Closed Caption Aligner

(A tool that analyzes video files and produces independent subtitle files from the closed captions data)

Bachelor of Technology

in

Computer Science and Engineering

by

AAKARSH SRIVASTAWA(112115001)

Under the Supervision of:

Semester:IV

Name of Department:Department of Computer Science and Engineering

Indian Institute of Information And Technology, Pune

(An Institute of National Importance by an Act of Parliament)

MAY 2023

BONAFIDE CERTIFICATE

This is to certify that the project report entitled **"Closed Caption Aligner Development"** submitted by **Aakarsh Srivastawa** bearing the **MIS No:112115001,** in completion of his/her project work for the project report submission in partial fulfillment of the requirements for the award of the degree of **Bachelor of Technology** in the **Department of Computer Science and Engineering**, Indian Institute of Information Technology, Pune (IIIT Pune), during the academic year **2022-23**.

Dr. Sanjeev Sharma

Head of the Department

Assistant Professor

Department of CSE

IIIT PUNE

Project Viva-voce held on 25/04/2023

Undertaking for Plagiarism

I/We **Students' Names** solemnly declare that research work presented in the **report/dissertation** titled **"CCExtractor Development"** is solely **my** research work with no significant contribution from any other person. Small contribution/help wherever taken has been duly acknowledged and that complete report/dissertation has been written by **me**. I understand the zero tolerance policy of **Indian Institute of Information Technology Pune** towards plagiarism. Therefore **I** declare that no portion of my **report/dissertation** has been plagiarized and any material used as reference is properly referred/cited. I undertake that if I am found guilty of any formal plagiarism in the above titled thesis even after award of the degree, the Institute reserves the rights to withdraw/revoke my **B.Tech** degree.

Aakarsh Srivastawa

25 April 2023

Conflict of Interest

Manuscript title:Closed Caption Aligner Development

The authors whose names are listed immediately below certify that they have NO affiliations with or involvement in any organization or entity with any financial interest (such as honoraria; educational grants; participation in speakers' bureaus; membership, employment, consultancies, stock ownership, or other equity interest; and expert testimony or patent-licensing arrangements), or non-financial interest (such as personal or professional relationships, affiliations, knowledge or beliefs) in the subject matter or materials discussed in this manuscript.

Aakarsh Srivastawa

25/04/2023

ACKNOWLEDGEMENT

This project would not have been possible without the help and cooperation of many. I would like to thank the people who helped me directly and indirectly in the completion of this project work.
First and foremost, I would like to express my gratitude to our honorable Director, **Prof. O.G. Kakde**, for providing his kind support in various aspects. I would like to express my gratitude to my project guide **Dr.Sanjeev Sharma, Department of CSE**, for providing excellent guidance, encouragement, inspiration, constant and timely support throughout this **B.Tech Project**. I would like to express my gratitude to the **Dr.Sanjeev Sharma, Department of CSE**, for providing his kind support in various aspects. I would also like to thank all the faculty members in the **Department of CSE** and my classmates for their steadfast and strong support and engagement with this project.

Abstract

Time lagging problem is needed to be solved when using closed caption as a source of features in video indexing with its host video. One of solution is aligning closed caption to transcripts then take the time code from transcripts as new references for closed caption. Closed captioning (CC) and subtitling are both processes of displaying text on a television, video screen, or other visual display to provide additional or interpretive information.The usual subtitle files have line by line synchronization in them i.e. the subtitles containing the dialogue appear when the person starts talking and disappears when the dialogue finishes. This continues for the whole video.

So the purpose of this project is to build a tool for word by word synchronization of subtitles with audio present in the video by tagging each individual word as it is spoken throughout the video.

