Automatic Caption Generator

Neel Patel[1], Hemil Pansuria[1], Jay Patel[1], Bhavik Darji[1], Prashant Sahatiya[2]

1. Student, Department of Information Technology, Parul University, Vadodara.

2. Assistant Professor, Department of Information Technology, Parul University, Vadodara.

Abstract:

The last ten years have been the witnesses of the emergence of any kind of video content. In the same time, certain individuals are deaf and occasionally cannot understand the meanings of such videos because there is not any text transcription available. Hence, it becomes important to make videos available to people who have these problems and even more to remove the gaps of native languages among them. This can be best done by providing subtitles of a video. However, downloading subtitles of any video from the internet is a tedious process. So, to generate subtitles automatically through the software itself and without the use of internet is the main concept of this paper. The objective of this paper is to provide an overview of generating subtitle on offline basis using CMUSPHINX4 java API. This system will first extract the audio, then recognize the extracted audio with CMUSPHINX4 java API. Later this system writes the recognized text to the text file with timestamp and saves it with .srt extension. Then, this .srt file can be opened in a media player to view the subtitles along with video.

Keywords: Audio Extraction, Speech Recognition, Sphinx4, Subtitle, .srt file.

Introduction:

The use of videos for the communication purpose has witnessed a phenomenal growth in the past few years. However, non-native language speakers or people with hearing disabilities are unable to take advantage of this powerful medium of communication. To overcome this problem caused by hearing disabilities or language barrier, subtitles will be an effective solution. The subtitles are provided in the form of a subtitle file most commonly having a .srt (SubRip Text) extension. Several software has been developed for manually creating subtitle file, however software for automatically generating subtitles are not available. This leads to a valid subject of work in the field of automatic subtitle generation. At present, in VLC media player, the subtitles have to be inserted first to the media player and then it is synchronized with the video. The file inserted needs to be .srt file containing the time intervals of the text spoken. It does not accept .txt file to synchronize the subtitles as it only contains sentences of spoken words in the video and not the time intervals of the spoken words. On the other hand, YouTube accepts both the .txt file and the .srt (SubRip Text) files for synchronizing the text. Nonetheless, software generating subtitles without intervention of individual using speech recognition have not been developed. So we came up with a solution which will generate subtitle file automatically using java speech recognition library (CMUSPHINX4) without any use of internet. For that we made a GUI using java in that user has to provide video file for which user want to create subtitle and then just click onto to generate subtitle button. The software will create subtitle and after creating subtitle it will play video with subtitle using VLC media player. For implementation we used java language and NetBeans IDE.

Concept of Automatic Caption Generator:

The Automatic Caption Generation process consists of four phases.
A) User Interface
B) Audio Extraction
C) Speech Recognition
D) Subtitle Generation

