iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL
FOR RESEARCH
IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 11 Issue: VII Month of publication: July 2023

DOI: https://doi.org/10.22214/ijraset.2023.54589

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 11 Issue VII Jul 2023- Available at www.ijraset.com

Text Summarization for Education in Vernacular Languages

Divya V[1], Prithica G[2], Savija J[3]
[1, 2]UG Student, [3]Assistant Professor, Loyola Institute of Technology, Chennai-3, Tamil Nadu, India

Abstract: This project proposes a video summarizing system based on natural language processing (NLP) and Machine Learning to summarize the YouTube video transcripts without losing the key elements. The quantity of videos available on web platforms is steadily expanding. The content is made available globally, primarily for educational purposes. Additionally, educational content is available on YouTube, Facebook, Google, and Instagram. A significant issue of extracting information from videos is that unlike an image, where data can be collected from a single frame, a viewer must watch the entire video to grasp the context. This study aims to shorten the length ofthe transcript of the given video. The suggested method involves retrieving transcripts from the video link provided by the user and then summarizing the by using Hugging Face Transformers and Pipelining. The built model accepts video links and the required summary duration as input from the user and generates a summarized transcript as output. According to the results, the final translated was obtained in less time when compared with other proposed techniques. Furthermore, the video's central concept is accurately present in the final without any deviations.

I. INTRODUCTION

The number of YouTube users in 2020 was approximately 2 billion, and has been increasing every year. Every minute, 300 hours of YouTube videos are uploaded. Almost one-third of the YouTube viewers in India access

videos on their mobiles and spend over 48 hours a month on the website, a Google study said. It is frustrating and time consuming to search for the videos that contains the information we are actually looking for. For instance, there are many Ted Talk videos available online in which the speaker talks for a long time on a given topic, but it Is hard to find the content the speaker is mainly focusing on unless we watch the entire video. Many machine learning based video summarization techniques are present but they require devices with large processing powers, this is because each video contains thousands of frames and processing all frames takes a very long time. we propose to use the LSA Natural Language Processing algorithm, which requires less processing power and no training data required to train the algorithm.

II. METHOLOGY

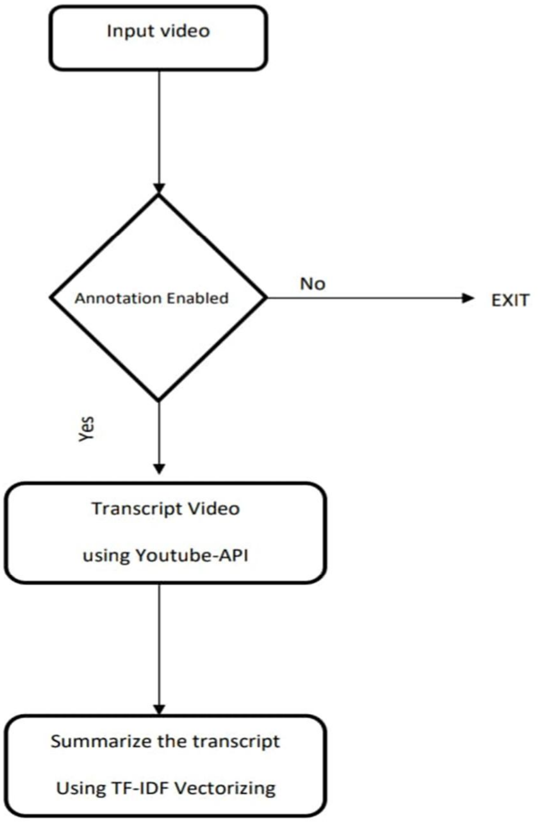
First, we need to get the subtitles or transcript for a given Youtube video id by using the python API known as youtube_transcript_api. Since there are three types of transcript that we can extract - manually generated transcript, automatically generated transcript, and the videos that contain no transcript. We are not considering videos that do not have transcripts.