For eg-
00:10:24,233 -> 00:12:19,887
My name is Aakarsh Srivastawa

This appears for around 2 second but the aim of this project is that the word by word synchronization is tagged such as-

My - [00:10:24,233—00:10:36,685]
Name -[00:10:36,685–00:10:49,666]
Is -[00:10:49,666—-00:11:00,102]
Aakarsh - [00:11:00,102—00:11:38,301]
Srivastawa -[00:11:38,301—00:12:19,887]

In the above example each word from the subtitle is tagged with beginning and ending timestamps based on audio and remained throughout the process.It is the one tool that is free, portable, open source and community managed that can take a recording from a TV show and generate an external subtitle file for it.

The tool will be primarily in C++ or C as is its parent program CCExtractor with some probable Python components.

Keyword- C , C++ , PocketSphinx , Kaldi ,XML,XML schema

TABLE OF CONTENTS

List of Figures / Symbols/ Nomenclature

INTRODUCTION

Overview of Work-

Subtitles and captions are an integrated part of the lives of numerous people . People use them for a variety of reasons ranging from learning a new language to as an aid for the hearing impaired . The capability which this project will add makes me extremely excited. This technique of tagging each word to audio unlocks completely new avenues like far better subtitle synchronization, audio video editing. So the aim is to provide it to users/developers in as flexible a form as possible so that people can build upon it for their personal use cases that they can think of.

Motivation of Work-

In video processing, both video and audio dimensions of content are intuitively helpful. Recently, lots of video features are applied for better performance, but research about valuable audio features, also believed to contain significant information, is relatively much less than video ones . From my perspective the speech transcript is a feature carrying most of semantic information among lots of features produced from audio streams. Instead of taking advantage of transcripts from machines, transcripts are also available from humans, especially on TV programs with more reliability in many dimensions, such as names, new terms, and accuracy. Moreover, special markers and punctuation could be valuable references also. These close captions, available in almost all programs, however, usually suffer lag problems constraining its application in video indexing especially when the program is broadcasted live. In order to provide a good feature for video processing, the project solves the lag problem in closed caption of live news programs by doing alignment.

Literature review -

On the basis of Research paper of Automatic Closed Caption Alignment Based on Speech Recognition Transcripts By Chih-wei Huang describes a work using limited resources to generate decent alignment results.It describes that both video and audio dimensions of content are intuitively helpful These close captions, available in almost all programs, however, usually suffer lag problems constraining its application in video indexing especially when the program is broadcasted lively. In order to provide a good feature for video processing, the project solves the lag problem in closed caption of live news programs – by doing alignment. It has also given algorithm using Dynamic Programing and Scoring.

Research Gap -

In the research paper it does not state how to caption the host video that is broadcasted without any type of lagging. People with multilingualism are sometimes unable to catch up with the accent of another language . The Research helps to caption the People learning a new language as an aid for hearing impaired. The caption of live video can be captioned here without any buffer in the time with the proper alignment .

Problem Statement

Research Objectives -

In this project, two kinds of word streams, automatic speech recognition (ASR) output and closed caption (CC), from the same spoken document are put to alignment. After that, CC words are affiliated with new time codes from corresponding ASR words. There are two main contributions in this paper: 1) Proving the possibility using the speech recognizer embedded in Windows XP system to do alignment even though it is capable of mediocre performance only, and the free SDK on the official site provides enough power to handle what the work needs. This is valuable for other groups or individuals to replicate the alignment work. 2) Fusing temporal and structural attributes in the alignment process based on classic dynamic programming algorithms since using simply the word identity comparison method cannot handle the entire document especially on commercial and fast spoken parts.

The capability which this project will add makes me extremely excited. This technique of tagging each word to audio unlocks completely new avenues like far better subtitle synchronization, audio video editing (e.g. Videogrep), commentary and podcasts analysis, accurate transcription, enhancement of aforementioned projects et cetera.

I look at this technology from the perspective of enabling a whole new dimension of use cases. So the aim is to provide it to users/developers in as flexible a form as possible so that people can build upon it for their personal use cases that they can think of.