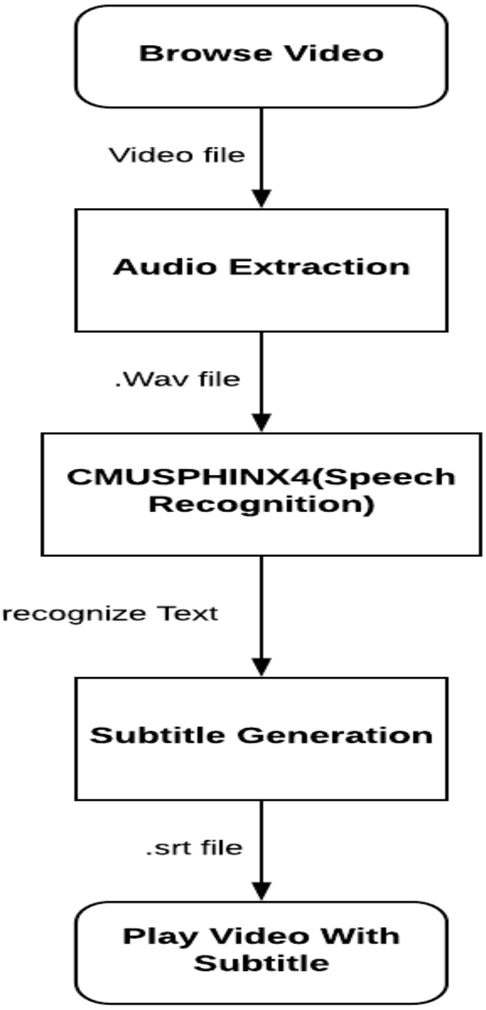


Fig. Subtitle generation mechanism

A) **User Interface:** Using user interface user can access to the software. We use Jframe to implement GUI. In GUI there is one button "Open" for browsing the video file for which user want to generate subtitle file and another button "Generate Subtitle" to Generate subtitle. After clicking on that button it will Start generating the subtitle and that whole process follows the below given methodology.

B) **Audio Extraction:** The speech recognition engine requires a .wav audio file as input. Hence, it is necessary to convert the video into .wav format. For that we use JAVE library to convert video file into Audio file. The **JAVE** (**J**ava **A**udio **V**ideo **E**ncoder) library is Java wrapper on the ffmpeg project. After extracting the audio from video files, it writes the file with .wav extension and then provided to speech recognition module for further processing. After completing this task software will display ".wav file Generated" dialogue message.

C) **Speech Recognition:** The .wav file obtained from the audio extraction phase will be passed forward for speech recognition. An open source speech recognition engine called CMU Sphinx4 will be used in this stage. CMU Sphinx4 requires following inputs: – Acoustic Model – Language Model – Dictionary file.

1) Acoustic model: An acoustic model is used in speech recognition to represent the relationship between an audio signal and the phonemes that make up speech. The model is learning from a set of audio recordings and their corresponding transcripts.

2) Language model: A language model is a grammar file for an entire language. It defines which word could follow previously recognized words and helps to significantly restrict the matching process by stripping words that are not probable.

3) Dictionary file: A dictionary file that contains the list of all words in the language along with their pronunciations. Like,
   - hello H EH L OW

- world W ER L D
- the TH IH
- the(2) TH AH

Our system deals with Engish language videos. Hence, the .wav file generated in previous phase is passed to CMU Sphinx4 along with the US English language model and dictionary file. The text output for the given audio will be generated and forwarded to subtitle generation phase.

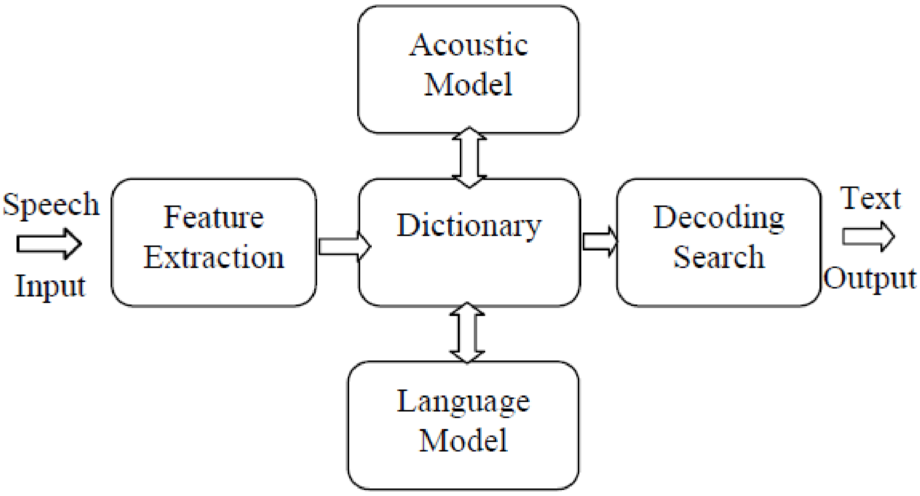
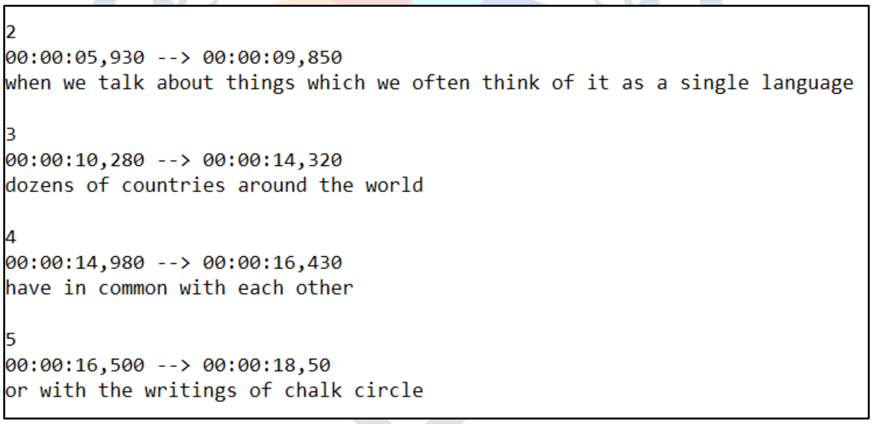


