2: Full Text

VII. CONCLUSION

The recent project showcased the utilization of new technologies and tools for processing and analyzing multimedia content. It is equipped with Python, Flask, and other libraries such as whisper, yt_dlp, and sklearn which help the user perform tasks in a streamlined manner which includes downloading, extracting, and transcribing video and audio data. By adding such functionalities as retrieval of YouTube transcripts and supporting various media types, the system is guaranteed to be applicable in a wide range of circumstances.

The proposed system can also be the starting point for applications in higher education, media production and accessibility. It facilitates the creation of practical products that allow the user to obtain target information from a variety of multimedia resources. This project illustrates the type of advances that can be made with new ideas to real problems and can be expected to have further development and applications in the future.

VIII.REFERENCES

[1] B. Zhao is credited in 2010. Many ideas from the work are recognized. Videos generate automatic thumbnails.

[2] Chen, Huang, Kong and Lee contributed to the study. Key term extraction from spoken course lectures was accomplished in 2010, while multiple methods were tested to improve the clarity and relevance of the identified terms.

[3] Ong L and Theobald H helped Harvey B and Bowden R in 2006. Many researchers contributed to this work. Automated Video-to-Text (V2T) technology helps with lip reading. This technology finds applications in multiple fields.

[4] Xu, C. and Wang, R. conducted a study. Lin, S. and Luo, X. co-authored it in 2019. Automatically generate lecture notes from videos that use slides.

[5] Tan, Y. X., Lee, C.-P., Neo, M. and Lim, K. M. authored a work. Text-to-Picture (TTP) synthesis creates images from written descriptions, while Picture-to-Text (PTT) synthesis generates detailed text based on the analyzed content of images.

[6] Pratheeksha, D. R., along with Reddy, R. P. S. and Jayashree, R., conducted a study that contributed costly understandings to their field. Lecture videos create automatic notes.

[7] Cheng, X. and colleagues conducted a study. We improve the retrieval of information from videos and text combinations by using advanced techniques. Our goal is to make it easier for users to find relevant content efficiently.