From a technical viewpoint, the high level workflow for this task basically involves extracting audio from a video file, pre-processing if required, do audio analysis - mainly ASR, use existing subtitles as reference and then generate timestamps.

Methodology of Work -

1. A generalized tool with the following functionality:
   a. Subtitles parsing and processing.
   b. Words and audio tagging.
   c. Generation of timestamps for each word.
   d. Audio based subtitle synchronization.
   e. Speech Recognition based transcription.
   f. Generation of difference between transcript and subtitles.
   g. Indication of missing / extra words.
   h. Generation of output in various formats.

2. Provision for extending the tool so the user can define the method being used to perform the action (e.g. if the user wants to perform subtitle synchronization by offset calculation method or by thorough analysis).
3. Minimal and functional UI.
4. Tests for various logical units.
5. Documentation within the tool.
6. Setup scripts.
7. Blog posts on a weekly basis + in case of any special event.

IMPLEMENTATION DETAILS

The tool will be primarily in C++ or C as is its parent program CCExtractor with some probable Python components.
The tools that will be leveraged in the overall workflow are:

1. Automatic Speech Recognition system :
   a. **PocketSphinx** : PocketSphinx is a lightweight large vocabulary, speaker-independent continuous speech recognition engine written in C++. They are under BSD-style license.
   b. **Kaldi** : Kaldi is a toolkit for speech recognition written in C++ and licensed under the Apache License v2.0.

We will make use of this to create and locate phonemes or words from the audio files. PocketSphinx is the preferred choice, but options are kept open for discussion and testing.

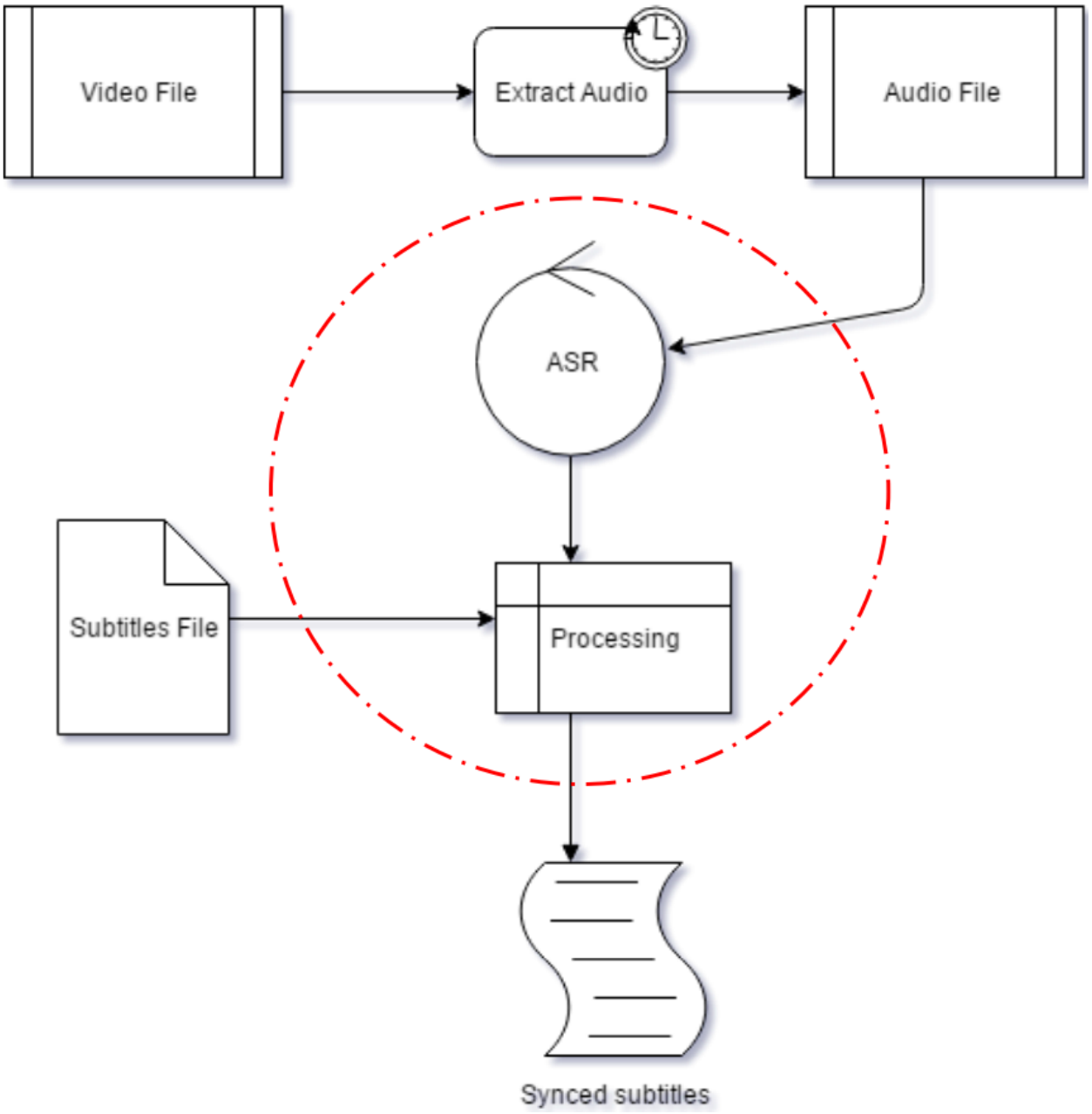
2. **FFmpeg**: This tool will be used to extract audio from video files and for format conversions and processing of audio files when needed.