Fig. Overview of ASR system

D) Subtitle Generation: This module is expected to get a list of words and their respective speech time-frames from the speech recognition module and then produce a .srt subtitle file.



```
2
00:00:05,930 --> 00:00:09,850
when we talk about things which we often think of it as a single language

3
00:00:10,280 --> 00:00:14,320
dozens of countries around the world

4
00:00:14,980 --> 00:00:16,430
have in common with each other

5
00:00:16,500 --> 00:00:18,50
or with the writings of chalk circle
```

Fig. Content of generated .srt file

The above figure shows an example of subtitle file generated using the system. Subtitles are text translations of the dialogues in a video displayed in real time during video playback on the bottom of the screen. A typical .srt file has three sections first the Subtitle number indicating which subtitle it is in the sequence. Then the Subtitle timing that the subtitle should appear on the screen, and then disappear and at last the text followed by a blank line to indicate end of current subtitle. To get time-frames we use pattern matching concepts and we use java FileWriter class to perform create and write operation to the file.

Conclusion:

We analyzed that software is working correctly and the process is little bit time consuming. The Accuracy of speech recognition library (CMUSphinx4) is about 75-80 % for US English ascent videos. We encountered very few errors in .srt file. In some cases, mismatch of sample rate detected because of different frequencies of voice and therefor accuracy level compromised. We found that in order to increase the accuracy we need to train speech recognition module from scratch. In conclusion, Denser the dictionary of words and more number of data for training, more precise the subtitles are generated. We can train CMUSPHINX4 model for other language with sufficient datasets and make a system which can generate subtitle for other language videos.

REFERENCES:

[1] Abhinav Mathur, Tanya Saxena, “Generating Subtitles Automatically Using Audio Extracting And Speech Recognition”, IEEE 2015

[2] Rizwan Sheikh, Swapnil Suryajoshi, Shivam Gupta, Sushant Tayde, M. S. Burange, “An Approach Towards Generating Subtitles Automatically from Videos by Extracting Audio”, GRD Journals 2018

[3] Akshay Jakhotiya, Ketan Kulkarni, Chinmay Inamdar, Bhushan Mahajan, Alka Londhe, “Automatic Subtitle Generation for English Language Videos”, SSRG-IJCSE 2015

[4] Akhil Kanade, Sourabh Gune, Shubham Dharamkar, Rohan Gokhale, “Automatic Subtitle Generation For videos”, IJERGS-2015.

[5] Boris Guenebaut, “Automatic Subtitle Generation for Sound in Videos”, University west, Department of Economics and IT.