RSET
RAJAGIRI SCHOOL OF
ENGINEERING & TECHNOLOGY
(AUTONOMOUS)

Project Report On

Video Transcriber

Submitted in partial fulfillment of the requirements for the award of the degree of

Bachelor of Technology

in

Computer Science and Engineering

By

Daryl Antony Luiz (U2103074)
Eshaan Kolloth (U2103086)
Melissa Biju Kalayil (U2103136)
Milin Chandrakumar Alamanda (U2103138)

Under the guidance of

Mrs. Mehbooba P Shareef

Department Of Computer Science and Engineering
Rajagiri School of Engineering & Technology (Autonomous)
(Parent University: APJ Abdul Kalam Technological University)
Rajagiri Valley, Kakkanad, Kochi, 682039
November 2024

CERTIFICATE

This is to certify that the project report titled ***"Video Transcriber"*** *is a bonafide record of the work done by* ***Daryl Antony Luiz (U2103074)***, ***Eshaan Kolloth (U2103086)***, ***Melissa Biju Kalayil (U2103136)*** *and* ***Milin Chandrakumar Alamanda (U2103138)*** *, submitted to Rajagiri School of Engineering & Technology (RSET) (Autonomous) in partial fulfillment of the requirements for the award of the degree of Bachelor of Technology*
(B. Tech.) in Computer Science and Engineering during the academic year 2024-2025.

Mrs. Mehbooba P. Shareef
Project Guide
Assistant professor
Dept. of CSE
RSET

Mrs. Anu Maria Joykutty
Project Co-ordinator
Assistant professor
Dept. of CSE
RSET

Dr. Preetha K. G
Professor and HoD
Dept. of Computer Science
RSET

ACKNOWLEDGMENT

We wish to express our sincere gratitude towards **Rev. Dr. Jaison Paul Mulerikkal CMI**, Principal of RSET, and **Dr. Preetha K. G**, Head of the Department of Computer Science for providing us with the opportunity to undertake our project, "Video Transcriber".

We are highly indebted to our project coordinators, **Ms. Anu Maria Joykutty** , Associate Professor, Department of Computer Science and Engineering and **Dr. Jisha G**,Associate Professor, Department of Computer Science and Engineering , for their valuable support.

It is indeed our pleasure and a moment of satisfaction for us to express our sincere gratitude to our project guide **Mrs. Mehbooba P Shareef** for her patience and all the priceless advice and wisdom she has shared with us. We also express our sincere thanks to our co-guide, **Dr. Sminu Izudheen** for her support.

Last but not the least, we would like to express our sincere gratitude towards all other teachers and friends for their continuous support and constructive ideas.

Daryl Antony Luiz

Eshaan Kolloth

Melissa Biju Kalayil

Milin Chandrakumar Alamanda

Abstract

This project basically involves a software application that transcribes text from YouTube videos and generates summaries from the produced texts and transforms these summaries into interactive quiz applications to accelerate the learning and retention process. The software will enable transcription from multiple languages and include a lot of other advanced features like speaker identification, keyword highlighting, in real-time transcript and summary, time-coded summary, and topic segmentation.

It will also be a place for accessibility tools like text-to-speech, summarization control, and editing transcriptions and tools for the blind and hearing-impaired, such as lip-reading assistance, audio-visual enhancement, noise reduction, live captioning, and display sign language.

There will be an application of speech-to-text for real-time and multilingual transcription, where the summarization process would be very concise with time coding of information on active keywords in addition to effectively presenting the different topics through the use of NLP and ML techniques. Thus quizzes based on mcq questions shall be formulated with the summarized text by the AI-based quiz application, which will propel the active learning process.

Web application will be developed using the frameworks Django or Flask along with React that will make it user-friendly and seamless. Enhanced performance and accuracy will be given to every process through AI and ML making it accessible and beneficial for all users.

Contents

List of Abbreviations

NLP - Natural Language Processing

ML - Machine Learning

AI - Artificial Intelligence

ASR - Automatic Speech Recognition

RNN - Recurrent Neural Network

LSTM - Long Short-Term Memory

MCQ - Multiple Choice Questions

CTC - Connectionist Temporal Classification

CNN - Convolutional Neural Network

NMT - Neural Machine Translation

MG - Motion Graph

GAN - Generative Adversarial Network

HLR-Net - High-Level Reasoning Network

GRU - Gated Recurrent Unit

CER - Character Error Rate

WER - Word Error Rate

GRID - General-purpose, Research, and Industry Data

TF-IDF - Term Frequency-Inverse Document Frequency

LSA - Latent Semantic Analysis

LDA - Latent Dirichlet Allocation

EDUVSUM - Educational Video Summarization

ROUGE - Recall-Oriented Understudy for Gisting Evaluation

List of Figures

List of Tables

Chapter 1

Introduction

Itoday's fast-paced digital world, especially on sites such as YouTube, video content is the gold standard of communication. Video content is not accessible for people who are blind, deaf, or simply do not know the language. We are working on a revolutionary unique software application which transcribes YouTube video into text to access it, summarizes the content then builds an interactive quiz application to go along with it, hopefully creating more learning, accessibility, and engagement through rich, interactive, experiences for all users no matter the ability. Our software will have a multilingual transcription engine, real-time transcription, speaker identification and keyword highlighting. Additionally we will have time coded summaries and topic segmentation for video content navigation and digestibility. We are also looking to include some key accessibility features for the blind and deaf community and they are planning to do this by implementing text to speech, live captioning, display in American Sign Language, and lip reading assistance. We will use modern web technologies (e.g., Django or Flask with React) to build the application and use artificial and machine learning for accurate and efficient processing.

1.1 Background

The burgeoning spectrum in the availability of online video content has rendered YouTube indispensable; it has become the platform through which much learning and entertainment are increasingly disseminated. Video has established itself as a principal medium through which educational content, tutorials, news updates, and podcasts are distributed-it has spread itself across a vast number of nodes reaching billions of audiences. However, while information is available more widely than available to date, it still does not quite meet the criteria for flexibility and adaptability that might define it as efficient learning and retention material for different-the complicated audiences it serves.

Existing transcription and summarization tools for video content are largely limited in their coverage of utility and availability. Basic transcription tools may have an option for closed captioning, but they lack much feature besides fairly useful ones like live transcription, speaker identification, or multilingual support. Such limitations constrain the ability of video content in terms of usefulness to users who could gain from such features, especially in educational and professional environments where understanding and retention of important information are paramount. Current approaches do not tend to provide interactive elements-such as quizzes or structured segmentation of topics-that would allow more active involvement and ultimately strengthen learning outcomes.

Access further suffers according to the general definitions. Most of the time, automatic captioning provides by websites such as YouTube is most inaccurate in multilingual contexts and also when background noises obscure things or when otherwise disrupted conditions repeat. Also, basic captions do not cater much to the overall accessibility needs of blind and hearing-impaired users. Features like text-to-speech and sign language interpretation are required.

1.2 Problem Definition

The main issue addressed by this project is the challenge individuals face with lengthy educational videos, which leads to wasted time and reduced concentration. Learners often struggle to maintain focus, resulting in inefficient learning. This problem is especially acute for people with disabilities, as they face additional barriers in accessing and engaging with multimedia contents such as videos.

1.3 Scope and Motivation

Applications that transcribe, summarize, and transform all YouTube video content in interactive mode are now being developed. Such an application would turn out to have real-time transcriptions in multiple languages, speaker identification, as well as highlight keywords to improve the user access and user engagement. The user navigates the contents quite easily, thanks to the time-coded summaries and topic segmentation features. In addition, the project will also feature accessibility features such as text-to-speech, live captioning, and display of sign language essential for people with visual and hearing

impairments. The software is developed with Django or Flask with React front end using AI and machine learning techniques for quality processing and smooth user experiences. Project motives borne of high increase in demand for accessible digital content even at times, current loses of ability of transcription and summarization tools in addressing it.

As online educational and informative materials increase in volume, the need for applications to convert such content into an extension that everyone can use has become more urgent. Most of the current platforms totally exclude features that are supposed to make them suitable for use by impaired users who have to rely on many different forms of adaptive technologies for meaningful interactions with the digital content. The proposed Advanced Transcription, Summarization, and Accessibility Tool is intended to bridge this gap and thus ensure a broader audience can access information and enhance the learning experiences of diverse audiences. Access standards have been met.

1.4 Objectives

- Create a state-of-the-art application software that transcribes Youtube videos live and in different languages.
- It is also going to have speaker identification and keyword highlighting features for enhanced clarity and use of transcriptions.
- A summarization feature with time-stamped summaries and topic segmentation shall also be included to improve content navigation.
- An Interactive Quiz feature could be nice to prep our user for really learning with the video content subject.
- Feature support should be added for text-to-speech, real-time captioning, transcription editing, display sign language to entice and feast users who would use those with visual and auditory disabilities.
- Dwell on making it more human user-friendly with AI and machine learning capabilities for more accurate processing which is very fast during design excursion.

1.5 Challenges

It establishes a challenge where to achieve real-time transcription accuracy, it should be put into numerous languages and different audio records which are from background noise or obtrusive speakers. More so, to create a high-level accessibility feature such as sign language display and assisted lip-reading requires complex technologies with the capabilities of seamless user experience across diverse environments and devices.

1.6 Assumptions

It will be those YouTube recordings that are going to be used for transcription whose audio quality is clear and video quality is good or fair. Such audience will have devices that are made with the ability to support real-time transcription and interactive application features. This application would generally be used by users requiring multilingual support and/or having disability access capabilities, either visual or hearing impaired.

1.7 Societal / Industrial Relevance

This initiative has considerable social relevance, especially in making educational and informational video-based content available to the disability population with the consideration for those who are blind and deaf. It is also applicable to the education, e-learning, media, and entertainment industries where accessible, interactive content can engage the wider audience. Overall, the project aims to facilitate interaction with video content as part of an inclusive approach, in response to the increasing digital content trend towards universal accessibility, for greater inclusivity in education and work contexts.

1.8 Organization of the Report

The report is organized as follows:

- Chapter 1: Introduction - Provides an overview of the project, including the background, problem definition, objectives, challenges, assumptions, societal relevance, and the report's structure.

- Chapter 2: Literature Review - Discusses existing technologies, research, and tools related to transcription, summarization, accessibility features, and AI applications.
- Chapter 3: System Design and Architecture - Outlines the design principles, system architecture, and technologies used in the development of the application.
- Chapter 4: Implementation - Details the development process, including coding, algorithm implementation, and integration of features.
- Chapter 5: Results and Evaluation - Presents the evaluation of the system's performance and effectiveness, including user feedback and testing results.
- Chapter 6: Conclusion and Future Work - Summarizes the project outcomes and suggests areas for future improvement or expansion..

1.9 Conclusion

This project makes an effort to change how online video content is used so that all individuals can enjoy what the wealth of information through platforms like YouTube, no matter the difference in ability or language: Melding current technologies such as AI and ML with real-time transcription, all under the argument for enhanced access, ensures the that the product innovatively works. Along with such time-coded summaries, multilingual support, and the interactive quiz, the learning experience will be enhanced because individuals with disabilities will be empowered through text-to-speech, sign language display, and live captioning.

Through such software, we hope to be all possible by making digital content attractive and educative across the world's audience, breaking barriers for an inclusive future.

Chapter 2

Literature Survey

This chapter highlights progressions and functionalities in video transcription, which is being applied with increased frequency across all realms including: education, media, and accessibility. Video transcription systems have developed to contain separate functionalities addressing very different problems that accompany the conversion of video content into text and improve serviceability. The central function is automatic speech recognition, which means it takes speech-as-model approaches like recurrent neural networks, long-short-term memory, and transformer approaches, all of which provide strong accuracy across such complicated audio circumstances.Similarly, modules like transcript summarization improve readability by taking long transcriptions and condensing the text; text-to-sign language conversion helps those who are deaf and hard-of-hearing by making the content accessible with sign language. More robust transcription gets more organized with the capacity to separate thematic areas for an automatic multiple-choice question (MCQ) generation in education applications. Lip-reading is one such technology that will generate coherent or busy sounds derived from meaningful sounds.These technologies enhance the potential applicability of video transcription platforms across areas.

2.1 Speech Recognition via CTC-CNN Model [1]

The CTC-CNN model explored in this article is a CTC based model with a CNN attached for improved speech recognition. The acquisition of CTC serves a valuable purpose inthis end-to-end framework process, as the model is evaluated on how well it performs differentiation forinputs and outputs of different lengths: a natural obstacle of speech recognition. CTC alliesthe input speech sequences with the text sequences, regardless of the data being aligned frameby frame. The likelihood of sequence optimization presents a source of energy for the training process.

The CTC extracts high-level acoustic features of speech for sequence classification, thereby transforming audio input into a string of text. The architecture of the model consists of several convolutional layers for feature extraction, a few fully connected layers and finally a Softmax layer to output the probability distribution of the label sequence. During training, the CTC framework enables a forward-backward algorithm to compute loss at each time step which allows the model to learn the temporal dependencies of the speech data. The CTC-CNN model was trained on several speech databases with considerable gains observed in word error rate using batch normalization, residual connections and dropout for model robustness and to combat overfitting.

This structure allows the model to work even in highly noisy environments: suitable for real speech environments applications.

2.2 Sign Language Production Using Neural Machine Translation and Generative Adversarial Networks [2]

To change spoken language sentences into glosses of a sign language using some of the most advanced research methods in recurrent neural networks based based machine translation, specifically neural machine translation based on attention, we made the decision to implement an encoder-decoder framework. Assuming the spoken language has a sentence of multiple words, the encoder is used to map the sequence into hidden representation or latent representation of the encoded sentence. This hidden representation is passed to a decoder with a attention mechanism which will model long-term dependencies in the encoding and create a probability distribution on the glosses. The decoder will also represent a motion graph (MG) is constructed to produce a sequence of 2D skeletal poses for a gloss sequence. An MG is a Markov Process and can be used to model sequences of motion since they represent actual motion, yet support of animator's intent. Once we synthesize subject-specific 2D skeletal poses, we will create sign language video using Generative Adversarial Networks (GAN). GANs model two models - generator (G) creates samples of new data, while discriminator (D) determines whether the new data is from the same distribution of the training data.

2.3 HLR-net: a hybrid lip-reading model based on deep convolutional neural networks [3]

This article discusses a model for lip-reading called HLR-Net that aims to enhance automatic lip-reading with visual lip motion that can then be converted into a text output. The model was designed for hearing-impaired individuals and consists of three stages: preprocessing, encoding, and decoding. The model utilizes a preprocessing component that extracts and normalizes video frames, which scans specifically the mouth region in order to capture lip motion. The encoder uses layers with inception and preservation of gradients packed with additional layers that utilize a bidirectional GRU to pull from the frames that encode meaningful features. Finally, using an attention layer and CTC, the decoder processes the aforementioned features into human-readable text.HLR-Net was evaluated against a sentence-level data set of lip-reading samples: the GRID corpus. Overall, and in direct comparisons with existing models such as LipNet, HLR-Net yielded results with substantial improvements in both CER and WER for tests with both unseen and overlapped speakers indicating its ability to robustly recognize the lip motion of different speakers. HLR-Net has a desire for field application and potential for consideration in better aids for communication for individuals who are hearing impaired.

2.4 Educational Videos Subtitles' Summarization Using Latent Dirichlet Allocation and Length Enhancement[4]

The work entitled Use of Latent Dirichlet Allocation and Length Enhancement in Academic Videos Subtitle Summarization tackles the necessity of summarizing the subtitles of educational videos: crawling the key content for the student in order that he retains major elements of such learning. Most of the time, traditional approaches for summarizing textual material generally rely on TF-IDF or LSA. However, this study applies LDA techniques of summarization based on the ability of the techniques in locating the significant topics. In the process, preprocessing of the subtitle text, next training an LDA model for generating topic keywords, and finally extracting sentences that are containing those keywords for summarization would be carried on. A length enhancement method is also being introduced to tackle the issue of producing very long summaries by LDA as a result of discarding words that are not nouns. Evaluation carried out on EDUVSUM

dataset alongside ROUGE scoring show how LDA base summaries surpassed those by TF-IDF and LSA in terms of precision and recall.

The research developed human summaries to validate the EDUVSUM dataset, as annotations were not provided in previous datasets. The model based on LDA produced longer summaries compared to the TF-IDF and LSA models, and subsequent length enhancement increased precision without altering recall. Both human evaluation and ROUGE scores captured core ideas from the LDA model and, thus, it is suitable for educational purposes, but it is adaptable to other contexts, such as news video summarization. Possible future work could include enhancing LDA generated summaries in punctuation completion, as well as refining filtering with contextually specific keywords.

2.5 Summary and Gaps Identified

2.5.1 Summary

The second chapter examined recent developments in video transcription systems, which can be used for educational purposes and as a media and accessibility tool. The video transcription systems module identifies various issues with taking a video input and converting it into a text output. Transcription systems are based on automatic speech recognition (ASR) employing the following models: recurrent neural networks (RNNs) and long short-term memory networks modified with transformers to increase accuracy when transcribing video input with difficult-to-hear audio conditions. Other identified modules include transcript summarization and text to sign language translations intended to communicate with Deaf and Hard of Hearing communities. Additional modules include representations for topic segmentation that organize transcript a topic theme, and lipreading models that use visual cues to transcribe the video input in noisy contexts. The various module capabilities and function allow transcription systems to be useful in a range of contexts and differ user needs.

This chapter also includes several specific models and methodologies such as the CTC-CNN for ASR—Connectionist Temporal Classification—is such a model, in which Convolutional Neural Networks (CNNs) are involved with varying sequences of input and output. One major advancement is in the area of text-to-sign, which is using attention-based neural machine translation methods, Motion Graphs, and Generative Adversarial

Networks (GANs) to convert skeletal poses to videos of sign languages. The lipreading model of HLR-Net improves the accuracy of transliterating visually observed lip movements into text, providing possible communication supports for persons with hearing loss. Finally, educational video caption summarization is being assisted with Latent Dirichlet Allocation (LDA) to identify key topics and a length-extending strategy to obtain longer summaries while still being concise in length.

2.5.2 Gaps Identified

Weak Audio Conditions Still Leave a Mark on Current Models-Transcription accuracy is limited by realistic, noisy places. Accent and Dialect Variability: The different accents and dialects will be addressed in the future since they will still "affect" the accuracy of ASR over different user demographics.

Multilingual Constraints: There was robust progress in the area of multilingual support, but in reality, many transcription models had not reached any of the big languages. Sometimes, these models that exist today don't even handle emotion or sentiment in speech would be useful to develop in the context of customer service or therapeutic type settings. Bad Compatibility with other Waste Transition Videos would have been much a better development using other systems as would lead to better workflow and functionality while still enhancing succession planning and preparing learners.

2.5.3 Conclusion

Chapter 2 highlights the incredible features of video transcription, as well as those developing methods that will enable further access, use, and quality in diverse applications. By now, however, most of the systems still have drawbacks like handling audio variability, quality, differentiation of accents, language support, and integration with other technologies. Some of the wider holes that should be filled in future research on real-world applications include building up models to become more resilient and adaptive in different environmental and linguistic conditions.

Chapter 3

System Design

The project about the transcription of video content to an accurate text transcription, known as Video Transcriber. This chapter is all about the system design of the project that includes its architecture, components, algorithms, and data flow.

Such a system is developed to carry out a single task such as video files processing, extraction of available audio from the file, and other specific manipulation exercised by advanced algorithms that will either be machine learning or an advanced natural language processing (NLP) to achieve very accurate precision transcriptions. But it should also help smoothen that architecture's integration into any front-end user interface, back-end modules, and third-party APIs used for transcription or audio analysis.