3. **Forced Aligners** : The tool may use code inspired by / taken from several available open source tools such as Gentle, Rhubarb LipSync and PocketSphinx inbuilt continuous recognisor.

4. **CMUCLMTK** : Used for training language models. More open source tools maybe used as per need.

Analysis and Design

The main workflow is -



The main focus and time dedication will be towards the encircled region.

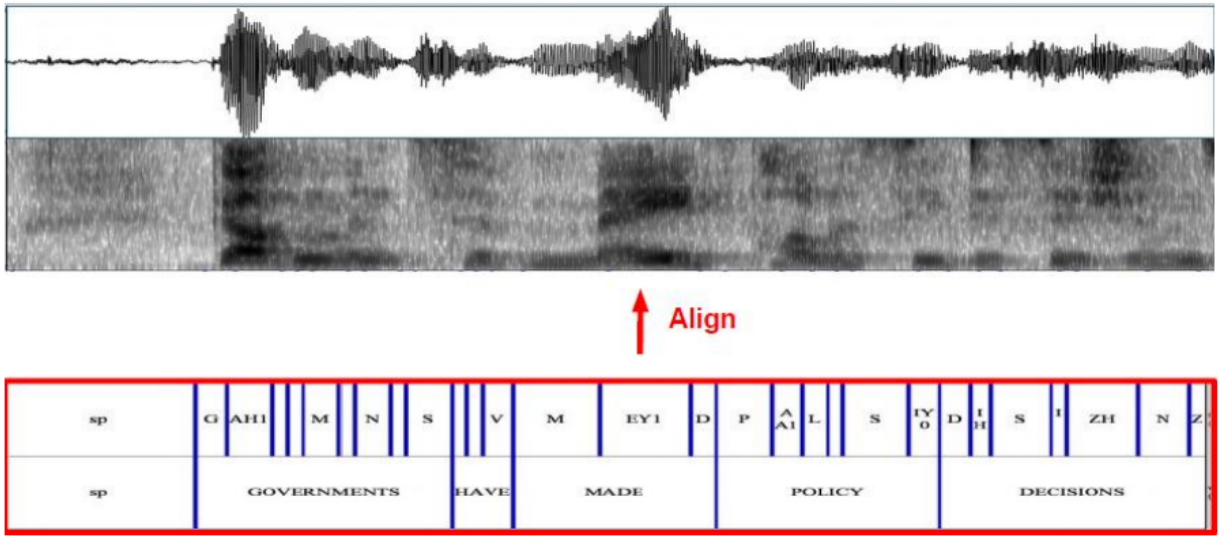
ASR technology will be used as a toolkit to perform Speech Recognition and Forced Alignment for lexical transcription or timestamps.