Secondly, when we get the transcript of a given Youtube video since it does not contain any punctuations like comma(,), full stops(.) which is very important for us in finding the boundaries of a sentence, so we will restore punctuations from our extracted transcript by using the python library known as "punctuator". Now we will apply the preprocessing methods to clean the extracted transcript by tokenizing the sentences as well as the words, lowercasing it, removing stop words like a, an, the, etc., removing punctuations, and stemming or lemmatization to generate the root form of inflected words. Performing summarization: This task consists of shortening a large form of into a precise summary that keeps all the necessary information intact and preserves the overall meaning. For this purpose in NLP for summarization, there are two types of methods used :Extractive Summarization: In this type of summarization, the output is only the important phrases and sentences that the model identifies from the original .For the purpose of extractive summarization, we have used the TF-IDF model with Rank Algorithm TF-IDF(Term Frequency - Inverse Document Frequency)After the cleaning process, we have to convert the words into it's vectorized form so that our algorithm will process it by using TF-IDF. This is a technique to measure the quantity of a word in documents, we compute a weight to each word which signifies the importance of the word in the document and corpus. TF(Term Frequency): TF calculates the frequency of a word in a document. TF = No. of repetition of the word in the sentence / No. of wordsin a sentence IDF(Inverse Document Frequency): IDF is the inverse of the document frequency which measures the informativeness of term t. IDF = log(No. of sentences / No. of sentences containing words) After this, we will multiply both matrices to obtain the vectorized form which tells us which words are the most important.

III. PROPOSED SYSTEM

The proposed system takes input of a YouTube video link and the time duration to which video has to be summarized. Using NLP based LSA algorithm, video will be transcripted and summarized.

FLOW/ ER DIAGRAM



IV. PROJECTMODULES

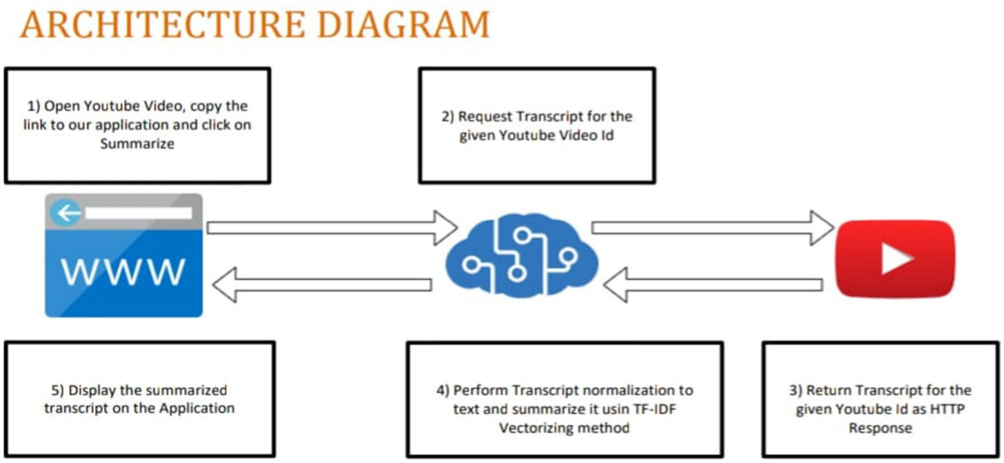
The basic structure of the youtube summarizer is that we are downloading the subtitles of the provided youtube video using the python module, Youtube-Transcript-API, and then performing the preprocessing techniques and then finally doing different summarization algorithms for summarizing the given .

A. *Summarization using TF-IDF Vectorizer.*

TF-IDF is the acronym of Term Frequency-Inverse document Frequency, and it is a measure used to evaluate how important a word is to a document in a collection or corpus. TF-IDF numerical statistic is used in information retrieval and mining.