Hence this would also include the proposes tool and techniques such a project along with dividing a work-up into much smaller handles modules and estimating time for the project in order to ensure on-time delivery.

So this chapter covers the ground to make sure the prototype working or functional system will be implemented successfully and logically. It defines the data flow defined in the planning as well as the deliverables. It should also smooth out the integration of this architecture into any front-end user interface, back-end modules, or third-party APIs employed for transcription and/or audio analysis.

Thus, this chapter includes the proposes tool and techniques for such a project in addition to splitting a work into modules small enough to be handled and an estimated time for the project to ensure timely delivery.

This chapter thus lays the groundwork to make sure the successful and logical implementation of the prototype working or functional system. It defines the data flow defined in the planning as well as the deliverables.

3.1 System Architecture

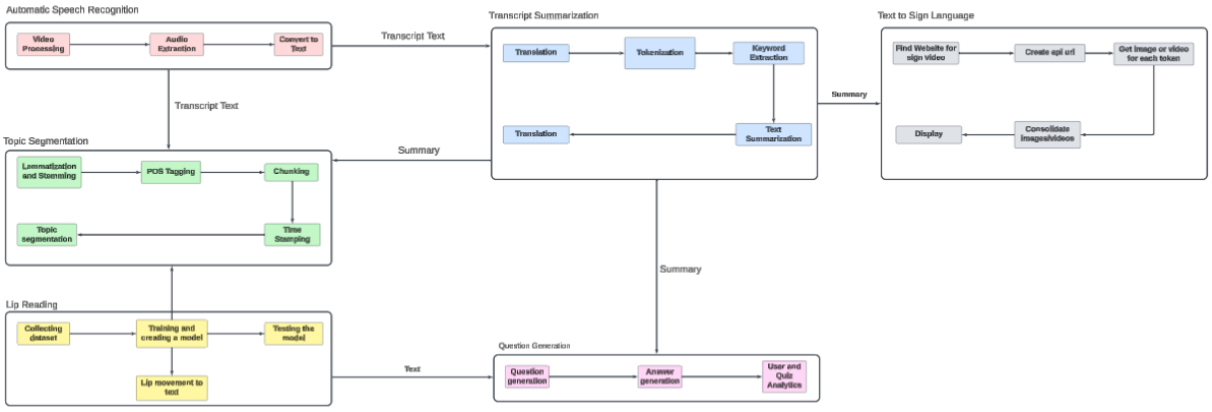


Figure 3.1: System Architecture

3.2 Component Design

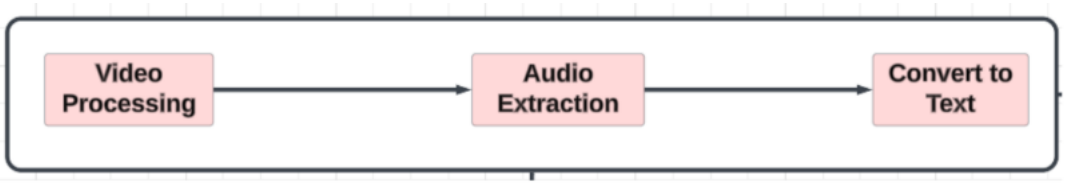


Figure 3.2: ASR Module

The Automatic Speech Recognition module converts spoken language in audio files into text. This module is important for several applications, like transcription, speech to text systems, and indexing content. We are using SpeechRecognition library along with pydub to process and transcribe an audio file either extracted from a video or downloaded from Youtube. The advantage with processing audio is that in case any video has embedded an audio file, it would first be extracted with moviepy, while in case of an online video, the audio gets downloaded using yt-dlp. After having obtained the required audio from either of the ways, pydub makes conversion from MP3 to WAV since SpeechRecognition library works well with audio files in WAV format. It is then transcribed using the SpeechRecognition library. A Recognizer() is used to load the audio WAV file, which is done by calling Google Web Speech API, the most widely used speech recognition service. The recognized speech is translated into text by the function recognizegoogle(). Upon successfully recognizing the speech and converting to text, the text is printed out;

otherwise UnknownValueError will occur when the speech was unclear. RequestError will happen when there is something problematic with the internet connection, which must be handled. This ASR module has huge implications in making the transcription of speech into text by the utilization of tools highly readable and operational. Based on the exploitation of open-source toolkits including SpeechRecognition, pydub, and Google Web Speech API, transcription of speech is provided in a time-efficient and scalable way.

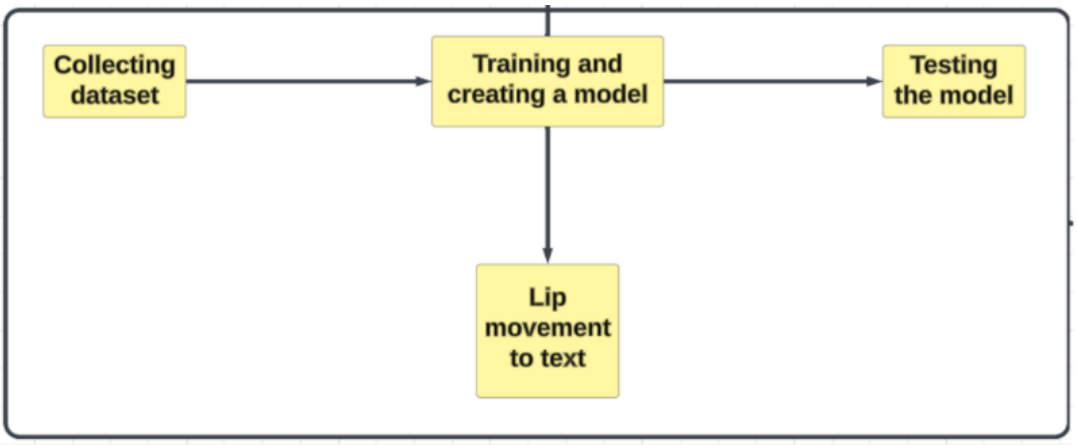


Figure 3.3: Lip Reading Module

This lipreading module employs a deep learning architecture designed to interpret lip movement video frames and predict spoken words. The model integrates 3D Convolutional Neural Networks (3D CNNs) and Recurrent Neural Networks (RNNs) to extract spatiotemporal features from video sequences, capturing the dynamic nature of lip movements. The architecture begins with three 3D convolutional layers, each followed by a ReLU activation function and a max-pooling layer, allowing the network to learn spatial features while reducing dimensionality. A TimeDistributed(Flatten()) layer is then applied to preserve the features of each frame and prepare them as sequences for temporal modeling.

These extracted features are passed to two bidirectional Long Short-Term Memory (LSTM) layers, which enable the model to consider both past and future context, enhancing recognition accuracy based on visual information alone. The final output layer is a dense layer with a softmax activation function, used to classify the features into characters—forming the basis for generating a textual representation of speech. Training is performed using the fit() function, running the model on video-text paired datasets for 100 epochs, along with several callbacks such as model checkpointing, learning rate schedul-

ing, and monitoring to improve efficiency. Dropout layers are also included to prevent overfitting by randomly deactivating neurons during training. This robust architecture effectively bridges the gap between visual speech and text interpretation, enabling accurate lipreading even under challenging conditions. To evaluate the model's performance, standard metrics were used: Word Error Rate (WER), Character Error Rate (CER), and BLEU score. WER accounts for the number of substitutions, deletions, and insertions relative to the reference word count, while CER measures similar errors at the character level. The BLEU score, which uses a modified precision method, evaluates how many candidate words match those in the reference. Together, these metrics provide a comprehensive assessment of the model's accuracy in translating visual inputs into coherent textual outputs.

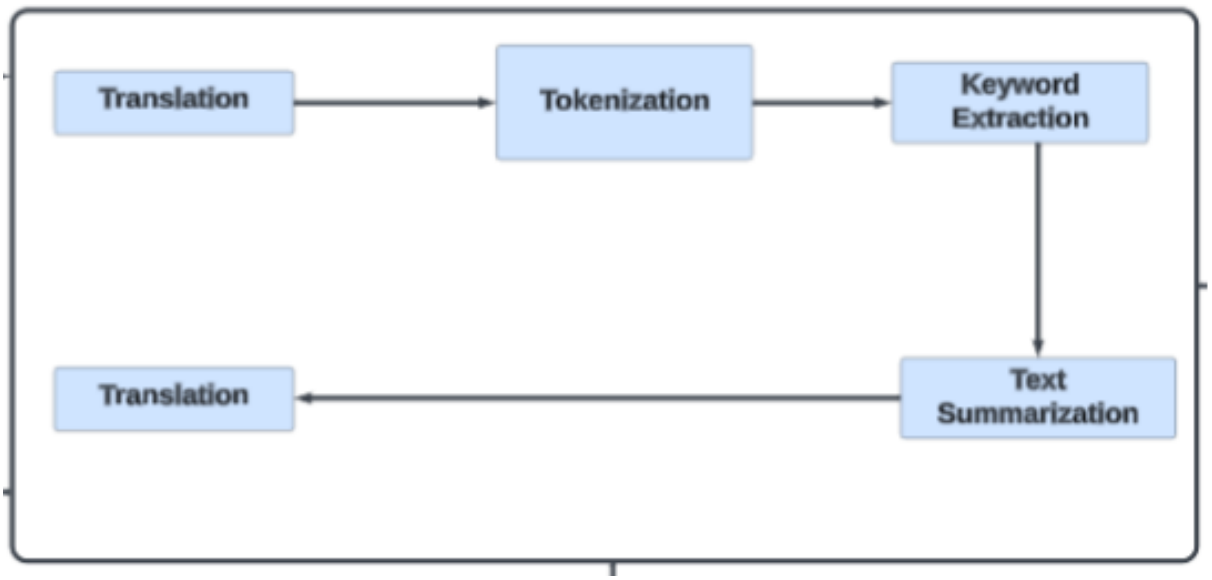


Figure 3.4: Text Summarization Module

The Transcript Summarization combination starts by processing the video content, hence converting video to audio, and then doing the speech to text conversion. It first converts the transcribed text into English to keep everything standardized during processing and for all future operations, as a consistent language model can be used. Subsequently, the text is tokenized into individual words, while stop words like those common and inconsequential words like "and," "the," and "is" that basically hardly carry any meaning of text are eliminated. The remaining keywords then form the basis for the summarization process. These keywords pinpoint the vital terms in the text while serving as impor-

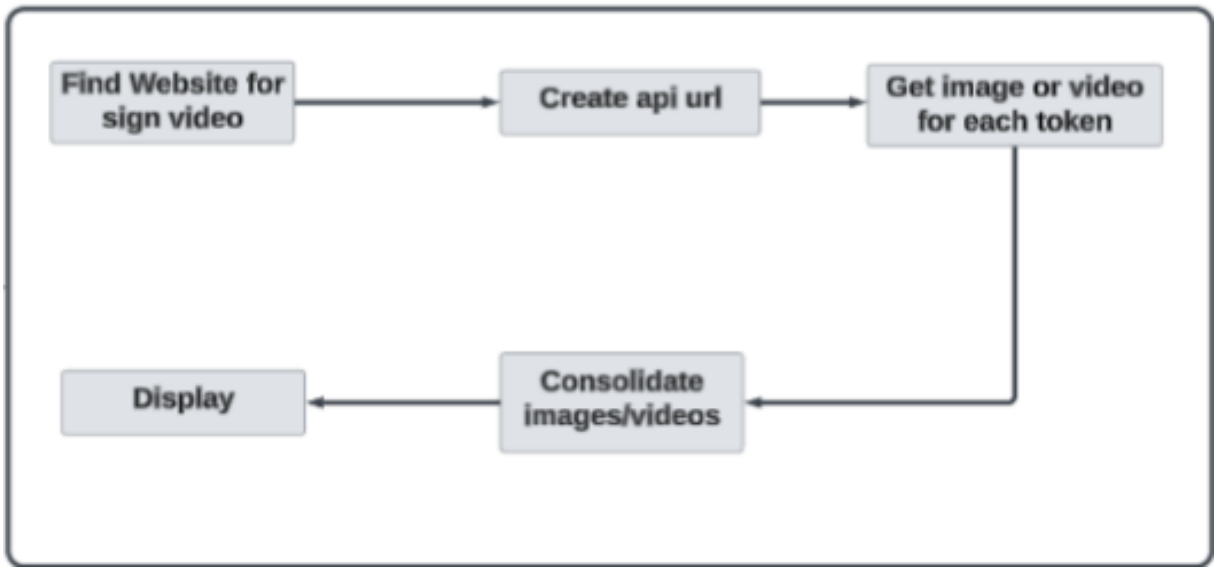


Figure 3.5: Text to Sign Module

tant inputs to the summarization models, thus speeding up the process and enhancing accuracy. After obtaining the keywords, the module runs both extractive and abstractive summarization. The extractive summarization method summarizes the transcript by extracting sentences that are most important from the source text based on the keywords while keeping the structure of the passage from the transcript intact. The other method is abstractive summarization which creates new sentences and rewrites the content in a shorter, smoother way while capturing the core meaning or essence of the text. After both forms of summarization are done, the summarized output is translated into the source language so as to keep multilingual accessibility. Hence, in brief, the processing helps retain correctness and comprehensibility in both forms of summarization while ensuring flexibility in the consideration of different linguistic audiences.

The Text to Sign Language Model starts by taking the summarized transcript and removing all the stopwords from it so as to keep only the meaningful words needed for translation purposes. After that, each of these words is processed to create an API URL that goes out and gets the sign language videos from an external site. This step provides the model with the video for each word it needs for each word. However, if a video for a specific word isn't available, the model goes for finger spelling and presents images of hand signs for each individual letter of the word. This fallback or backup method ensures that all words are represented in sign languages even if there is no direct video to support for easy understanding. Once the videos and/or images are ready, the model brings together

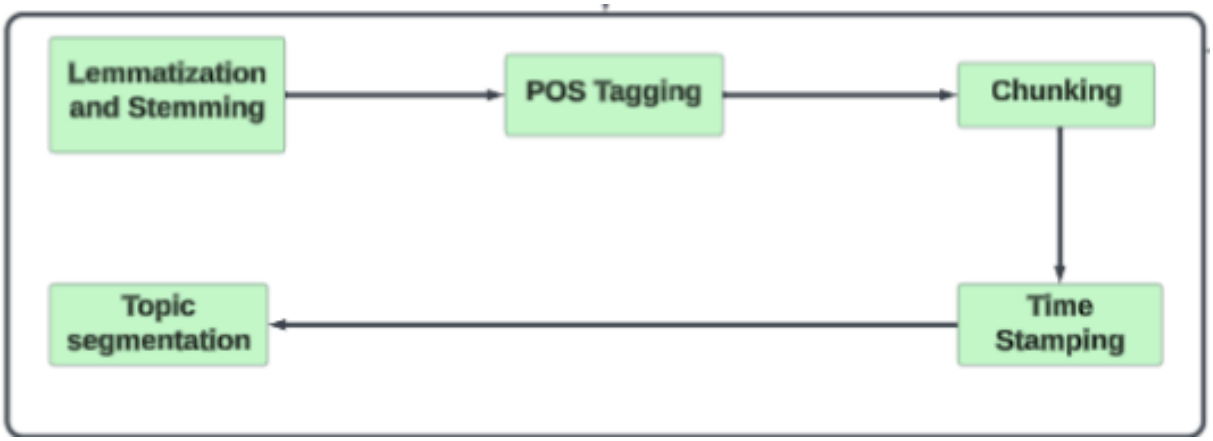


Figure 3.6: Topic Segmentation Module

the different types of media into a structured sequence. Each single video and set of photos of letters for a word is shown subsequently and in the right order that maintains sign language's smoothness and grammar in communication. Such a dynamic approach not only provides easy readability for the transcript scrolling through a sign language but also encourages widening the function toward the hearing-impaired community. The unbroken joining of videos and images makes the translated content smooth and accurate as it links effortlessly between text to sign language.

The Topic Segmentation Module is meant to split the text into logical segments and optionally assign meaningful topic titles, and is particularly helpful for a long transcription of speeches, lectures, or audio recordings since it organizes the content into structured topics with timestamps. It uses KeyBERT for keyword extraction -the latest and quite efficient model for any keyword extraction-the model for automatic topic identification within the text. The KeyBERT model is initiated first, which does the keyword extraction from a text input. It then switches to the generatetopic(), which, given a sentence or paragraph, hooks up a succinctly built topic with the set of key phrases/words detected using KeyBERT. It chooses the most relevant keyword/key phrase, capitalizing each word for readability. If none is selected, a default topic title could be, for example, General AI Topic. This method allows for every text section to have an appropriate theme title easily identified by its topic. The next function, segmenttext(), is responsible for breaking down the input text into smaller portions and generating relevant categories based on them. To make it presentable, this text will then be wrapped to a maximum width with the help of textwrap.wrap(). It gets lined through with all output/printed lines taken in by the

text to define the right topic, given by the generatetopic() method. Also, the program saves themes in a storage called topicsdictionary, so that there are not repeated topics: it does not allow a certain section to be assigned with a new heading, unless it is otherwise. Each of the sections is given a timestamp: which assumes some default value (0 seconds, for example), and increases every section by fixed value (5.2 seconds, for example). This timestamp creates an organized flow and allows attaching a topic to defined time slots in a transcription for audio or video. The final output comprises well-structured modules: topical weldings bounded with a time range.

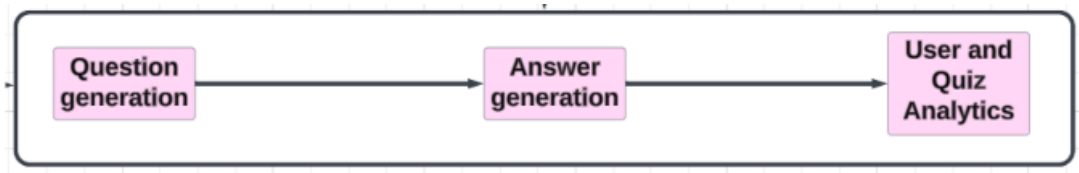


Figure 3.7: MCQ Module

The script in Python generates interactive quiz questions from the summary of a text. It does so in one of three ways: Fill in the Blanks, MCQs, or Multiple Blanks. First, it makes use of imports for the really necessary ones like random and nltk for sentence tokenization while checking if the required data have been downloaded. All the keywords and the summary will be converted into lowercase to allow for case-insensitive matching. The performance by the user in various question types is recorded in a dictionary. The rest of the question functions have similar structures such as taking a random keyword and pulling out a few sentences containing it, only this time, modifying them into questions. Fill in the Blanks replaces one word with a blank, MCQs provide four answer choices, one being the correct one and three random distractors, and Multiple Blanks replaces two words with blanks. The main() function provides a menu for the user to select the question type and the number of questions, and it runs inside a continuous loop until the user chooses to exit. Moreover, all user inputs are validated: only positive numbers will be accepted. Performance Data will be saved in a file for further insights. The tool offers strong protection for all incorrect input and opens a window into slots for expanding features such as a graphical performance report.