There are two primary cases that are possible during speech recognition :

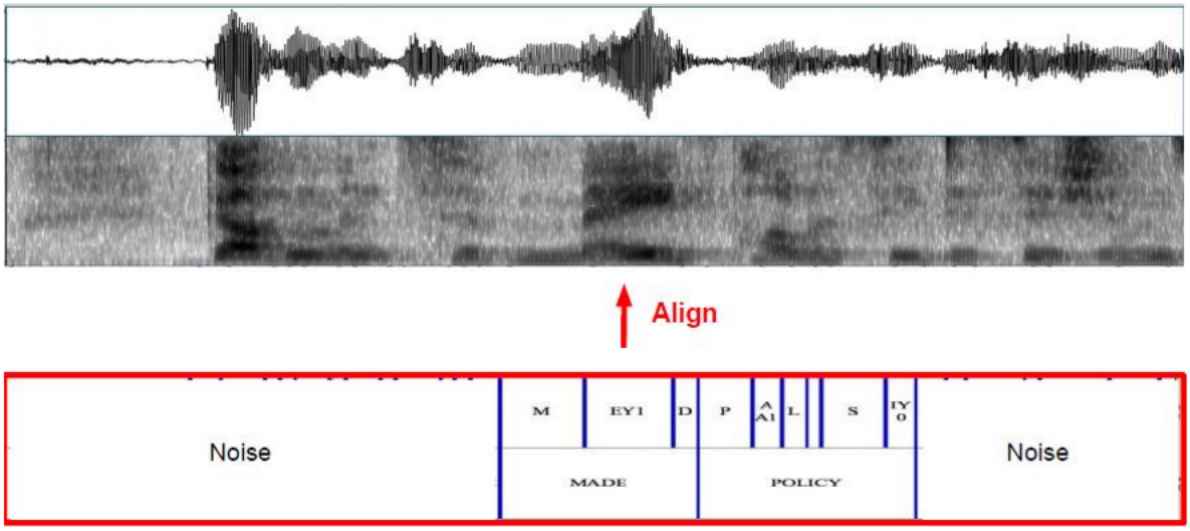
1. **Forced Alignment**

   This is the case where the transcription is already available and we know what exactly is in the

Audio.



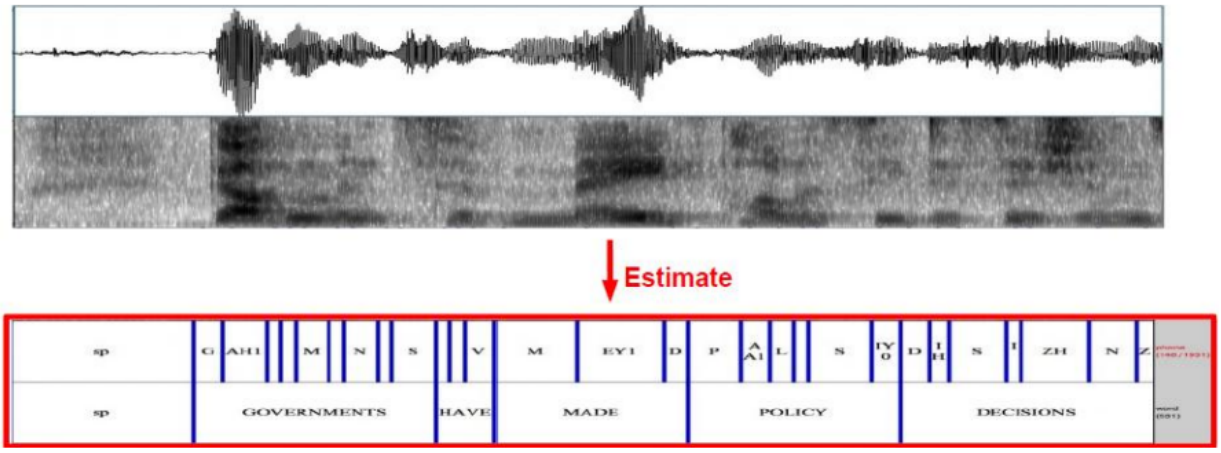
This is the case where some transcription is available and we know what is in some of the audio.



This is the case for which our tool needs ASR. While we have subtitles available sometimes they don't match audio 100%. The tool aims to provide a perfect audio-subtitle synchronization for those words that do match. For those words in the audio that don't appear in the subtitles, add a different indicator or maybe the word itself .

2. Speech Recognition

This is the case where the transcription is missing and we do not know what is in the audio.



The ideal recognizer is with very high accuracy then we do not have to use CC. Presently, however, the recognizers with enough accuracy to beat closed captions are available in very few labs only or cost a lot of money.Even with sufficient accuracy, the power of special markers like “>>>” and “>>” in CC cannot be replaced. Therefore, using not perfect recognizer combining CC alignment is one of the best ways to take advantage of CC to compensate for the lack of inaccurate ASR.

In alignment, a word in closed caption might be correctly, or closely, assigned a time code even its corresponding word in transcription is erroneously recognized. Obviously, higher recognition accuracy results in higher alignment accuracy, because two identical words in closed caption and speech have strong probability to occur at the same time spot. If erroneously recognized words can be located and matched to closed captions appropriately along time axis, word error rate could be tolerated to some extent.

RESULTS AND DISCUSSION

1. Approximation and Probability Based :

While using this method, the tool will calculate timestamps of the words based on the probability of them appearing within a given timeframe. This can be better visualized from the demonstration below

Consider a part of input subtitle :

347
00:09:29,241 --> 00:09:31,764
My name is Aakarsh

This means that all the words [My , name , is , Aakarsh] are definitely falling within the timeframe of [00:09:29,241] and [00:09:31,764] inclusive. Thus, taking a broad assumption - if each word was pronounced for equal amount of time, the time taken by each word will simply be ({end_timestamp} -{ start_timestamp}) / {no_of_words}

This of course will be wrong because in real life each word takes a different time to be spoken. This can be improved by considering the probability of duration of time a word could have taken. For example one criteria could be the length of the word, the higher the length the more time it'll take to pronounce.

Start full blown ASR work. Begin word detection, and write code for intelligently assigning timestamps on basis of frames and also probability. Try different setting and combinations to achieve maximum accuracy. The challenging part will be to incorporate missing and additional words present in subtitle. I will try to create a logic based on fuzzy search that shall look for words approximately ahead and behind the set domain.

```
static void print_word_times()
{
        int frame_rate = cmd_ln_int32_r(config, "-frate");
        ps_seg_t *iter = ps_seg_iter(ps, NULL);
        while (iter != NULL)
         {
            int32 sf, ef, pprob;
            float conf;
            ps_seg_frames(iter, &sf, &ef);
            pprob = ps_seg_prob(iter, NULL, NULL, NULL);
            conf = logmath_exp(ps_get_logmath(ps), pprob);
            printf("%s %.3f %.3f %f\n", ps_seg_word(iter),
```

```
            ((float)sf / frame_rate), ((float) ef /   frame_rate), conf); iter = ps_seg_next(iter);
            }
    }
```

The above code is mirror to the option -time which is used in command line tool pocketsphinx_continuous and will be adapted for our use case

2. Using phoneme detection :

Both Kaldi and PocketSphinx are capable of generating phonemes based on audio provided a trained model and suitable dictionary are supplied. One of the tasks will be to figure out a method to quickly generate relevant dictionary and model based on subtitles provided (since they are generally small) which should give more accuracy.