3.3 Algorithm Design

3.3.1 Video to Text Transcription Algorithm

1. Start

1.1 The process starts with a video file in any format, such as .mp4, etc.

2. Extract Audio from Video

2.1 Extract the audio track from the video using a library like ffmpeg.

3. Preprocess Audio (Optional)

3.1 Preprocess the audio if necessary (e.g., noise reduction).

4. Transcribe Audio to Text

4.1 Use an Automatic Speech Recognition (ASR) system to transcribe the audio into text.

5. Read the Audio File:

5.1 Load the extracted audio file in the correct format.

5.2 Use library speech recognition to transcribe the audio and return the text.

5.3 Save the transcription to a text file

6.Stop

3.3.2 Text Summarization algorithm

1. Start

2. Translate the transcript to english

3. Tokenize text into words and sentences

4. Remove stopwords and apply pos tagging to get key words

5. Map each word to its frequency using wordFreq

6. Initialize a map sentFreq=[]

7. Repeat steps 7.1 to 7.2 for sent in sentences

7.1 Repeat steps 7.1.1 to 7.1.2 for word in words

7.1.1 Check if word in wordFreq then

7.1.1.1 Check if sent is not in sentFreq then

7.1.1.1.1 sentFreq[sent]=wordFreq[word]

7.1.1.2 Otherwise

7.1.1.2.1 sentFreq[sent]+=wordFreq[word]

7.1.2 Increment word

7.2 Increment sent

8. Sort the sentFreq summary in reverse order of word frequency

9. Print top 80 percent of the summary

10. Stop

3.3.3 Text to Sign Language

1. Start

2. Preprocess the summarized transcript

3. Combine the preprocessed tokenized words to a sequence

4. For each word, retrieve sign image or video.or avatar

5. Combine them into single video

6. Play the video/animation

7. Stop

3.3.4 Lip-Reading Algorithm

1. Start

2. Video Input: Capture the video of the person speaking.

3. Frame Extraction: Divide the video into individual frames.

4. Region of Interest (ROI) Detection: Use facial landmark detection to locate the mouth region.

5. Preprocess Frames:

5.1 Normalize the frames by resizing the mouth region (e.g., 50x100 pixels).

6. Remove Irrelevant Frames:

6.1 Discard frames where no mouth movement is detected (e.g., silence).

7. Feature Extraction:

7.1 Use Convolutional Neural Networks (CNN) to extract spatial features.

7.2 Use Recurrent Neural Networks (RNN), like GRU or LSTM, for temporal feature extraction.

8. Viseme Mapping:

8.1 Map each frame to a corresponding viseme (a visual representation of phonemes).

9. Sentence Generation:

9.1 Use a sequence-to-sequence model to generate sentences from viseme sequences.

9.2 Apply the Connectionist Temporal Classification (CTC) layer for sequence alignment.

10. Post-processing: Correct common lip-reading errors using a language model.

11. Display Result: Output the generated text as subtitles.

12. Stop

3.4 Data Flow Diagrams (DFD)/ USE CASE diagram

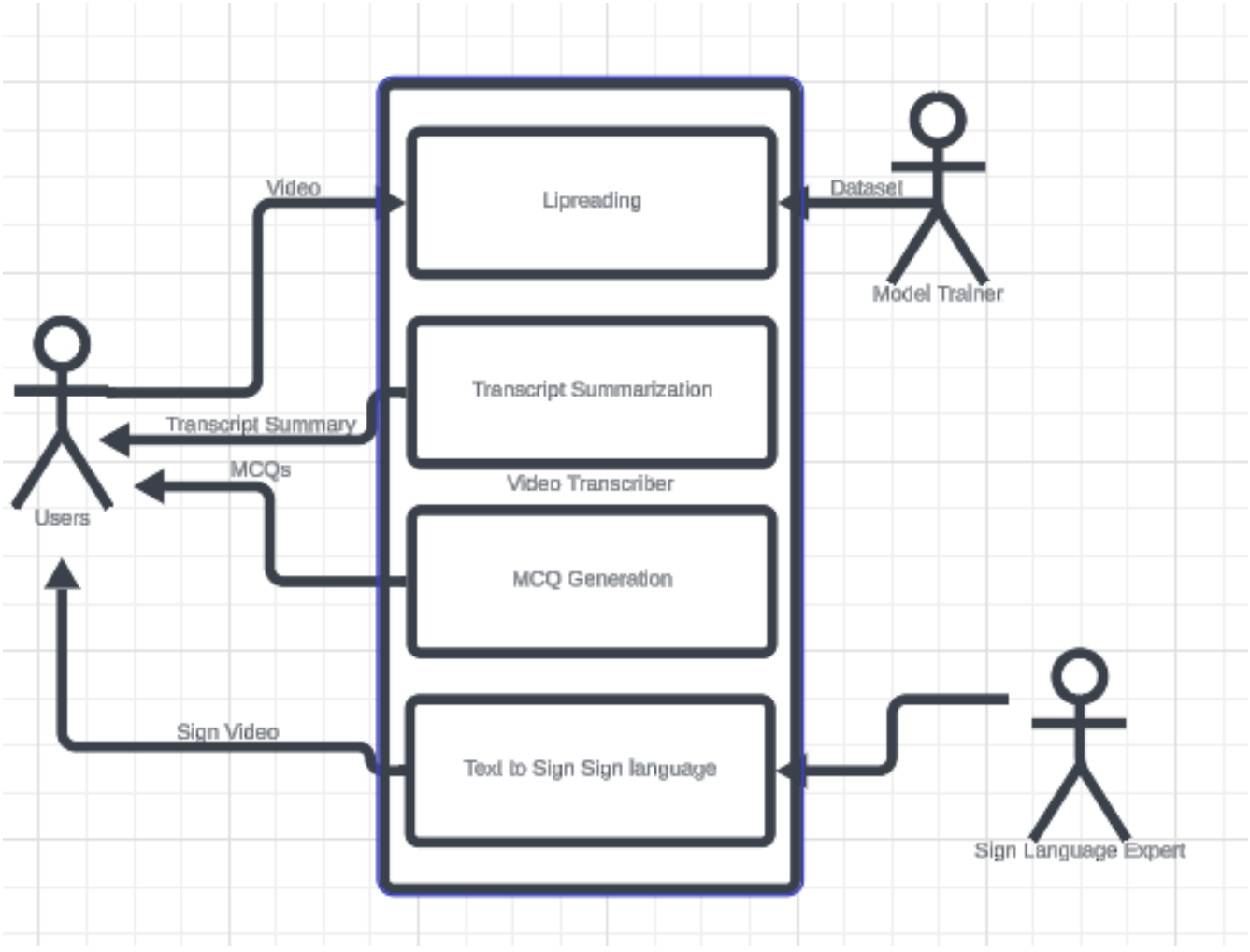


Figure 3.8: Use Case Diagram

3.5 Tools and Technologies

3.5.1 Software Requirements

Operating System: Linux-based system (like Ubuntu etc.) or Windows.

Programming Languages: Python, HTML, JS, CSS

Web Framework: React, Flask, or Django

Libraries:

- Ntlk
- Googletrans
- Pytube
- movie.py
- Speechrecognition

3.5.2 Hardware Requirements

CPU: 4-8 cores (e.g., Intel Core i7 or AMD Ryzen 7)

RAM: 16-32 GB

Storage: 512 GB SSD

3.6 Data set Identified

LipNet - Lip reading

Handspeak Library - Text to Sign Language

3.7 Module Divisions and work break down

3.7.1 Module Division

- Automatic Speech Recognition
- Transcript summarization and Text to Sign Language

- Topic Segmentation and MCQ Generation
- Lip Reading
- Web platform

3.7.2 Work Breakdown

Daryl

- Translation
- Identifying and extracting keywords
- Summarizing large transcript and highlighting key information
- Dynamic user transcript editing
- Searching and filtering
- Text to speech
- Text to sign language.
- Web development and designing
- Other member's coding work

Melissa

- Processing uploaded video
- Extracting audio from video
- Speech to text conversion
- Time stamping
- Topic segmentation
- Speaker identification
- Web development and designing

- Other member's coding work

Eshaan

- Collecting data for lip reading
- Train and create a model for lip reading
- Lip reading to text conversion
- Topic segmentation
- Speaker identification
- Time stamping
- Web designing ideas

Milin

- Question and answer generation
- Question format selection
- User analytics and quiz analytics.
- Summarization
- Topic segmentation
- Web designing ideas

3.8 Key Deliverables

- Lipreading Model
- Website
- Transcript Summary
- MCQ Questions
- Performance Dashboard
- Video to Audio to Text

3.9 Project Timeline

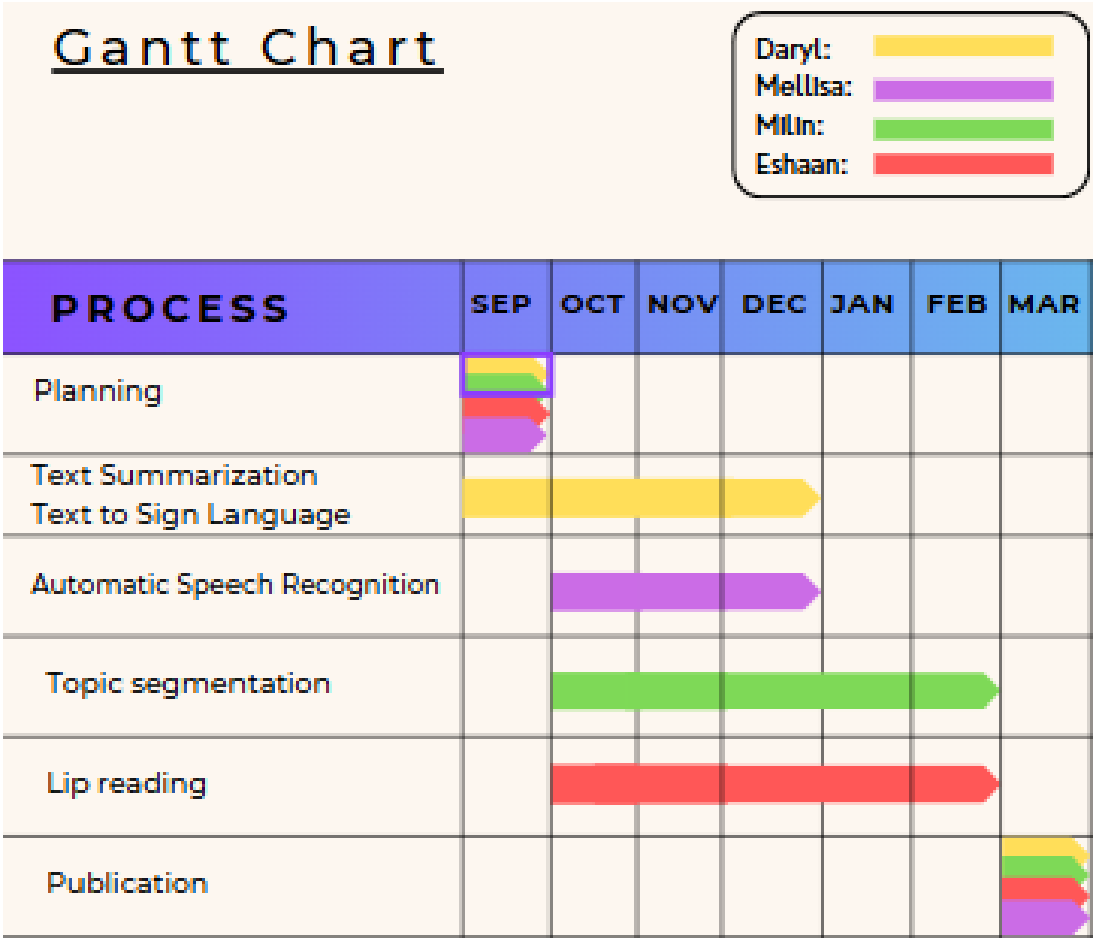


Figure 3.9: Gantt Chart

The design phase of the Video Transcriber project establishes a solid framework for development. By defining the high-level system architecture, key components, algorithms, and data flow, this chapter ensures a clear understanding of the system's functionality and technical requirements.

Breaking down the project into modules and tasks allows for systematic development, while the identification of tools, technologies, and datasets ensures resource optimization. The project timeline provides a realistic schedule, ensuring milestones are met and the final deliverables are achieved within the specified timeframe.

In conclusion, this chapter acts as the blueprint for creating an efficient and accurate video transcription system, setting the stage for successful implementation and achieving the project's objectives.

Chapter 4

Experiments and Results

This chapter presents the experiments conducted to evaluate the performance and effectiveness of the developed Automatic Speech Recognition (ASR) system. Each module—audio processing, lipreading-based speech recognition, transcript summarization, text-to-sign language conversion, topic segmentation, and interactive quiz generation—was tested under various conditions to assess accuracy, speed, and usability. The experiments involved datasets with diverse speech patterns, noise levels, and speaker variations to ensure robustness. We analyze the results obtained from different deep learning models and natural language processing techniques, comparing their efficiency in handling real-world scenarios. The findings highlight the strengths and limitations of each module, demonstrating the overall feasibility and scalability of the ASR system.

4.1 Video to Audio Conversion

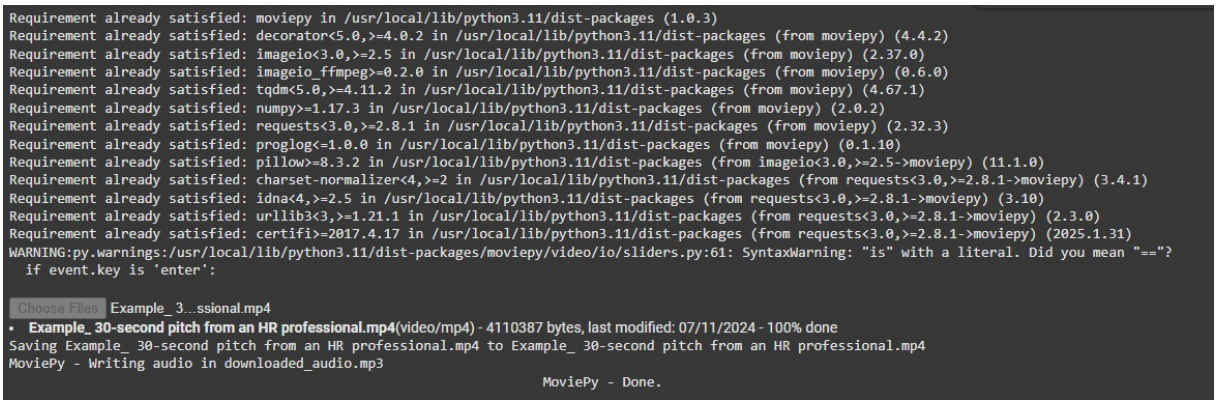


Figure 4.1: Downloaded Video to Audio

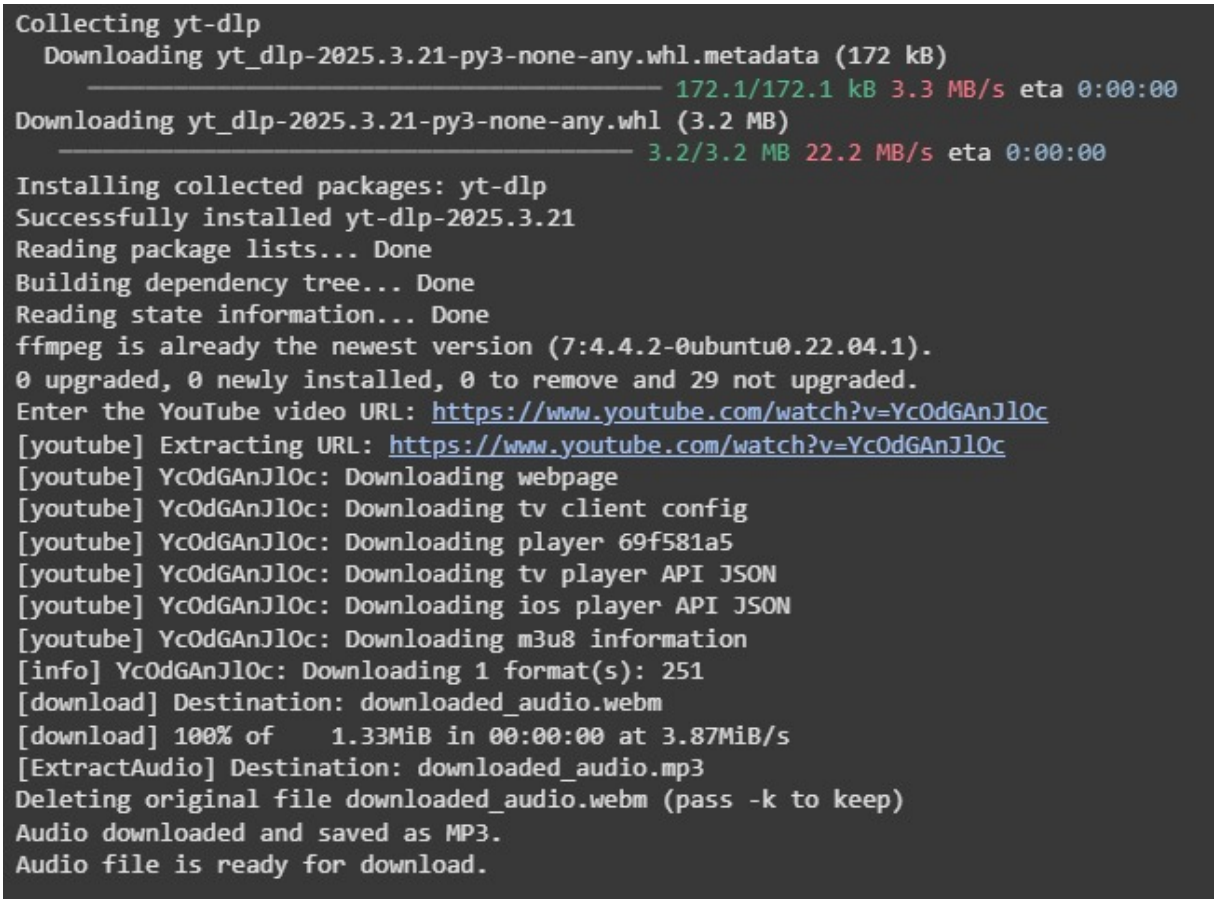


Figure 4.2: Video to Audio using video URL

4.2 Audio to Transcribe Text conversion

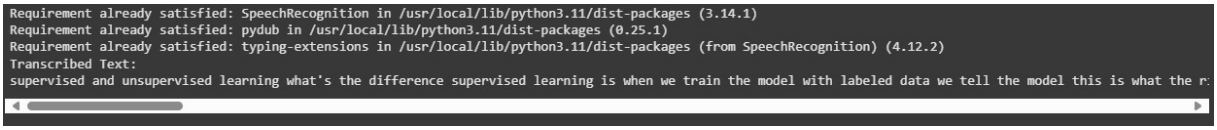


Figure 4.3: Transcribed Text

4.3 Topic Segmentation

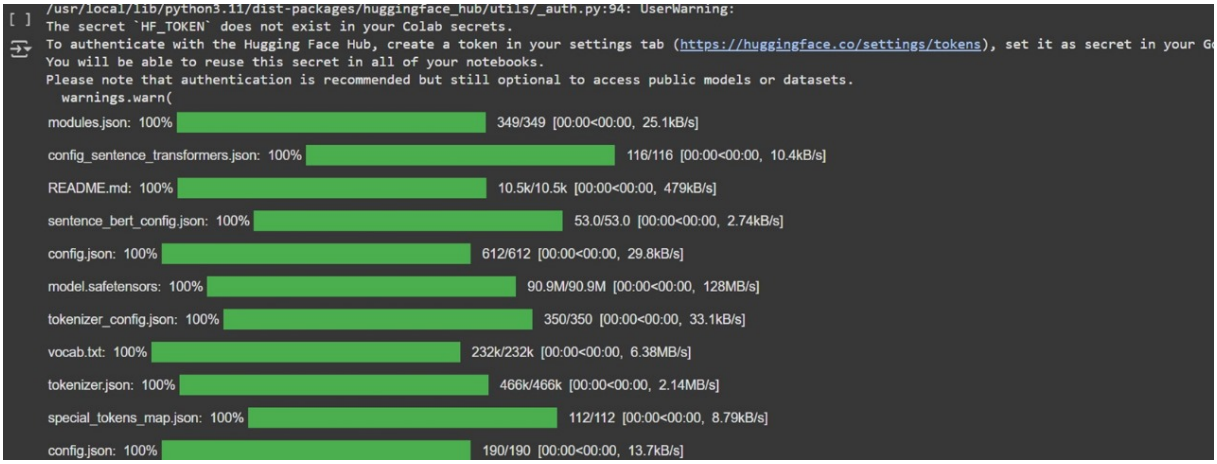


Figure 4.4: Topic Segmentation

4.4 Time Stamping

```
=== Topic 0 ===
[0.00s - 5.20s] Supervised Unsupervised

=== Topic 1 ===
[5.20s - 10.40s] Info Unsupervised

=== Topic 2 ===
[10.40s - 15.60s] Model Unlabeled

=== Topic 3 ===
[15.60s - 20.80s] Use Type

=== Topic 4 ===
[20.80s - 26.00s] Purchases History

=== Topic 5 ===
[26.00s - 31.20s] Model Customers

=== Topic 6 ===
[31.20s - 36.40s] Buy Jelly

=== Topic 7 ===
[36.40s - 41.60s] Cart Supervised

=== Topic 8 ===
[41.60s - 46.80s] Shoppers Spend

=== Topic 9 ===
[46.80s - 52.00s] Related Products
```