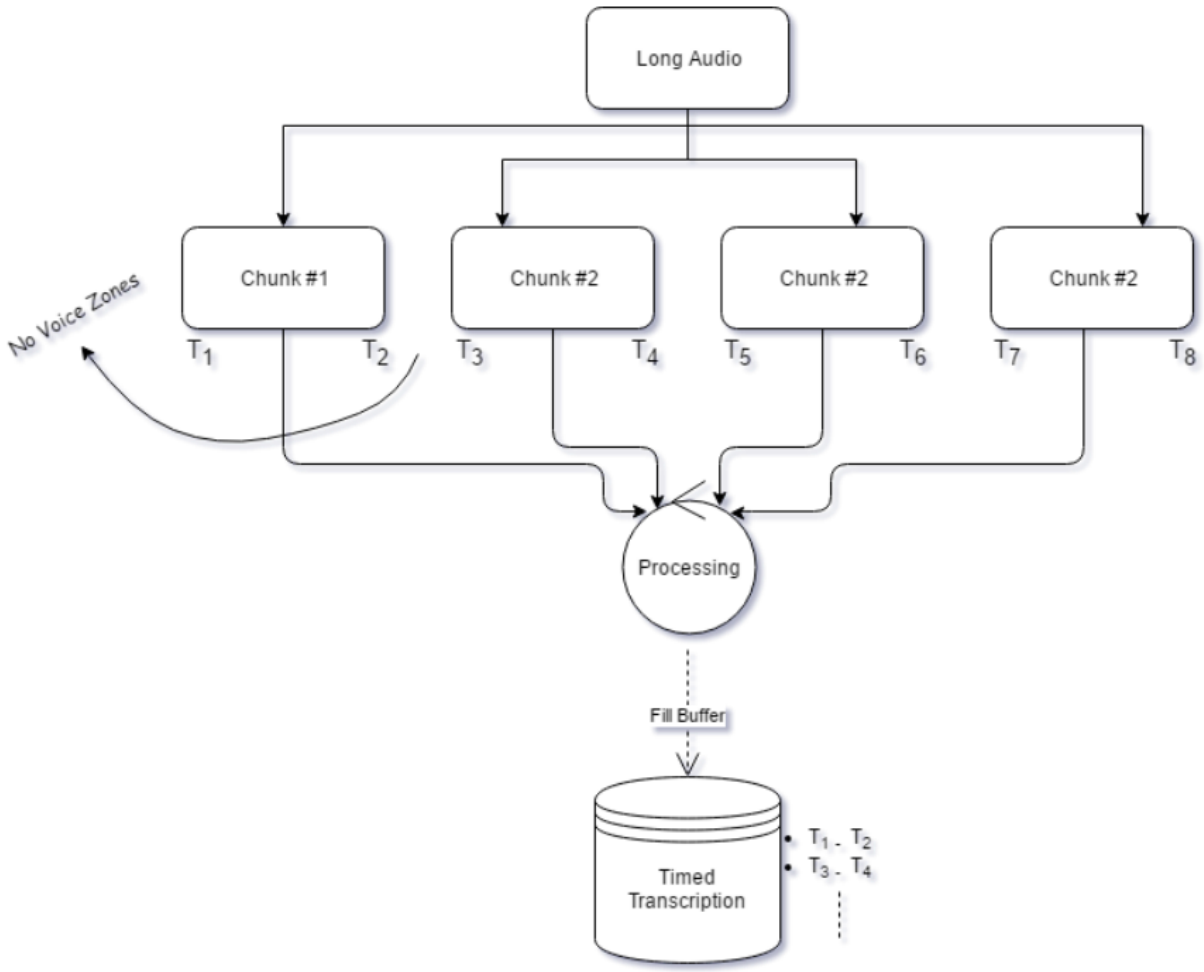
3. Using word detection :

Another approach is to directly perform speech to text from audio and based on the probability of the word being detected and create a timeline of words - thus getting word by word audio subtitle synchronization. Frames from which words begin and end can be very well used to calculate starting and ending timestamps.

All the other details such as breaking audio into smaller chunks with the help of silence zones or VAD(Voice Activity Detection) to process them more accurately, making the tool practically usable than just an academic tool et cetera are also kept in mind.

Implement VAD using either WebRTC or PocketSphinx (currently WebRTC is preferred choice as it gives more reliable data). Use VAD to make timestamps more accurate by ignoring the silence zone within a time span and then applying the synchronization algorithm.

Also, if it increases speed, cut videos into smaller chunks based on VAD and silence regions. The small chunks of video will be processed separately, adaptive to their content and their results then later combined to form a single transcription. The different timestamps T1, T2, .. can be used as a placeholder, against which calculated offset (time in millisecond) will be added to find timestamp of each word


Long Audio
Chunk #1
Chunk #2
Chunk #2
Chunk #2
No Voice Zones
T1
T2
T3
T4
T5
T6
T7
T8
Processing
Fill Buffer
Timed Transcription
T1 - T2
T3 - T4

Conclusion and Future Scope

The importance of captioning lies in its ability to make video more accessible in numerous ways. It allows Deaf and hard of hearing individuals to watch videos, helps people to focus on and remember the information more easily, and lets people watch it in sound-sensitive environments. As the use of video increases, organizations and video creators must keep in mind that captioning, though originally created for those with hearing loss, is beneficial to everyone.

-By 2030, the number of people over the age of 65 will be 20% of the population; around 1 in 2 adults over 65 experience hearing loss.

-There's been an increase in mild hearing loss among adolescents due to headphone use.

-Noise Pollution known as the "modern unseen plague," is a leading cause of hearing loss.

The Deaf and hard of hearing community is growing, and it's critical that as technology changes and video use increases that video is made accessible to this group. Since captioning serves Deaf and hard of hearing individuals the most, the importance of captioning will only grow as the population with hearing loss grows.

Closed captioning is likely to stay near the forefront of video provider's regulatory concerns going forward, as the FCC R&O also contained a "Further Notice of Proposed Rulemaking" (FNPRM) section.

Closed captioning benefits specific populations; however, measuring the broader educational impact of this technology is a worthwhile endeavor. Assessing educational technology for effectiveness is important prior to making recommendations for widespread use. Replication of the current study could be done by randomly assigning individual participants to a "caption" or "no-caption" condition. Future studies could collect data on an individual participant or group basis and should control for the influences of room size, video volume, and background noise. These factors may affect how participants learn via video-based information and perform on subsequent assessments. Furthermore, a large variety of videos and assessments should be utilized that cover different topics.

There are multiple benefits to closed captioning and subtitling, and given the importance of videos, including in the technical communication field, it is fair to say that adding closed captions to a video is a necessity to provide better access and user experience.

References

1)David C. Gibbon, Generating Hypermedia Documents from Transcriptions of Television Programs Using Parallel Text Alignment, Continuous-Media Databases and Applications. Eighth International Workshop on, 23-24 Feb. 1998

2) R.A. Wagner and M.J. Fischer, The String-to-string Correction Problem, Journal of the ACM, 21(1):168-173, January 1974

3)EIA-708-B digital closed captioning implementation by Robert N. Blanchard

4)https://www.ee.columbia.edu/ln/dvmm/publications/03/align03huang.pdf

5)Images are the courtesy of Forced Alignment and Speech Recognition Systems, Oxford University

6)Nye, H., The Use of a One-Stage Dynamic Programming Algorithm for Connected Word Recognition, IEEE Transactions on Acoustics, Speech, and Signal Processing, 1984