Figure 4.5: Time Stamping

4.5 Text Summarization

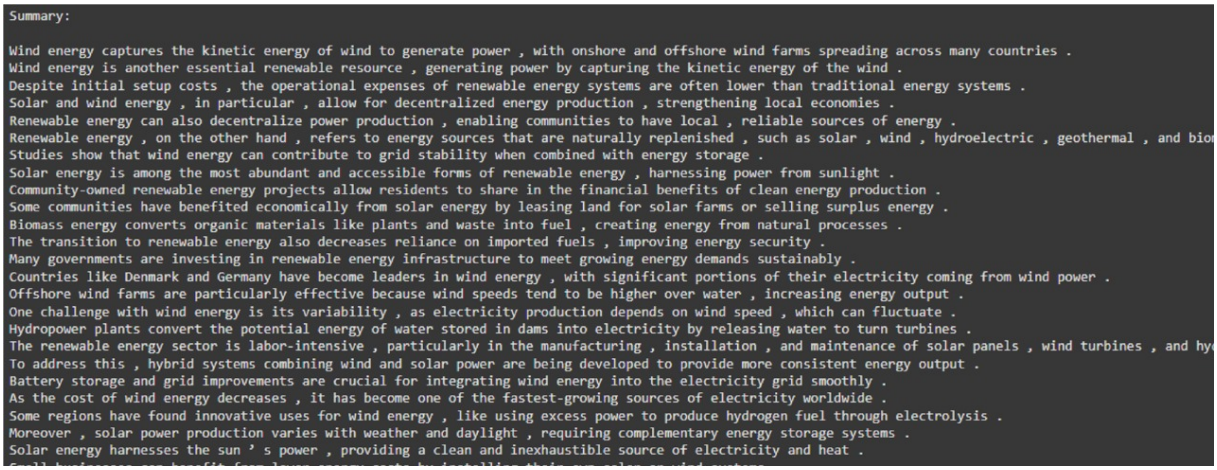


Figure 4.6: Summarized text

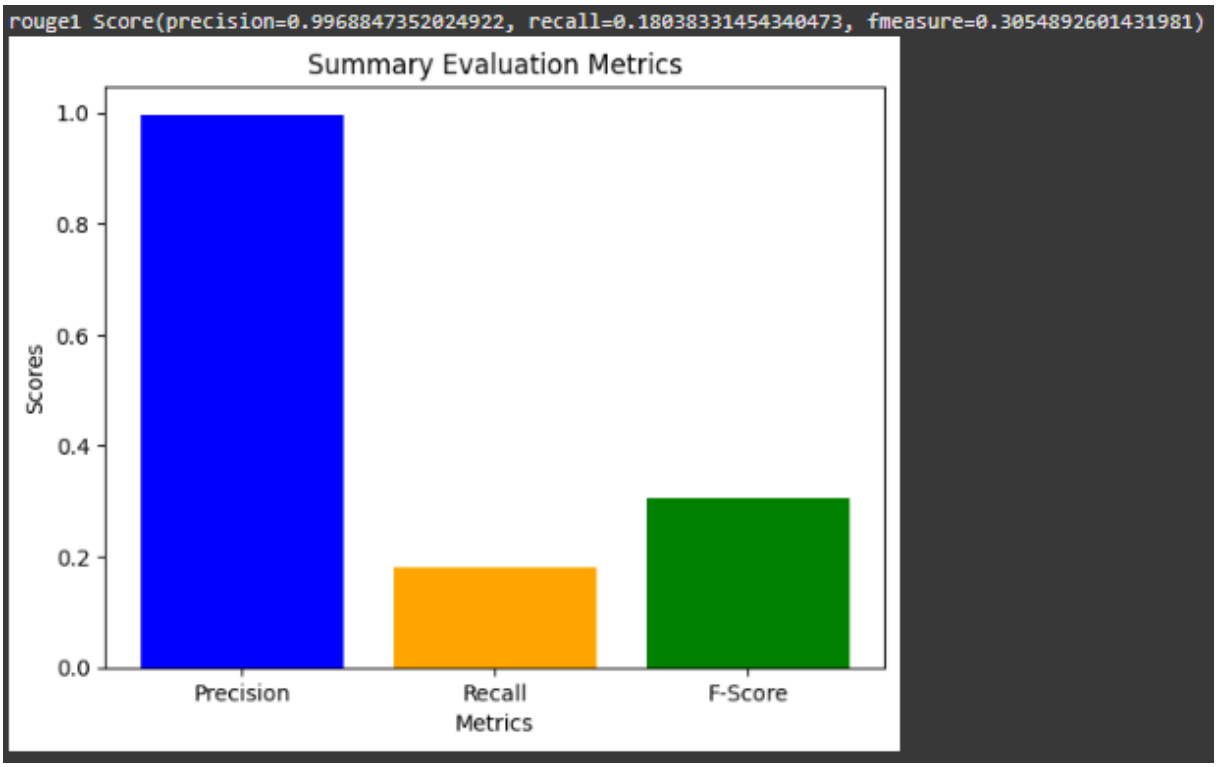


Figure 4.7: Extractive Summary Rouge1 Score

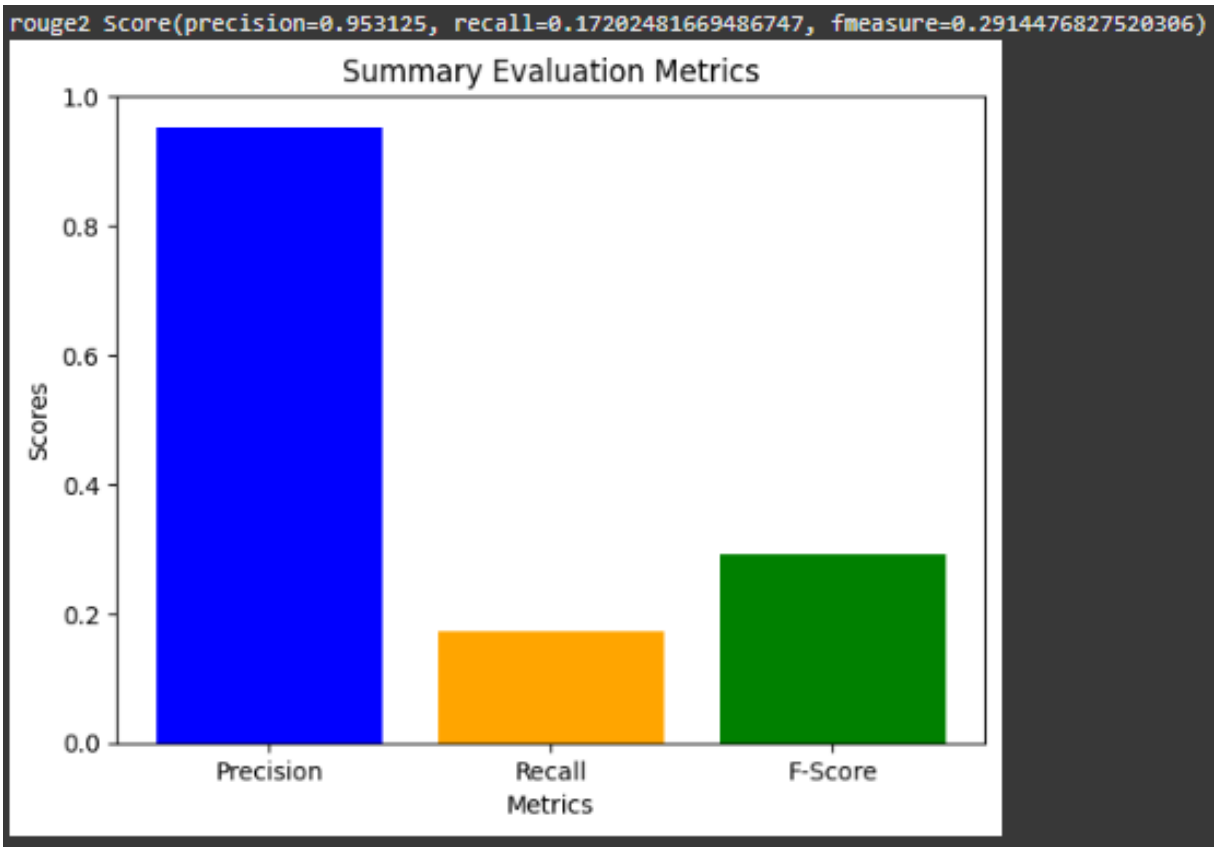


Figure 4.8: Extractive Summary Rouge2 Score

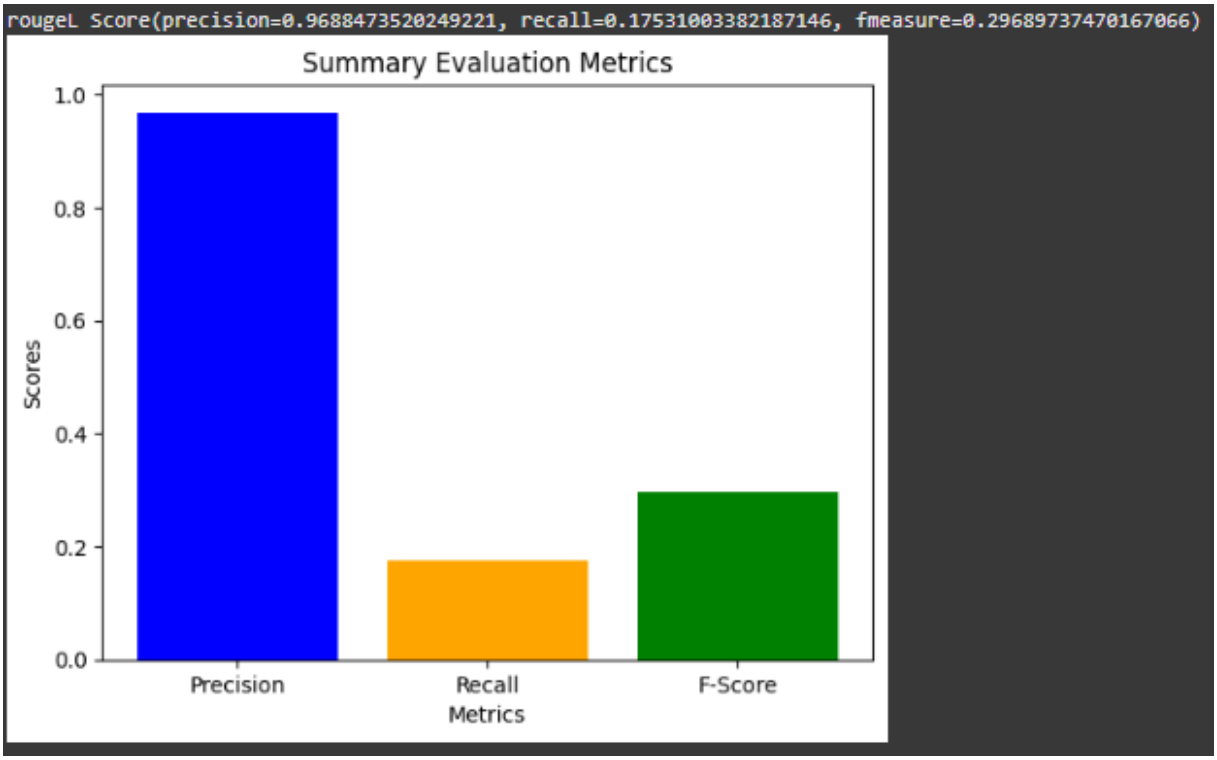


Figure 4.9: Extractive Summary RougeL Score

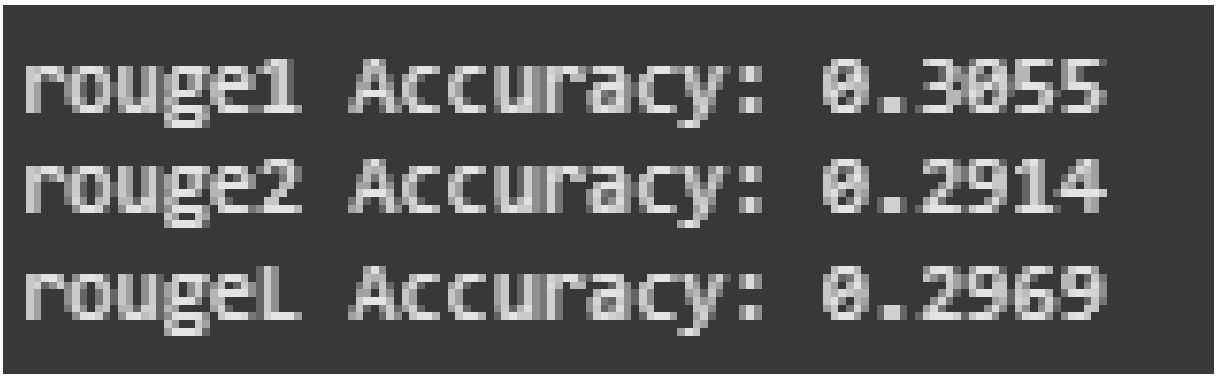


Figure 4.10: Extractive Summary Accuracy

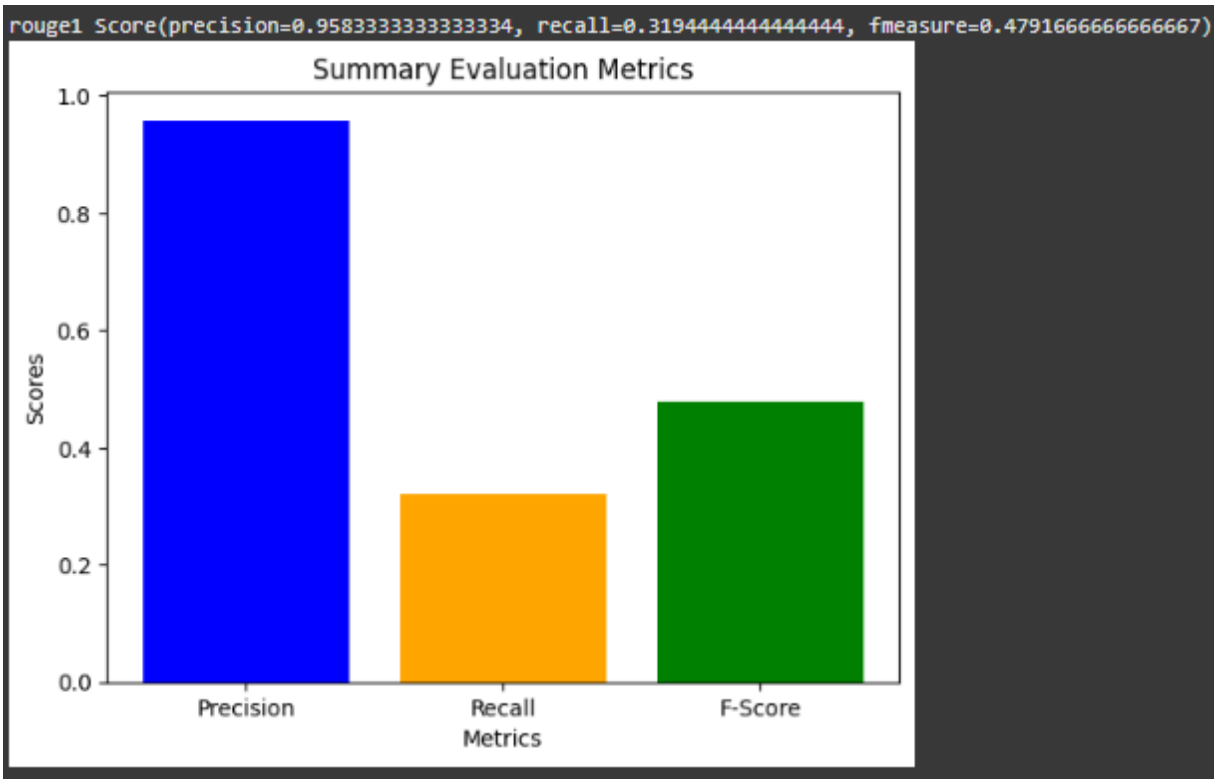


Figure 4.11: Abstractive Summary Rouge1 Score

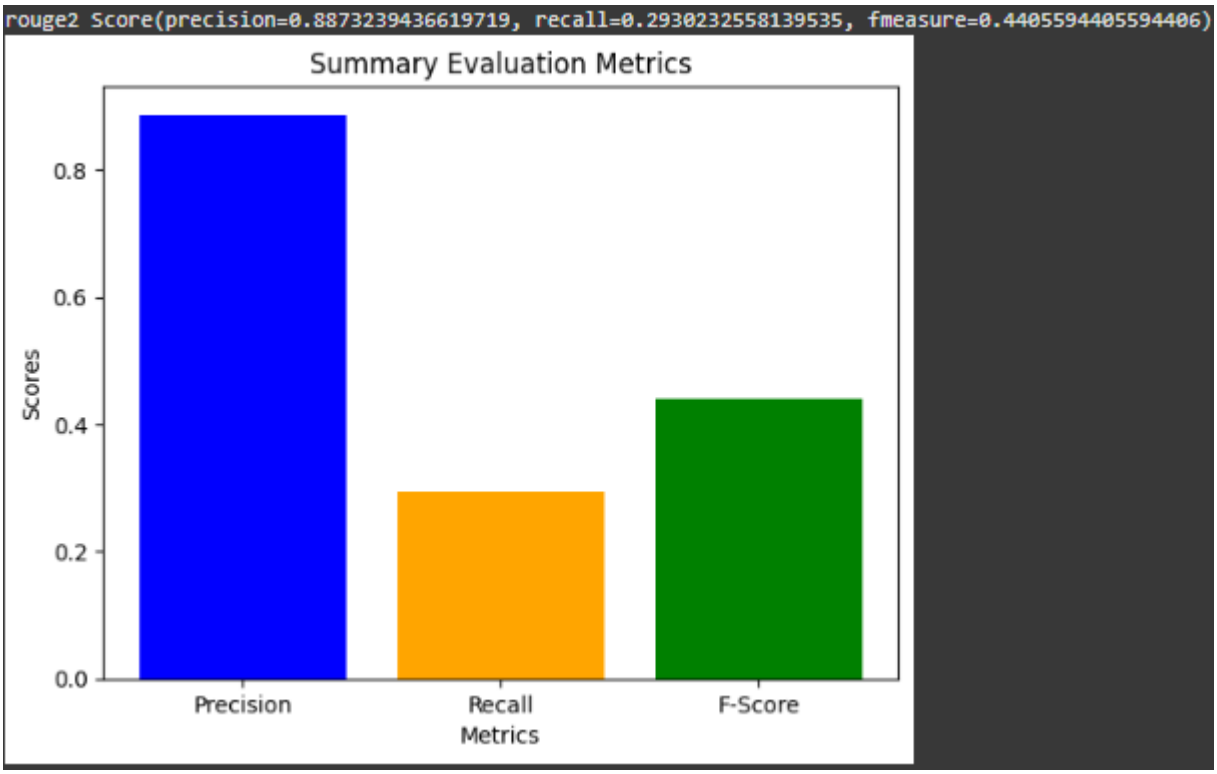


Figure 4.12: Abstractive Summary Rouge2 Score

Figure 4.13: Abstractive Summary RougeL Score

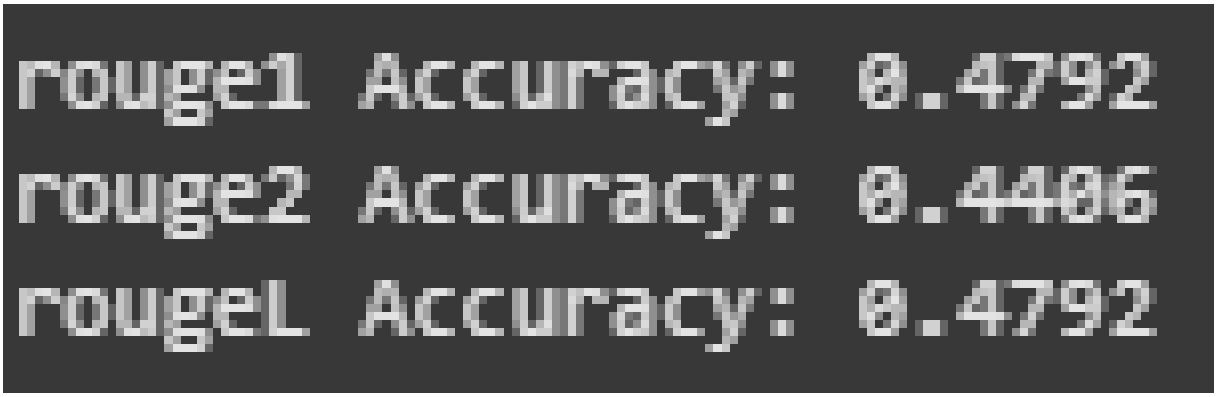


Figure 4.14: Abstractive Summary Accuracy

4.6 Keyword Information Highlights

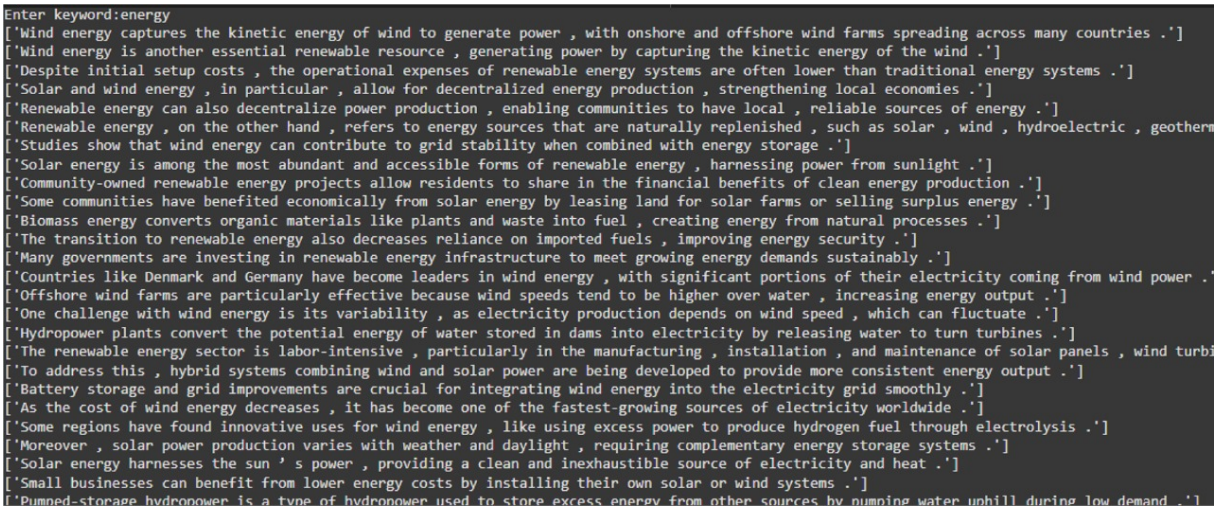


Figure 4.15: Keyword Information

4.7 Sign Language

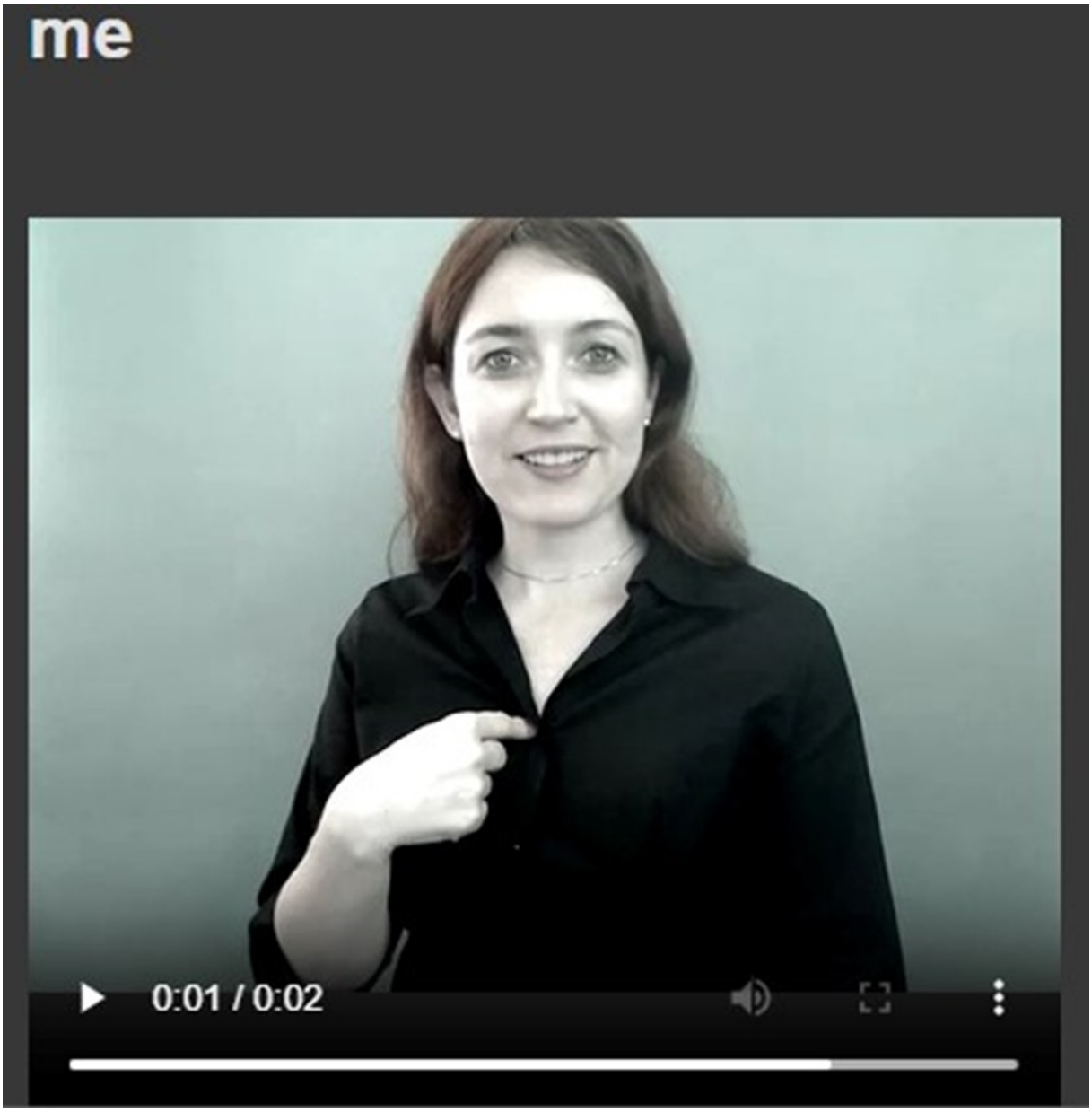


Figure 4.16: Human Representation

Figure 4.17: Hand Representation

4.8 Lip Reading

```
[ ]  model.summary()
```

Model: "sequential"

Layer (type)	Output Shape	Param #
conv3d (Conv3D)	(None, 75, 46, 140, 128)	3,584
activation (Activation)	(None, 75, 46, 140, 128)	0
max_pooling3d (MaxPooling3D)	(None, 75, 23, 70, 128)	0
conv3d_1 (Conv3D)	(None, 75, 23, 70, 256)	884,992
activation_1 (Activation)	(None, 75, 23, 70, 256)	0
max_pooling3d_1 (MaxPooling3D)	(None, 75, 11, 35, 256)	0
conv3d_2 (Conv3D)	(None, 75, 11, 35, 75)	518,475
activation_2 (Activation)	(None, 75, 11, 35, 75)	0
max_pooling3d_2 (MaxPooling3D)	(None, 75, 5, 17, 75)	0
time_distributed (TimeDistributed)	(None, 75, 6375)	0
bidirectional (Bidirectional)	(None, 75, 256)	6,660,096
dropout (Dropout)	(None, 75, 256)	0
bidirectional_1 (Bidirectional)	(None, 75, 256)	394,240
dropout_1 (Dropout)	(None, 75, 256)	0
dense (Dense)	(None, 75, 41)	10,537

Total params: 8,471,924 (32.32 MB)
Trainable params: 8,471,924 (32.32 MB)
Non-trainable params: 0 (0.00 B)

Figure 4.18: The Model

```
[ ] import jiwer
    import Levenshtein
    from nltk.translate.bleu_score import sentence_bleu

    # Compute WER (Word Error Rate)
    wer = jiwer.wer(expected_text, predicted_text)

    # Compute CER (Character Error Rate)
    cer = Levenshtein.distance(expected_text, predicted_text) / len(expected_text)

    # Compute BLEU score
    bleu_score = sentence_bleu([expected_text.split()], predicted_text.split())

    # Display results
    print(f"Word Error Rate (WER): {wer:.4f}")
    print(f"Character Error Rate (CER): {cer:.4f}")
    print(f"BLEU Score: {bleu_score:.4f}")
```

```
Word Error Rate (WER): 0.0000
Character Error Rate (CER): 0.0000
BLEU Score: 1.0000
```

Figure 4.19: Model Accuracy

```
sample = load_data(tf.convert_to_tensor('./data/s1/lwbl8p.mpg'))
print('~'*100, 'REAL TEXT')
#predicted=[tf.strings.reduce_join([num_to_char(word) for word in sentence]) for sentence in sample[1]]
[tf.strings.reduce_join([num_to_char(word) for word in sentence]) for sentence in [sample[1]]]
```

```
alignment =  tf.Tensor(
[39 12  1 25 39 23  8  9 20  5 39 19 16 39  2 25 39 12 39  5  9  7  8 20
 39 16 12  5  1 19  5], shape=(31,), dtype=int64)
data/s1/lwbl8p.mpg
data/alignments/s1/lwbl8p.align
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~ REAL TEXT
[<tf.Tensor: shape=(), dtype=string, numpy=b' lay white sp by l eight please'>]
```

Figure 4.20: Output

4.9 Question Generation

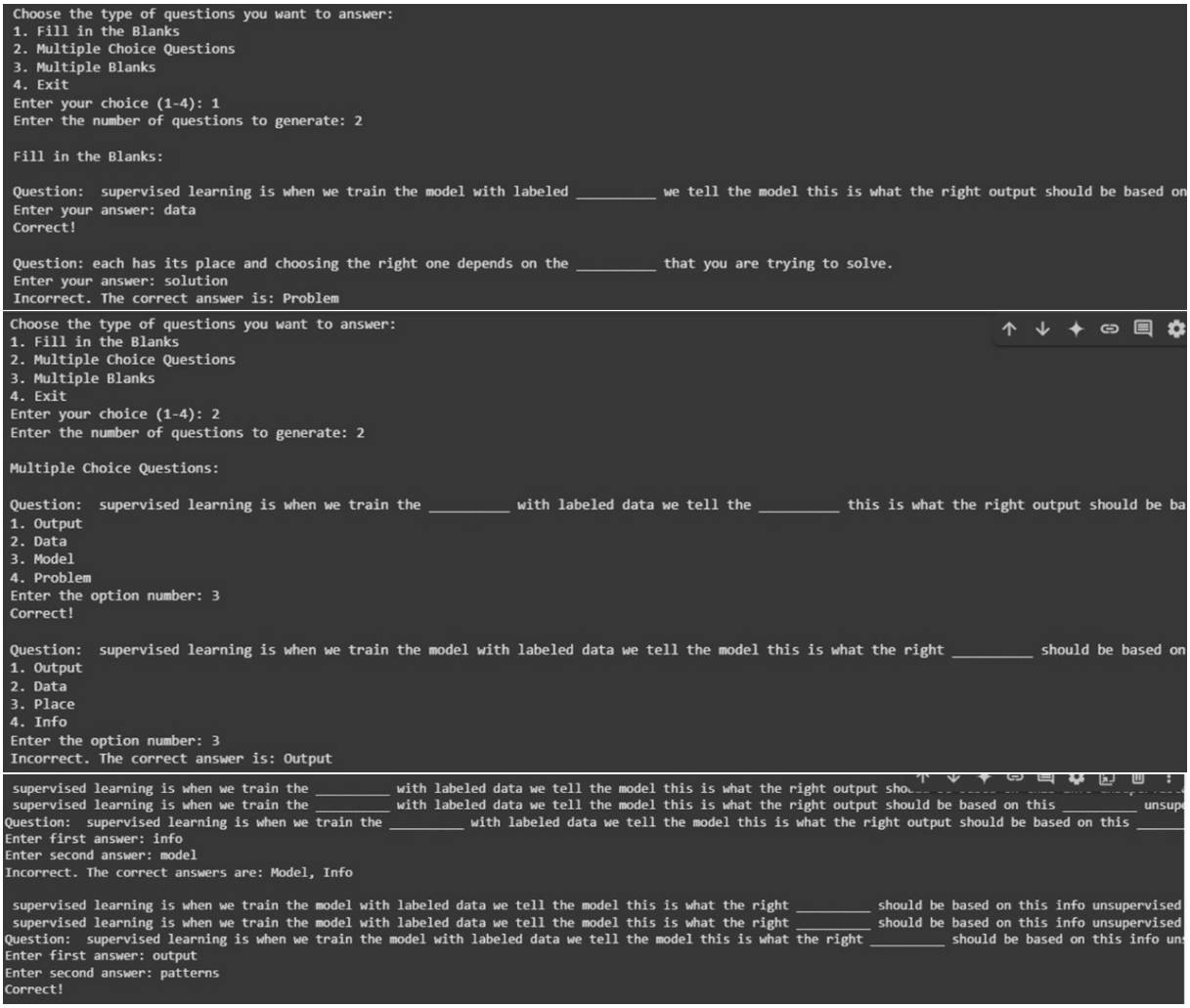


Figure 4.21: Generated Questions

4.10 The Website

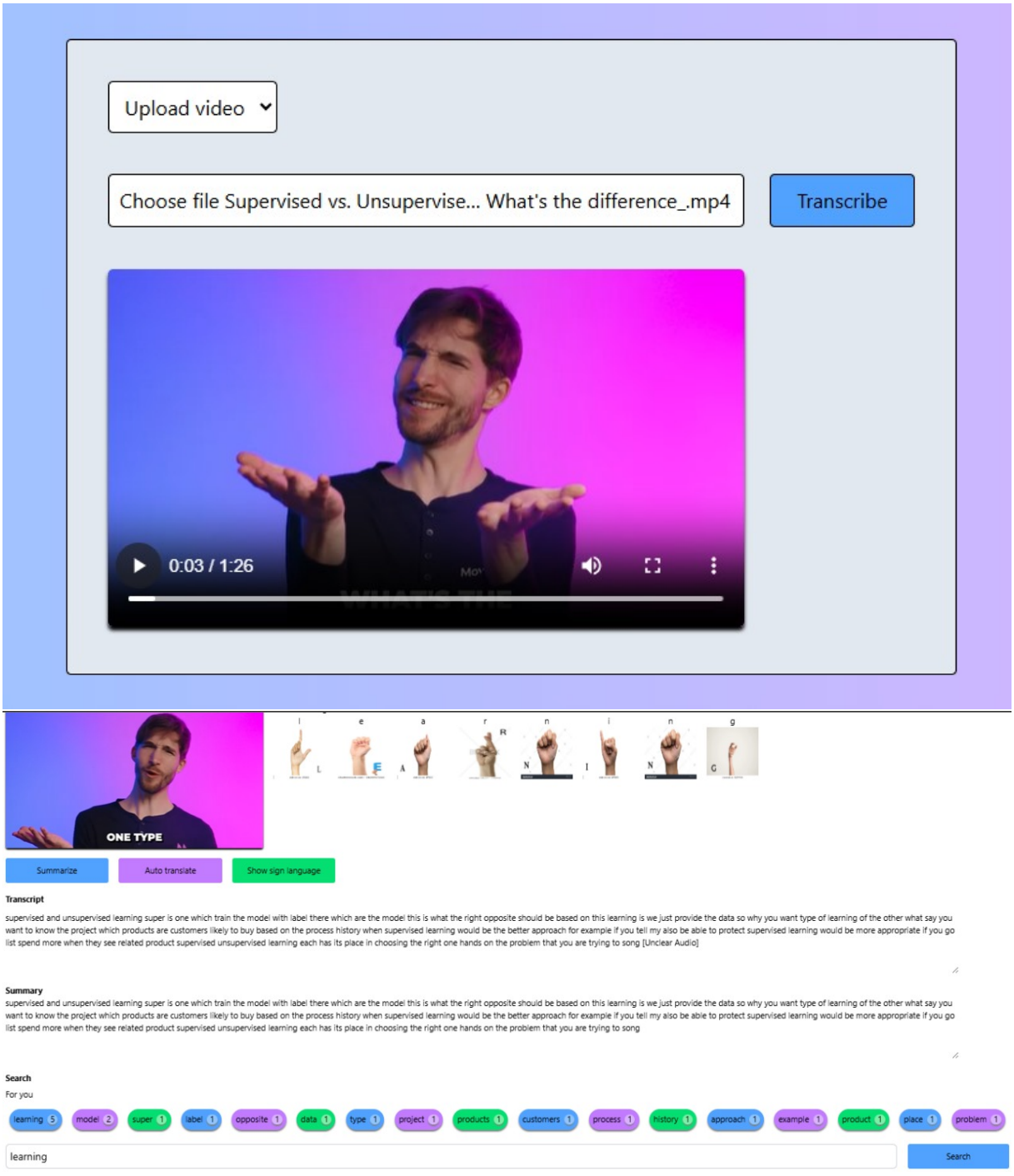


Figure 4.22: Preview page

In conclusion, the evaluation of the ASR system confirmed its effectiveness across key modules, including transcription, summarization, text-to-sign language conversion, and quiz generation. The system successfully enhanced accessibility and learning through

transcriptions and analytics. Overall, the results highlight its feasibility and scalability, with opportunities for further refinement.

Chapter 5

Conclusions & Future Scope

Video transcribers have become valuable tools across various industries, including media, education, business, and healthcare, due to their ability to improve accessibility, productivity, and searchability. By converting spoken content into text, they enable easier content indexing, making information retrieval faster and more efficient. In education, for example, transcriptions allow students to review lectures more effectively, while in media, transcribers assist in creating subtitles and enhancing content reach. Businesses benefit from improved meeting documentation, and healthcare applications see greater accuracy in medical documentation and compliance.

Despite their advantages, video transcribers face notable challenges, such as poor audio quality, background noise, and variations in accents and dialects, which can impact transcription accuracy. However, advancements in AI and machine learning continue to improve these tools by enhancing audio recognition and handling diverse speech patterns more effectively. Looking to the future, we can expect video transcribers to support real-time transcription, multilingual capabilities, and even emotion and sentiment analysis. Integration with other tools is also likely, which will broaden their applications and further streamline workflows across multiple domains.

References

[1] W.-T. Sung, H.-W. Kang, and S.-J. Hsiao, "Speech recognition via ctc-cnn model," *TechPress Science, vol. 2023, no.040024*, 2023.

[2] Stoll and Stephanie, "Sign language production using neural machine translation and generative adversarial networks," *International Journal of Computer Vision 128.4*, 2020.

[3] A. M. Sarhan, N. M. Elshennawy, and D. M. Ibrahim, "Hlr-net: a hybrid lip-reading model based on deep convolutional neural networks," *Computers, Materials and Continua, vol. 68, no. 2, pp. 1531–1549*, 2021.

[4] S. S. Alrumiah and A. A. Al-Shargabi, "Educational videos subtitles' summarization using latent dirichlet allocation and length enhancement," *Proceedings of the AAAI Conference on Artificial Intelligence. Vol. 34. No. 05.*, 2022.

Appendix A: Presentation

VIDEO TRANSCRIBER FOR HEARING IMPAIRED

Final Internal Evaluation Presentation

Daryl Antony Luiz
Eshaan Kolloth
Melissa Biju Kalayil
Milin Chandrakumar

Guided by: Mrs. Mehbooba P Shareef

CONTENTS

PROBLEM DEFINITION

2

- The main issue addressed by this project is the challenge individuals face with lengthy educational videos, which leads to wasted time and reduced concentration.
- Learners often struggle to maintain focus, resulting in inefficient learning.
- This problem is especially acute for people with disabilities, as they face additional barriers in accessing and engaging with multimedia contents such as videos.

PURPOSE AND NEED

3

- The Video Transcriber project is designed to address the need for an accessible and efficient learning tool.
- Its purpose is to provide a system that automatically transcribes video content into text with timestamps, summarizes the content, and generates quizzes.
- This tool will help users, especially those with hearing impairments, to save time, improve focus, and enhance their learning experience through increased accessibility to multimedia content.

PROJECT OBJECTIVE

4

- The objective of the Video Transcriber project is to create an innovative platform that enables the automatic transcription of video and audio files into text.
- It will also include additional features such as text summarization, quiz generation, and synchronization of video with transcribed text.
- The platform aims to provide an inclusive and user-friendly solution that enhances learning efficiency and makes video content more accessible to all users.

LITERATURE SURVEY

5

Sl	Paper	Methodology	
1	Speech Recognition via CTC-CN model	End to end speech recognition CTC model combined with CNN to handle spatial features in audio data	
2	Sign Language Production using Neural Machine Translation and Generative Adversarial Networks	Convert text to sign sentence, encode each word, pass to decoder, build MG to make poses and convert to video	

LITERATURE SURVEY

6

Sl	Paper	Methodology
3	Educational Videos Subtitles' Summarization Using Latent Dirichlet Allocation and Length Enhancement	Extractive text summarization process, LDA-based subtitle summarization model framework, Sample of a subtitle file portion in the EDUVSUM dataset and the preprocessed version
4	HLR-Net: A Hybrid Lip-Reading Model	Combination of Inception Layers, Gradient Preservation, Bi-GRU,and Attention Mechanisms

PROPOSED METHOD

7

- **Automatic Speech Recognition**
  - **Accept the video and process the video.**
  - **Extract audio from video**
  - **Convert the audio to text**
  - **Based on audio and text, identify speaker and segment video with time stamp.**

PROPOSED METHOD

8

- Text Summarization and Text to Sign Language
  - Translate the transcript text into English
  - Tokenize the text and extract keywords by removing stopwords
  - Based on extracted keywords and word count, summarize the text
  - Provide accessibility features like transcript editing tools, text to speech, searching and filtering etc.
  - For each word, fetch sign language image or video from dataset or url and make them into one.

PROPOSED METHOD

9

- Text segmentation and MCQ Generation
  - Question and answer generation
  - Question format selection
  - User analytics and quiz analytics
  - Text Segmentation.

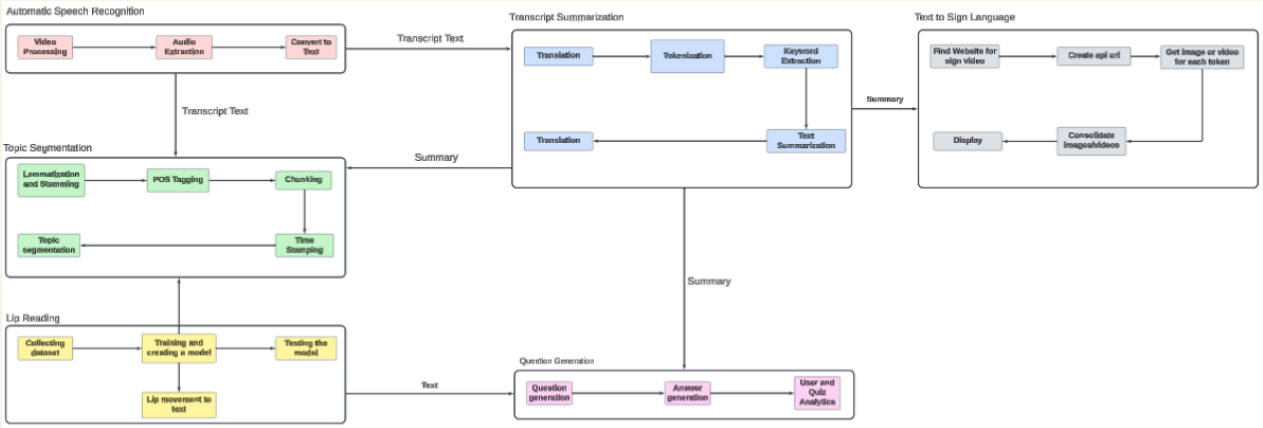
PROPOSED METHOD

10

- Lip Reading
  - Collect dataset for lip reading
  - Divide it into training and test set
  - Train and create a model using training dataset
  - Test the model using test dataset
  - Lip movement to text conversion
  - Use the text to segment video based on topic with time stamping.

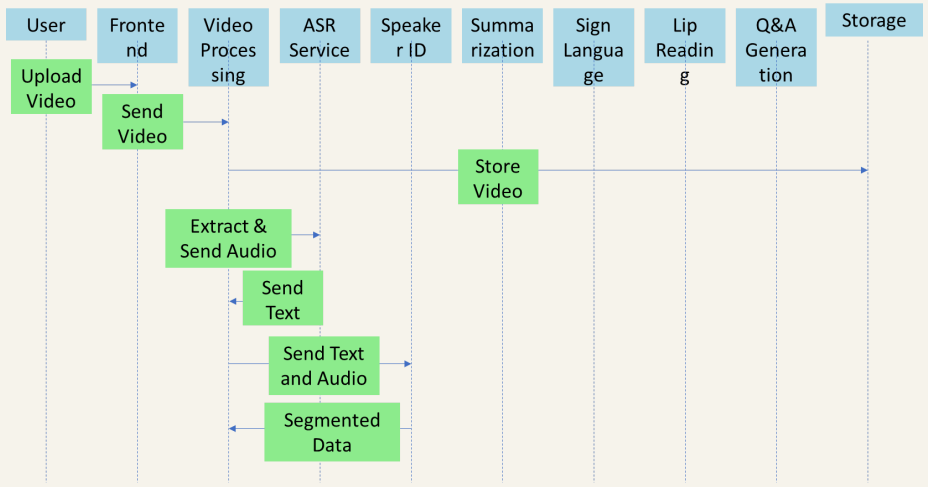
ARCHITECTURE DIAGRAM

11



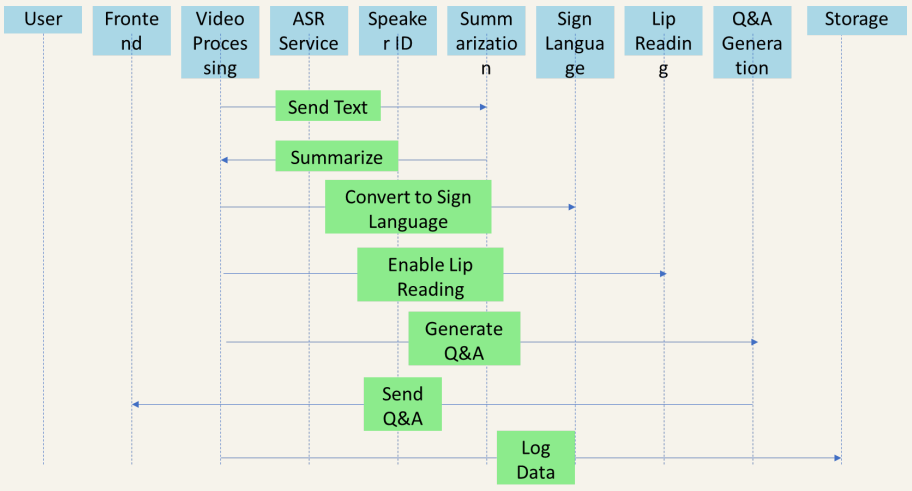
SEQUENCE DIAGRAM

12



SEQUENCE DIAGRAM

13



MODULES

14

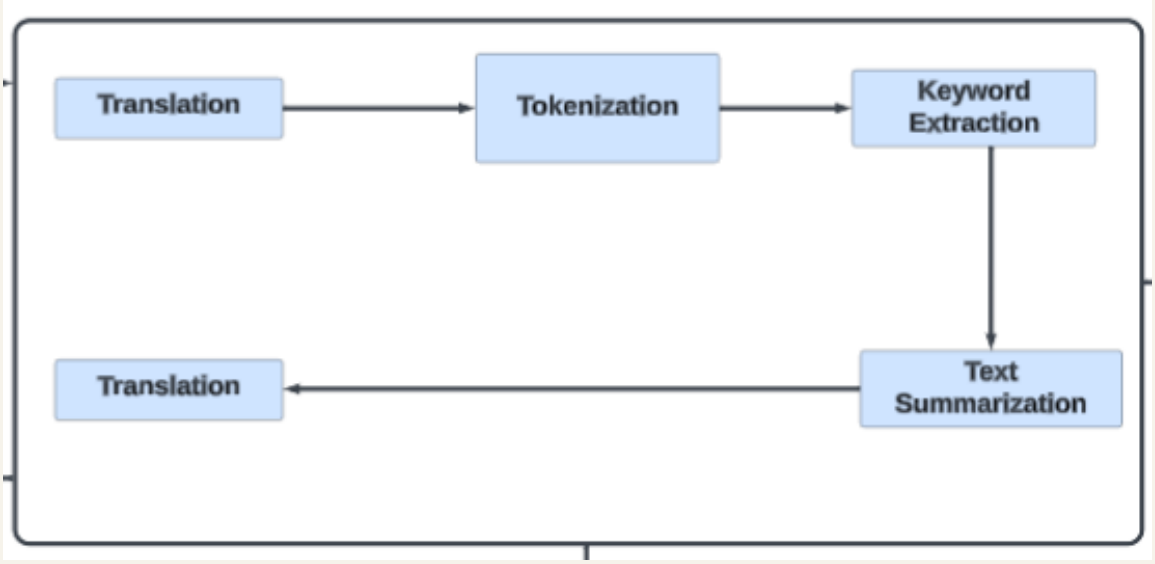
- Automatic Speech Recognition
- Transcript summarization and Text to Sign Language
- Topic Segmentation and MCQ Generation
- Lip Reading
- Web platform

ASR MODULE

15

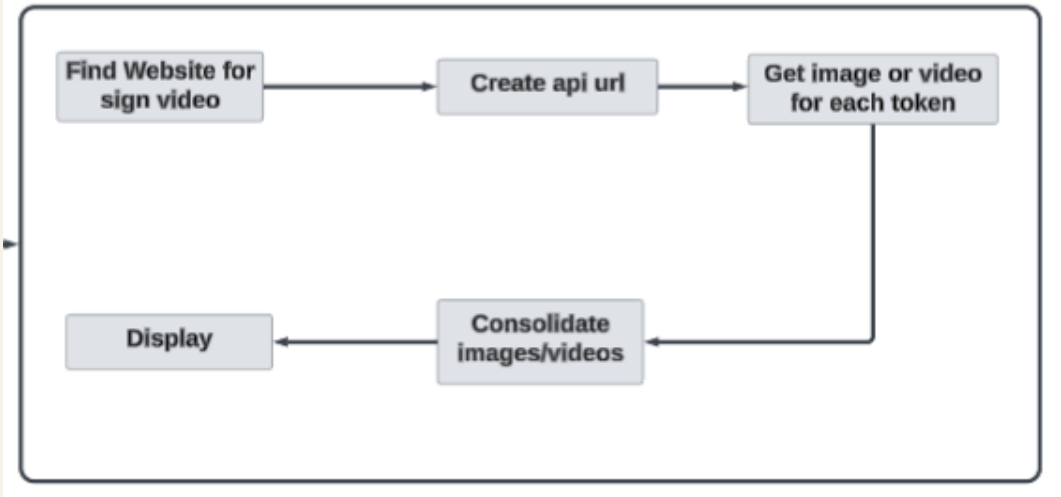
TRANSCRIPT SUMMARIZATION MODULE

16



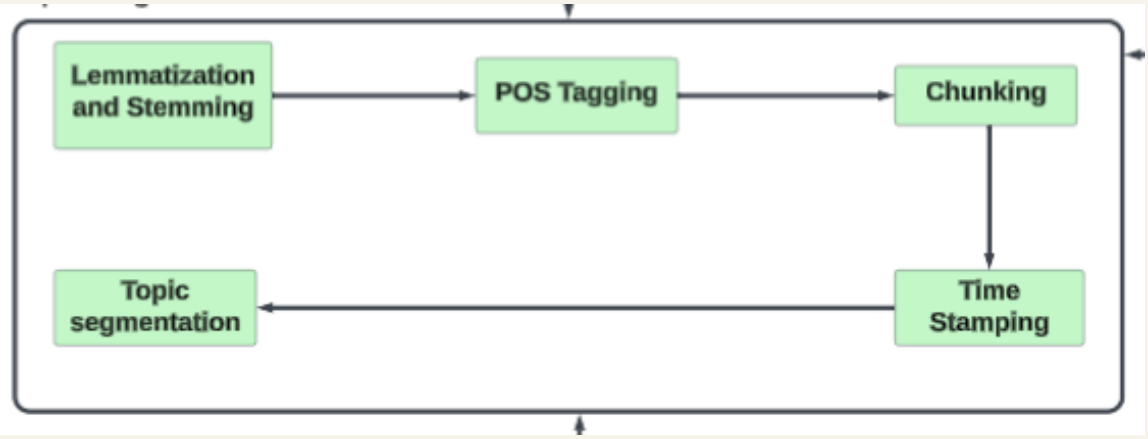
TEXT TO SIGN LANGUAGE MODULE

17



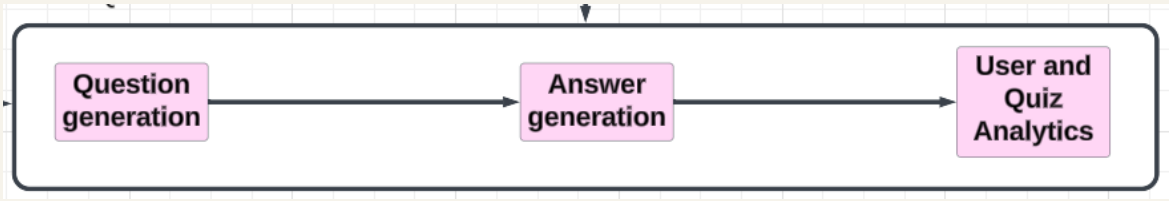
TOPIC SEGMENTATION MODULE

18

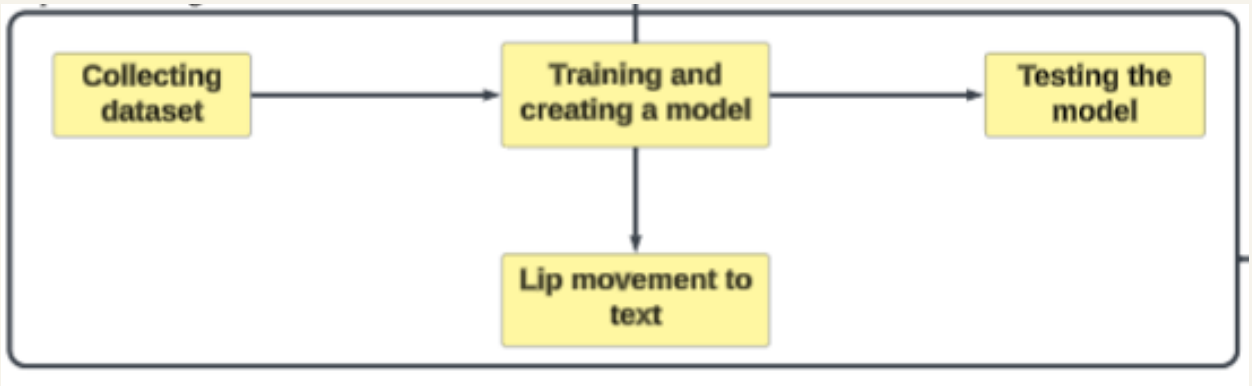


QUESTION GENERATION MODULE

19



LIP READING MODULE



WORK BREAKDOWN

- Daryl
  - Translation
  - Identifying and extracting keywords
  - Summarizing large transcript and highlighting key information
  - Dynamic user transcript editing
  - Searching and filtering
  - Text to speech
  - Text to sign language.
  - Web development and designing

WORK BREAKDOWN AND RESPONSIBILITIES

22

- Melissa
  - Processing uploaded video
  - Extracting audio from video
  - Speech to text conversion
  - Time stamping
  - Topic segmentation
  - Speaker identification
  - Web development and designing

WORK BREAKDOWN AND RESPONSIBILITIES

23

- Eshaan
  - Collecting data for lip reading
  - Train and create a model for lip reading
  - Lip reading to text conversion
  - Topic segmentation
  - Speaker identification
  - Time stamping
  - Web designing ideas

WORK BREAKDOWN AND RESPONSIBILITIES

- Milin
  - Question and answer generation
  - Question format selection
  - User analytics and quiz analytics.
  - Summarization
  - Topic segmentation
  - Web designing ideas

RESULTS

VIDEO TO AUDIO

26

```
Requirement already satisfied: moviepy in /usr/local/lib/python3.11/dist-packages (1.0.3)
Requirement already satisfied: decorator<5.0,>=4.0.2 in /usr/local/lib/python3.11/dist-packages (from moviepy) (4.4.2)
Requirement already satisfied: imageio<3.0,>=2.5 in /usr/local/lib/python3.11/dist-packages (from moviepy) (2.37.0)
Requirement already satisfied: imageio_ffmpeg>=0.2.0 in /usr/local/lib/python3.11/dist-packages (from moviepy) (0.6.0)
Requirement already satisfied: tqdm<5.0,>=4.11.2 in /usr/local/lib/python3.11/dist-packages (from moviepy) (4.67.1)
Requirement already satisfied: numpy>=1.17.3 in /usr/local/lib/python3.11/dist-packages (from moviepy) (2.0.2)
Requirement already satisfied: requests<3.0,>=2.8.1 in /usr/local/lib/python3.11/dist-packages (from moviepy) (2.32.3)
Requirement already satisfied: proglog<=1.0.0 in /usr/local/lib/python3.11/dist-packages (from moviepy) (0.1.10)
Requirement already satisfied: pillow>=8.3.2 in /usr/local/lib/python3.11/dist-packages (from imageio<3.0,>=2.5->moviepy) (11.1.0)
Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python3.11/dist-packages (from requests<3.0,>=2.8.1->moviepy) (3.4.1)
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.11/dist-packages (from requests<3.0,>=2.8.1->moviepy) (3.10)
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.11/dist-packages (from requests<3.0,>=2.8.1->moviepy) (2.3.0)
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.11/dist-packages (from requests<3.0,>=2.8.1->moviepy) (2025.1.31)
WARNING:py.warnings:/usr/local/lib/python3.11/dist-packages/moviepy/video/io/sliders.py:61: SyntaxWarning: "is" with a literal. Did you mean "=="?
  if event.key is 'enter':

Choose Files  Example_ 3...ssional.mp4
• Example_ 30-second pitch from an HR professional.mp4(video/mp4) - 4110387 bytes, last modified: 07/11/2024 - 100% done
Saving Example_ 30-second pitch from an HR professional.mp4 to Example_ 30-second pitch from an HR professional.mp4
MoviePy - Writing audio in downloaded_audio.mp3
                                                                    MoviePy - Done.
```

VIDEO URL TO AUDIO

27

```
Collecting yt-dlp
  Downloading yt_dlp-2025.3.21-py3-none-any.whl.metadata (172 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 172.1/172.1 kB 3.3 MB/s eta 0:00:00
Downloading yt_dlp-2025.3.21-py3-none-any.whl (3.2 MB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 3.2/3.2 MB 22.2 MB/s eta 0:00:00
Installing collected packages: yt-dlp
Successfully installed yt-dlp-2025.3.21
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
ffmpeg is already the newest version (7:4.4.2-0ubuntu0.22.04.1).
0 upgraded, 0 newly installed, 0 to remove and 29 not upgraded.
Enter the YouTube video URL: https://www.youtube.com/watch?v=YcOdGAnJlOc
[youtube] Extracting URL: https://www.youtube.com/watch?v=YcOdGAnJlOc
[youtube] YcOdGAnJlOc: Downloading webpage
[youtube] YcOdGAnJlOc: Downloading tv client config
[youtube] YcOdGAnJlOc: Downloading player 69f581a5
[youtube] YcOdGAnJlOc: Downloading tv player API JSON
[youtube] YcOdGAnJlOc: Downloading ios player API JSON
[youtube] YcOdGAnJlOc: Downloading m3u8 information
[info] YcOdGAnJlOc: Downloading 1 format(s): 251
[download] Destination: downloaded_audio.webm
[download] 100% of    1.33MiB in 00:00:00 at 3.87MiB/s
[ExtractAudio] Destination: downloaded_audio.mp3
Deleting original file downloaded_audio.webm (pass -k to keep)
Audio downloaded and saved as MP3.
Audio file is ready for download.
```

AUDIO TO TRANSCRIBED TEXT

28

```
Requirement already satisfied: SpeechRecognition in /usr/local/lib/python3.11/dist-packages (3.14.1)
Requirement already satisfied: pydub in /usr/local/lib/python3.11/dist-packages (0.25.1)
Requirement already satisfied: typing-extensions in /usr/local/lib/python3.11/dist-packages (from SpeechRecognition) (4.12.2)
Transcribed Text:
supervised and unsupervised learning what's the difference supervised learning is when we train the model with labeled data we tell the model this is what the r
```

TOPIC SEGMENTATION AND TIME STAMPING

30

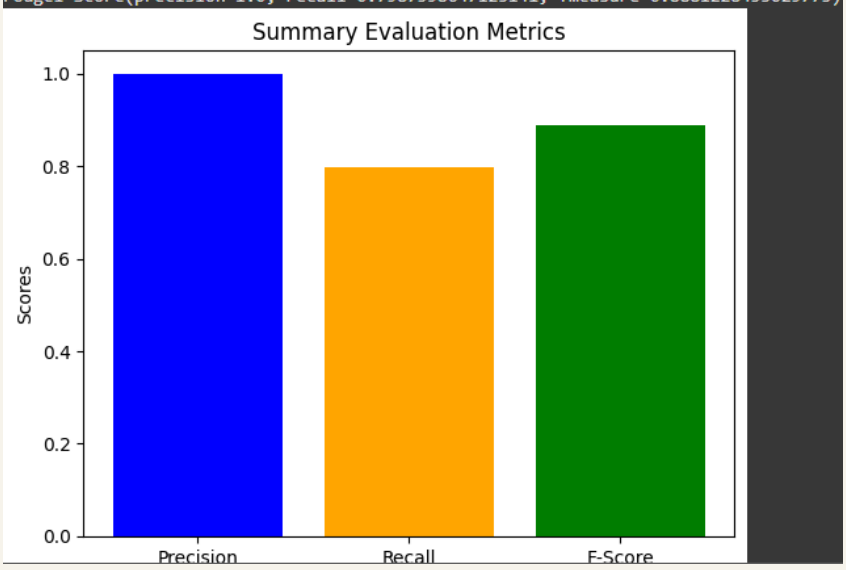
```
=== Topic 0 ===
[0.00s - 5.20s] Supervised Unsupervised

=== Topic 1 ===
[5.20s - 10.40s] Info Unsupervised

=== Topic 2 ===
[10.40s - 15.60s] Model Unlabeled

=== Topic 3 ===
[15.60s - 20.80s] Use Type

=== Topic 4 ===
[20.80s - 26.00s] Purchases History

=== Topic 5 ===
[26.00s - 31.20s] Model Customers

=== Topic 6 ===
[31.20s - 36.40s] Buy Jelly

=== Topic 7 ===
[36.40s - 41.60s] Cart Supervised

=== Topic 8 ===
[41.60s - 46.80s] Shoppers Spend

=== Topic 9 ===
[46.80s - 52.00s] Related Products
```

SUMMARIZATION EVALUATION

```
Summary:

Wind energy captures the kinetic energy of wind to generate power , with onshore and offshore wind farms spreading across many countries .
Wind energy is another essential renewable resource , generating power by capturing the kinetic energy of the wind .
Despite initial setup costs , the operational expenses of renewable energy systems are often lower than traditional energy systems .
Solar and wind energy , in particular , allow for decentralized energy production , strengthening local economies .
Renewable energy can also decentralize power production , enabling communities to have local , reliable sources of energy .
Renewable energy , on the other hand , refers to energy sources that are naturally replenished , such as solar , wind , hydroelectric , geothermal , and bio
Studies show that wind energy can contribute to grid stability when combined with energy storage .
Solar energy is among the most abundant and accessible forms of renewable energy , harnessing power from sunlight .
Community-owned renewable energy projects allow residents to share in the financial benefits of clean energy production .
Some communities have benefited economically from solar energy by leasing land for solar farms or selling surplus energy .
Biomass energy converts organic materials like plants and waste into fuel , creating energy from natural processes .
The transition to renewable energy also decreases reliance on imported fuels , improving energy security .
Many governments are investing in renewable energy infrastructure to meet growing energy demands sustainably .
Countries like Denmark and Germany have become leaders in wind energy , with significant portions of their electricity coming from wind power .
Offshore wind farms are particularly effective because wind speeds tend to be higher over water , increasing energy output .
One challenge with wind energy is its variability , as electricity production depends on wind speed , which can fluctuate .
Hydropower plants convert the potential energy of water stored in dams into electricity by releasing water to turn turbines .
The renewable energy sector is labor-intensive , particularly in the manufacturing , installation , and maintenance of solar panels , wind turbines , and hy
To address this , hybrid systems combining wind and solar power are being developed to provide more consistent energy output .
Battery storage and grid improvements are crucial for integrating wind energy into the electricity grid smoothly .
As the cost of wind energy decreases , it has become one of the fastest-growing sources of electricity worldwide .
Some regions have found innovative uses for wind energy , like using excess power to produce hydrogen fuel through electrolysis .
Moreover , solar power production varies with weather and daylight , requiring complementary energy storage systems .
Solar energy harnesses the sun ' s power , providing a clean and inexhaustible source of electricity and heat .
```


EXTRACTIVE SUMMARIZATION ROUGE SCORE

```
rouge1 Score(precision=1.0, recall=0.7987598647125141, fmeasure=0.8881228455029773)
```

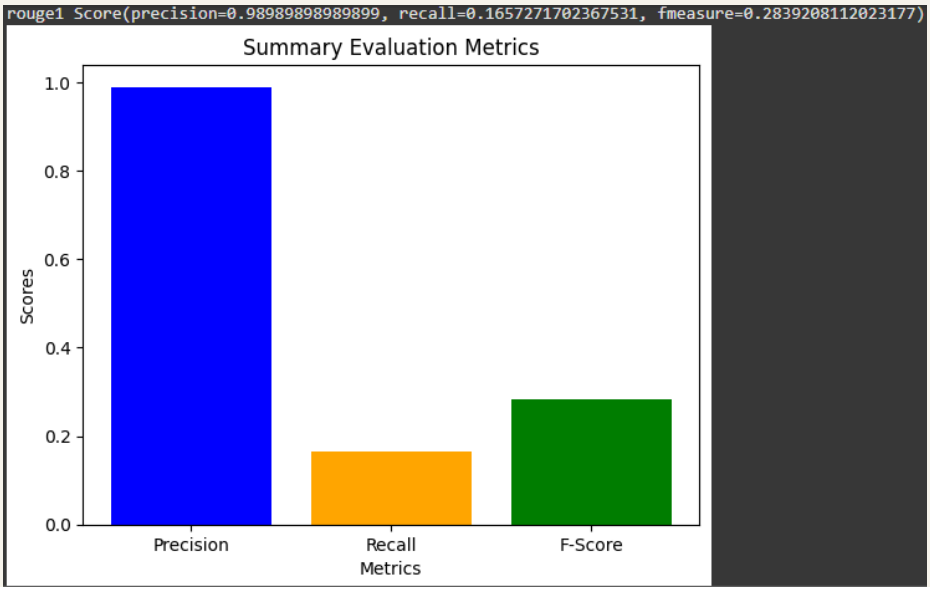


SUMMARIZATION EVALUATION

```
Summary:

Renewable energy is becoming increasingly important as the world faces environmental challenges and finite fossil fuel supplies.
 Fossil fuels, which include coal, oil, and natural gas, are non-renewable resources.
 Renewable energy, on the other hand, refers to energy sources that are naturally replenished.
 These renewable sources help reducecarbon emissions and are essential in the fight against climate change.


Technological advancements have significantlyreduced the cost of solar power, making it more accessible to individuals and businesses alike.
 Rooftop solar installations allow homeowners to generate their own power, often producing excess energy that can be sold back to the grid.
 Large-scale solar farms, often located in sunny, open areas, generate massive amounts of electricity for cities.


Wind energy is emission-free during operation, making it an environmentally friendly option.
 Offshore wind farms are particularly effective because windspeeds tend to be higher over water, increasing energy output.
 Wind farms can also be used in combination with agriculture, with turbines spaced far enough apart to allow for farming activities.
 Battery storage and grid improvements are crucial for integrating wind energy into the electricity grid smoothly.


Hydropower plants convert potential energy of water stored in dams into electricity.
 This form of energy is highly reliable, as river flow is generally consistent.
 Large-scale hydropower projects can produce electricity for millions, but smaller, run-of-the-river systems are also common.
 These smaller systems have a lower environmental impact, as they do not require large dams.


The adoption of renewable energy sources also positively impacts economic growth and job creation.
 The renewable energy sector is labor-intensive,particularly in the manufacturing, installation, and maintenance of solar panels, wind turbi
```


ABSTRACTIVE SUMMARIZATION ROUGE SCORE

```
rouge1 Score(precision=0.98989898989899, recall=0.1657271702367531, fmeasure=0.2839208112023177)
```



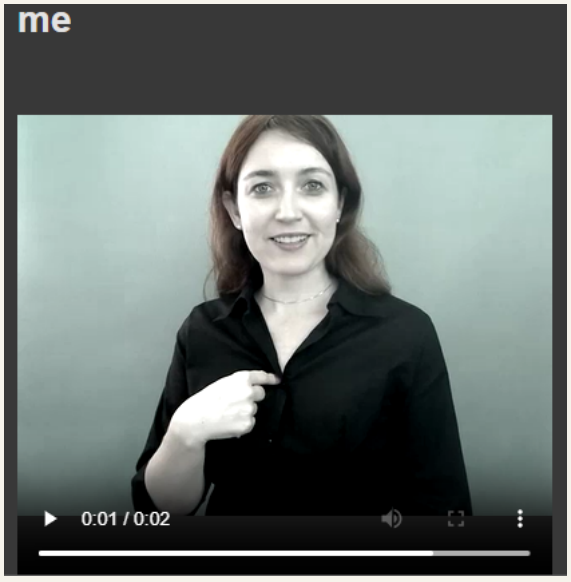
35

KEY INFORMATION HIGHLIGHTS

```
Enter keyword:energy
['Wind energy captures the kinetic energy of wind to generate power , with onshore and offshore wind farms spreading across many countries .']
['Wind energy is another essential renewable resource , generating power by capturing the kinetic energy of the wind .']
['Despite initial setup costs , the operational expenses of renewable energy systems are often lower than traditional energy systems .']
['Solar and wind energy , in particular , allow for decentralized energy production , strengthening local economies .']
['Renewable energy can also decentralize power production , enabling communities to have local , reliable sources of energy .']
['Renewable energy , on the other hand , refers to energy sources that are naturally replenished , such as solar , wind , hydroelectric , geotherm
['Studies show that wind energy can contribute to grid stability when combined with energy storage .']
['Solar energy is among the most abundant and accessible forms of renewable energy , harnessing power from sunlight .']
['Community-owned renewable energy projects allow residents to share in the financial benefits of clean energy production .']
['Some communities have benefited economically from solar energy by leasing land for solar farms or selling surplus energy .']
['Biomass energy converts organic materials like plants and waste into fuel , creating energy from natural processes .']
['The transition to renewable energy also decreases reliance on imported fuels , improving energy security .']
['Many governments are investing in renewable energy infrastructure to meet growing energy demands sustainably .']
['Countries like Denmark and Germany have become leaders in wind energy , with significant portions of their electricity coming from wind power .'
['Offshore wind farms are particularly effective because wind speeds tend to be higher over water , increasing energy output .']
['One challenge with wind energy is its variability , as electricity production depends on wind speed , which can fluctuate .']
['Hydropower plants convert the potential energy of water stored in dams into electricity by releasing water to turn turbines .']
['The renewable energy sector is labor-intensive , particularly in the manufacturing , installation , and maintenance of solar panels , wind turbi
['To address this , hybrid systems combining wind and solar power are being developed to provide more consistent energy output .']
['Battery storage and grid improvements are crucial for integrating wind energy into the electricity grid smoothly .']
['As the cost of wind energy decreases , it has become one of the fastest-growing sources of electricity worldwide .']
['Some regions have found innovative uses for wind energy , like using excess power to produce hydrogen fuel through electrolysis .']
['Moreover , solar power production varies with weather and daylight , requiring complementary energy storage systems .']
['Solar energy harnesses the sun ' s power , providing a clean and inexhaustible source of electricity and heat .']
['Small businesses can benefit from lower energy costs by installing their own solar or wind systems .']
['Pumped-storage hydropower is a type of hydropower used to store excess energy from other sources by pumping water uphill during low demand .']
```

36

TEXT TO SIGN LANGUAGE OUTPUT



37

TEXT TO SIGN LANGUAGE OUTPUT

38

LIP READING MODEL

```
[ ] model.summary()
```

Model: "sequential"

Layer (type)	Output Shape	Param #
conv3d (Conv3D)	(None, 75, 46, 140, 128)	3,584
activation (Activation)	(None, 75, 46, 140, 128)	0
max_pooling3d (MaxPooling3D)	(None, 75, 23, 70, 128)	0
conv3d_1 (Conv3D)	(None, 75, 23, 70, 256)	884,992
activation_1 (Activation)	(None, 75, 23, 70, 256)	0
max_pooling3d_1 (MaxPooling3D)	(None, 75, 11, 35, 256)	0
conv3d_2 (Conv3D)	(None, 75, 11, 35, 75)	518,475
activation_2 (Activation)	(None, 75, 11, 35, 75)	0
max_pooling3d_2 (MaxPooling3D)	(None, 75, 5, 17, 75)	0
time_distributed (TimeDistributed)	(None, 75, 6375)	0
bidirectional (Bidirectional)	(None, 75, 256)	6,660,096
dropout (Dropout)	(None, 75, 256)	0
bidirectional_1 (Bidirectional)	(None, 75, 256)	394,240
dropout_1 (Dropout)	(None, 75, 256)	0
dense (Dense)	(None, 75, 41)	10,537

Total params: 8,471,924 (32.32 MB)
Trainable params: 8,471,924 (32.32 MB)
Non-trainable params: 0 (0.00 B)

LIP READING OUTPUT

```
sample = load_data(tf.convert_to_tensor('./data/s1/lwbl8p.mpg'))
print('~'*100, 'REAL TEXT')
#predicted=[tf.strings.reduce_join([num_to_char(word) for word in sentence]) for sentence in sample[1]]
[tf.strings.reduce_join([num_to_char(word) for word in sentence]) for sentence in [sample[1]]]
```

```
alignment =  tf.Tensor(
[39 12  1 25 39 23  8  9 20  5 39 19 16 39  2 25 39 12 39  5  9  7  8 20
 39 16 12  5  1 19  5], shape=(31,), dtype=int64)
data/s1/lwbl8p.mpg
data/alignments/s1/lwbl8p.align
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~ REAL TEXT
[<tf.Tensor: shape=(), dtype=string, numpy=b' lay white sp by l eight please'>]
```


LIP READING MODULE ACCURACY

```
import jiwer
import Levenshtein
from nltk.translate.bleu_score import sentence_bleu

# Compute WER (Word Error Rate)
wer = jiwer.wer(expected_text, predicted_text)

# Compute CER (Character Error Rate)
cer = Levenshtein.distance(expected_text, predicted_text) / len(expected_text)

# Compute BLEU score
bleu_score = sentence_bleu([expected_text.split()], predicted_text.split())

# Display results
print(f"Word Error Rate (WER): {wer:.4f}")
print(f"Character Error Rate (CER): {cer:.4f}")
print(f"BLEU Score: {bleu_score:.4f}")
```

```
Word Error Rate (WER): 0.0000
Character Error Rate (CER): 0.0000
BLEU Score: 1.0000
```

QUESTION EVALUATION

```
Choose the type of questions you want to answer:
1. Fill in the Blanks
2. Multiple Choice Questions
3. Multiple Blanks
4. Exit
Enter your choice (1-4): 1
Enter the number of questions to generate: 2

Fill in the Blanks:

Question:  supervised learning is when we train the model with labeled _________ we tell the model this is what the right output should be based on
Enter your answer: data
Correct!

Question: each has its place and choosing the right one depends on the _________ that you are trying to solve.
Enter your answer: solution
Incorrect. The correct answer is: Problem
```

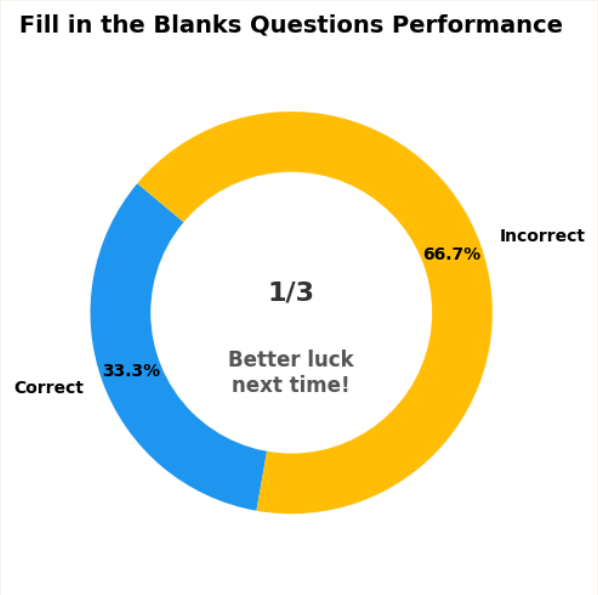

QUESTION EVALUATION

```
Choose the type of questions you want to answer:
1. Fill in the Blanks
2. Multiple Choice Questions
3. Multiple Blanks
4. Exit
Enter your choice (1-4): 2
Enter the number of questions to generate: 2

Multiple Choice Questions:

Question:  supervised learning is when we train the _________ with labeled data we tell the _________ this is what the right output should be ba
1. Output
2. Data
3. Model
4. Problem
Enter the option number: 3
Correct!

Question:  supervised learning is when we train the model with labeled data we tell the model this is what the right _________ should be based on
1. Output
2. Data
3. Place
4. Info
Enter the option number: 3
Incorrect. The correct answer is: Output
```

QUESTION EVALUATION

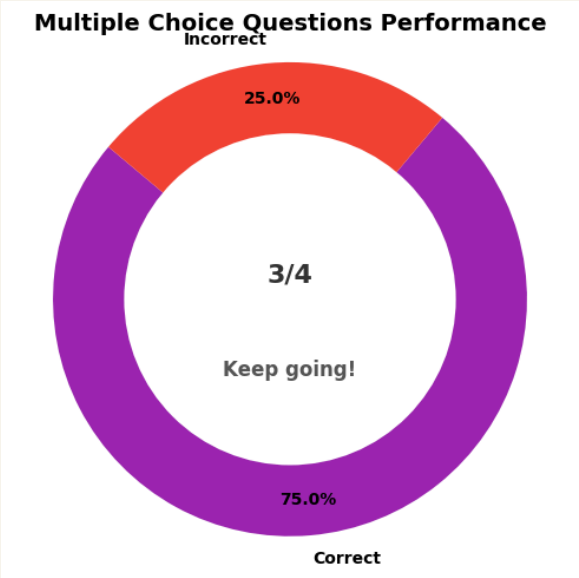

```
 supervised learning is when we train the _________ with labeled data we tell the model this is what the right output sho
 supervised learning is when we train the _________ with labeled data we tell the model this is what the right output should be based on this _________ unsupe
Question:  supervised learning is when we train the _________ with labeled data we tell the model this is what the right output should be based on this _____
Enter first answer: info
Enter second answer: model
Incorrect. The correct answers are: Model, Info

 supervised learning is when we train the model with labeled data we tell the model this is what the right _________ should be based on this info unsupervised
 supervised learning is when we train the model with labeled data we tell the model this is what the right _________ should be based on this info unsupervised
Question:  supervised learning is when we train the model with labeled data we tell the model this is what the right _________ should be based on this info uns
Enter first answer: output
Enter second answer: patterns
Correct!
```

QUESTION EVALUATION



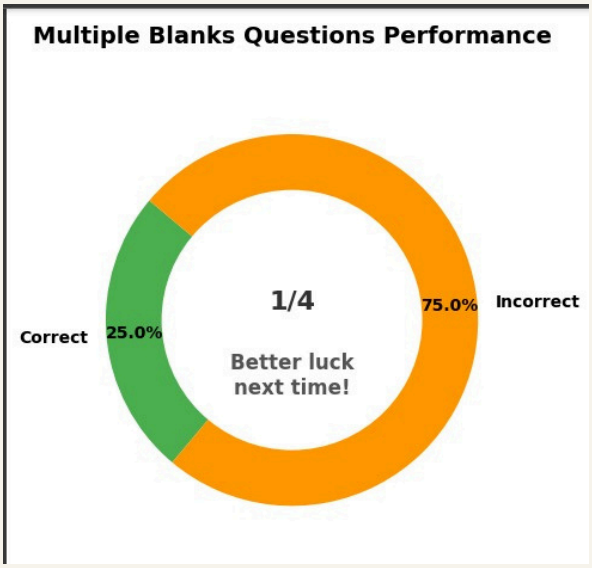
45

QUESTION EVALUATION



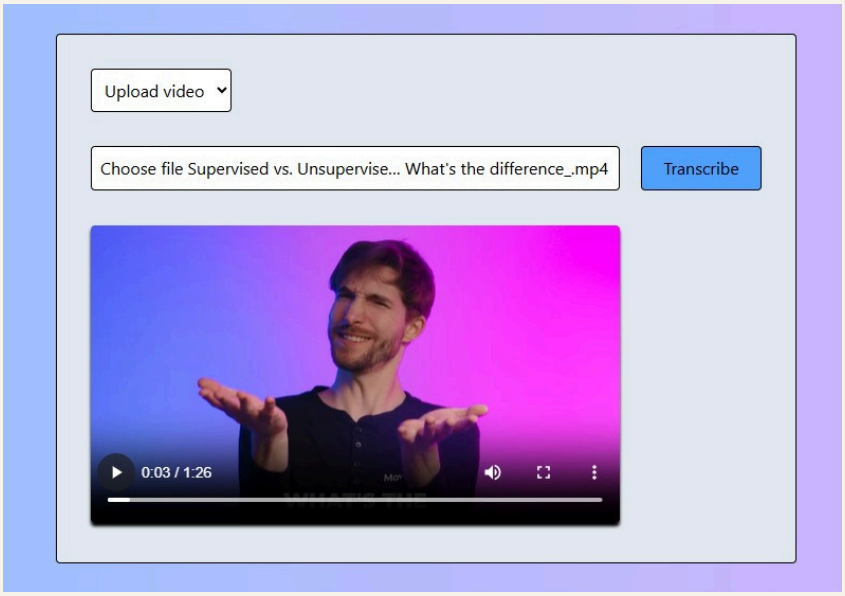
46

QUESTION EVALUATION



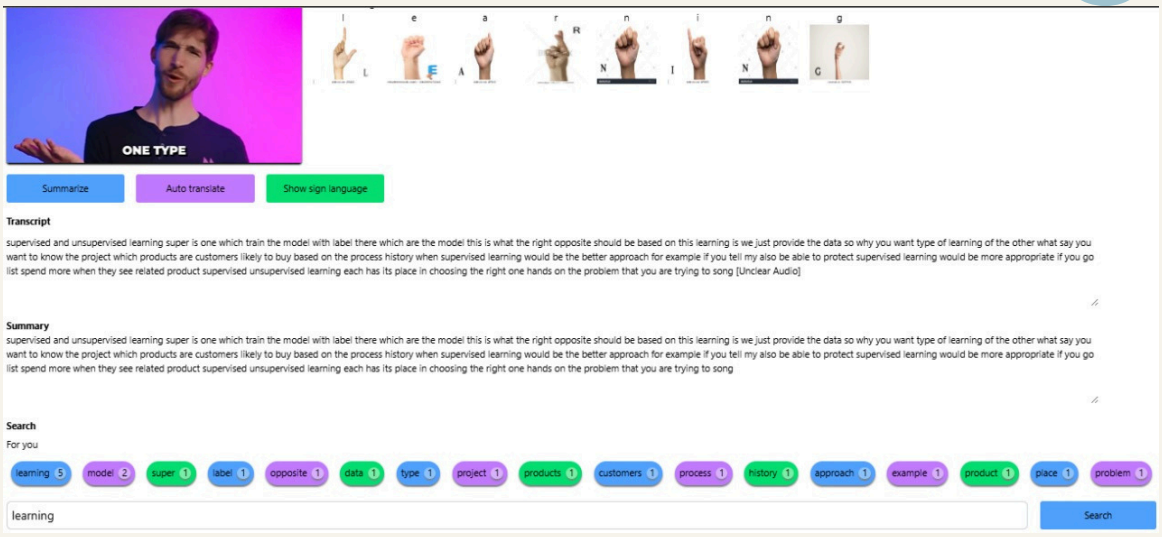
WEBSITE PREVIEW PAGE

47



WEBSITE PREVIEW PAGE

48



HARDWARE REQUIREMENTS

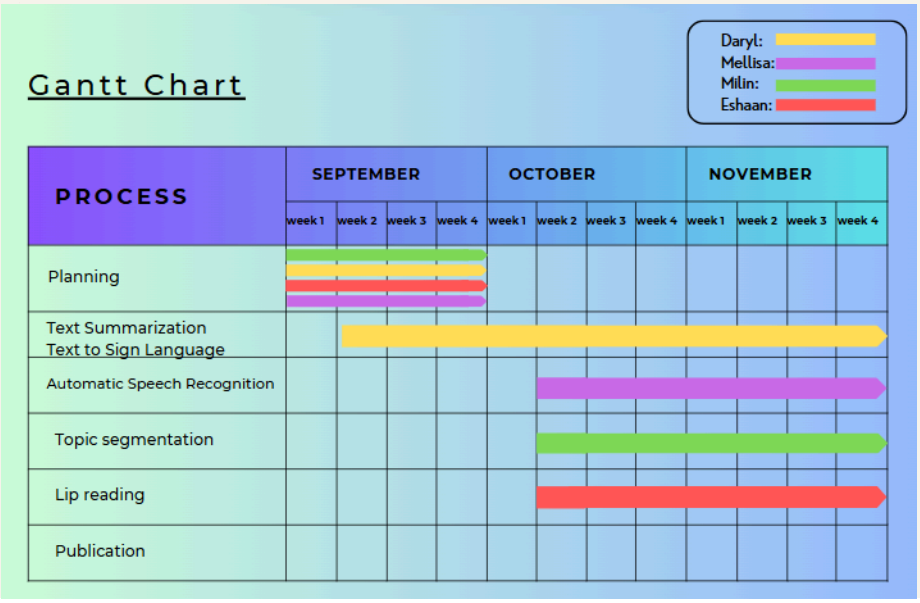
- CPU: 4-8 cores (e.g., Intel Core i7 or AMD Ryzen 7)
- RAM: 16-32 GB
- Storage: 512 GB SSD

SOFTWARE REQUIREMENTS

- Operating System: Linux based system (like Ubuntu etc.) or Windows.
- Programming Languages: Python, HTML, JS, CSS.
- Web Framework: React, Flask or Django
- Libraries: Ntlk, Googletrans, Pytube, movie.py, Speech_recognition

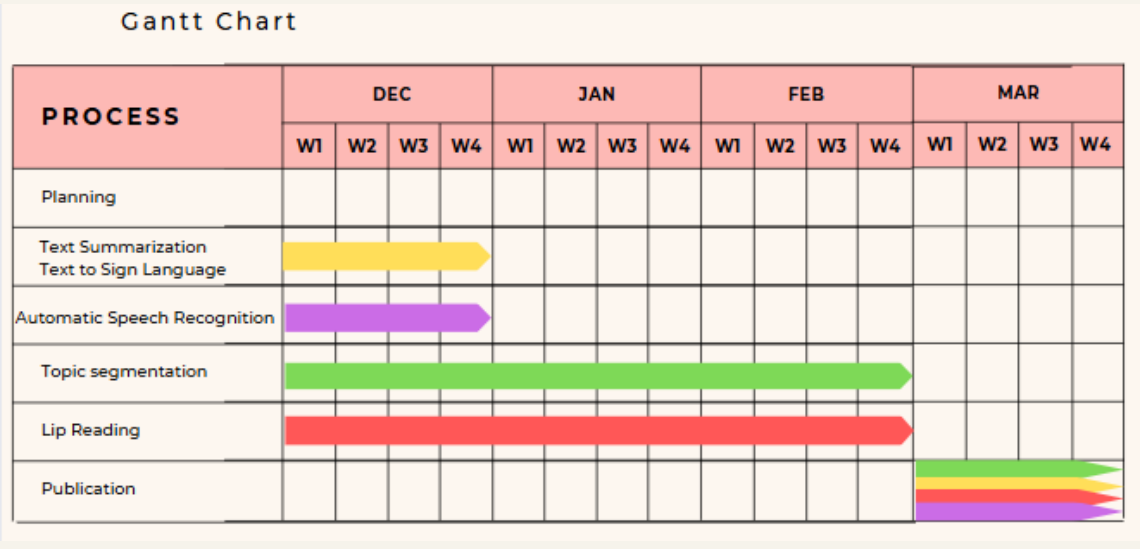
GANTT CHART

51



GANTT CHART

52



RISK AND CHALLANGES

53

- Data Privacy Concerns: Users may be hesitant to upload personal or sensitive videos due to privacy issues.
- Inaccurate Transcription: Errors in transcription can lead to misleading summaries or quiz questions, affecting the application's credibility.
- Dependence on External APIs: The project might rely on external APIs for speech recognition, translation, or sign language conversion, which could introduce latency, API limits, or sudden unavailability.

RISK AND CHALLANGES

54

- Real-time Speaker Identification: Identifying multiple speakers accurately in real-time is challenging, especially in overlapping conversations.
- Effective Summarization: Creating meaningful and contextually accurate summaries without losing key points from long video content.
- Quiz Generation: Automatically generating meaningful quiz questions from transcribed text without manual intervention.

FUTURE IMPROVEMENTS

55

- Enhanced ASR Accuracy – Improve transcription precision by integrating advanced speech recognition models like OpenAI's Whisper or DeepSpeech, reducing errors in noisy environments and handling multiple accents.
- Improved Lip Reading Capabilities – Upgrade the lipreading model using transformer-based architectures like Vision Transformers to enhance the recognition of complex facial movements and improve accuracy in challenging conditions.
- Advanced Summarization and Topic Segmentation – Utilize state-of-the-art NLP models such as PEGASUS and BART for more natural and multilingual summarization, while refining topic segmentation with BERT-based methods for smarter content organization.

FUTURE IMPROVEMENTS

56

- AI-Powered Sign Language Translation – Expand the sign language dataset, incorporate AI-driven 3D avatar-based sign animations, and ensure seamless accessibility for the hearing-impaired community.
- Real-Time Processing and Streaming – Enable live transcription, real-time lipreading, and sign language translation with reduced latency, making the system effective for video calls, lectures, and online meetings.
- Cloud Integration and Smart Device Compatibility – Implement cloud-based storage, collaborative editing tools, and AI assistant integration, along with smart device and AR/VR support, for a more immersive and user-friendly experience.

CONCLUSION

- The system will automatically transcribe video and audio content into text along with time-stamped text.
- Will generate concise summaries of the transcribed text by summarizing main ideas and key points.
- Will create questions based on the transcribed and summarized text.
- Will improve accessibility for individuals with visual or hearing impairments By providing text transcriptions of video/audio content and descriptive features.
- Will provide analytics on user behavior and quiz performance.

REFERENCES

[1]Sung, Wen-Tsai, Hao-Wei Kang, and Sung-Jung Hsiao. "Speech Recognition via CTC-CNN Model." (2022). https://www.techscience.com/cmc/v76n3/54353

[2]Stoll, Stephanie, et al. "Text2Sign: towards sign language production using neural machine translation and generative adversarial networks." International Journal of Computer Vision 128.4 (2020): 891-908. https://link.springer.com/content/pdf/10.1007/s11263-019-01281-2.pdf

[3]Alrumiah, Sarah S., and Amal A. Al-Shargabi. "Educational Videos Subtitles' Summarization Using Latent Dirichlet Allocation and Length Enhancement." Computers, Materials & Continua 70.3 (2022). https://cdn.techscience.cn/ueditor/files/cmc/TSP_CMC-70-3/TSP_CMC_21780/TSP_CMC_21780.pdf

[4]Sarhan, Amany M., Nada M. Elshennawy, and Dina M. Ibrahim. "HLR-net: a hybrid lip-reading model based on deep convolutional neural networks." Computers, Materials and Continua 68.2 (2021): 1531-49. https://cdn.techscience.cn/ueditor/files/cmc/TSP-CMC-68-2/TSP_CMC_16509/TSP_CMC_16509.pdf

THANK YOU

Appendix B: Vision, Mission, Programme Outcomes and Course Outcomes

Vision, Mission, Programme Outcomes and Course Outcomes

Vision

To become a Centre of Excellence in Computer Science Engineering, moulding professionals catering to the research and professional needs of national and international organizations.

Mission

To inspire and nurture students, with up-to-date knowledge in Computer Science Engineering, Ethics, Team Spirit, Leadership Abilities, Innovation and Creativity to come out with solutions meeting the societal needs.

Programme Outcomes (PO)

Engineering Graduates will be able to:

PO1: Engineering Knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

PO 2. Problem analysis: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

PO 3. Design/development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

PO 4. Conduct investigations of complex problems: Use research-based knowledge including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

PO 5. Modern Tool Usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

PO 6. The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal, and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

PO 7. Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

PO 8. Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

PO 9. Individual and Team work: Function effectively as an individual, and as a member or leader in teams, and in multidisciplinary settings.

PO 10. Communication: Communicate effectively with the engineering community and with society at large. Be able to comprehend and write effective reports documentation. Make effective presentations, and give and receive clear instructions.

PO 11. Project management and finance: Demonstrate knowledge and understanding of engineering and management principles and apply these to one's own work, as a member and leader in a team. Manage projects in multidisciplinary environments.

PO 12. Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and lifelong learning in the broadest context of technological change.

Programme Specific Outcomes (PSO)

A graduate of the Computer Science and Engineering Program will demonstrate:

PSO1: Computer Science Specific Skills
The ability to identify, analyze and design solutions for complex engineering problems in multidisciplinary areas by understanding the core principles and concepts of computer science and thereby engage in national grand challenges.

PSO2: Programming and Software Development Skills
The ability to acquire programming efficiency by designing algorithms and applying standard practices in software project development to deliver quality software products meet-

ing the demands of the industry.

PSO3: Professional Skills
The ability to apply the fundamentals of computer science in competitive research and to develop innovative products to meet the societal needs thereby evolving as an eminent researcher and entrepreneur.

Course Outcomes (CO)

CO1: Model and solve real world problems by applying knowledge across domains (Cognitive knowledge level: Apply).

CO 2: Develop products, processes or technologies for sustainable and socially relevant applications (Cognitive knowledge level: Apply).

CO 3: Function effectively as an individual and as a leader in diverse teams and to comprehend and execute designated tasks (Cognitive knowledge level: Apply).

CO 4: Plan and execute tasks utilizing available resources within timelines, following ethical and professional norms (Cognitive knowledge level: Apply).

CO 5: Identify technology/research gaps and propose innovative/creative solutions (Cognitive knowledge level: Analyze).

CO 6: Organize and communicate technical and scientific findings effectively in written and oral forms (Cognitive knowledge level: Apply).

Appendix C: CO-PO-PSO Mapping

Table 5.1: CO-PO AND CO-PSO MAPPING

	PO												PSO		
	1	2	3	4	5	6	7	8	9	10	11	12	1	2	3
CO 1	2	2	2	1	2	2	2	1	1	1	1	2	3		
CO 2	2	2	2		1	3	3	1	1		1	1		2	
CO 3									3	2	2	1			3
CO 4					2			3	2	2	3	2			3
CO 5	2	3	3	1	2							1	3		
CO 6					2			2	2	3	1	1			3

3/2/1: high/medium/low