1) *TF (Term Frequency):* estimates how frequently a term occurs in a document. Since every document is different in length, it is possible that a term would appear much more times in long documents than in shorter ones. Thus, the term frequency is often divided by the document length TF(t) = (Number of times term t appears in a document) / (Total number of terms in the document)
2) *IDF (Inverse Document Frequency):* which measures how important a term is. While some terms are considered equally important, other ones like “is”, “of” or “that” are less important but they may appear a lot of times in a document. Thus, we need to weigh down the frequent terms while scale up the rare ones by computing the following: IDF(t) = log_e (Total number of documents / Number of documents with term t in it)

V. ARCHITECTURE DIAGRAM


ARCHITECTURE DIAGRAM
1) Open Youtube Video, copy the link to our application and click on Summarize
2) Request Transcript for the given Youtube Video Id
WWW
3) Return Transcript for the given Youtube Id as HTTP Response
4) Perform Transcript normalization to text and summarize it usin TF-IDF Vectorizing method
5) Display the summarized transcript on the Application

VI. YOUTUBE

YouTube is the second most visited website worldwide. The range of videos on YouTube includes short films, music videos, feature films, documentaries, audio recordings, corporate sponsored movie trailers,livestreams, vlogs, and many other contents from popular YouTubers. YouTube users watch more than one billion hours of video every day. Hence, we have considered YouTube videos as the data for our proposed video summarization algorithm. Using the link, YouTube transcript API will extract subtitles from that particular video. Downloading videos from YouTube is difficult. To do so first we have to copy the link of the video we want to download then paste the link in the YouTube video downloader website. This method of downloading is time consuming. Pytube is a lightweight, dependency-free Python library which is used to download YouTube videos easily. This can be achieved with just one or two lines of code. Its library creates the object of the YouTube module by passing a YouTube link of the video as the parameter. Then, it gets the appropriate extension and resolution of the video. Name of the file can be kept based on user convenience. After that, download the file using the download function of pytube library. This download function takes only one parameter: the location where downloaded files need to be saved.

VII. PREPROCESSING

The initial processing step is the first significant stage in natural language processing which consists of three stages. The first one is tokenization which splits each phrase into a series of words or terms. The second is toeliminate English stop words, which is a way to efface letters and words with no denotement in the sentence and reiterate more than once in the so that the will be pristine from stop words. Table 1 shows a sample of the stopwords. The last stage is word-stemming; the central concept is to handle the word that cessations or beginning by minimizing the phrases or words to their word roots, kenned as a lemma. Stemming is typically performed before the word's final assignment to the index by deleting all affixed suffixes and prefixes (affixes) from index words.

VIII. FIND KEYWORDS

Finding keywords is the consequential step in the system, to filter the words in the , TF-IDF was utilized this approach measures the words consequential in a sentence and the number of times a word is included in a . The word is very paramount if it is reiterated in a sentence, but less reiterated in a document [21], [22]. TF-IDF is equal to TF*IDF, both TF and IDF were computed (1) and (2) respectively: where $f(i, j)$ is the number of repetitions of the word i in document j. It's worth noting that the numerator is just the entire number of phrases in document j (counting each occurrence of the same term separately where is the total number of sentences in the input , and di is the number of sentences where the word i appears. In our example, the number of sentences is after splitting it according to the end with one of the special characters (".", ",", ";", "?", "!") and the keywords are: let, cricket, bit lazy, saturday, play, week, monday, tell, page, essay, library, noticeboard, tomorrow, things postponing, come, articles and article.

IX. SENTENCES SCORE

After the calculation is culminated, the words must sort in descending order according to their value. The sorting of all words is very consequential to test the TF-IDF rank. Afterward, the sentence's consequentiality value should be calculated utilizing the sum of each verb and entity in it, the values should be sorted in decrementing order.

X. LIBRARY

A. Youtube_transcript_api

This is an python API which allows you to get the transcripts/subtitles for a given YouTube video. It also works for automatically generated subtitles, supports translating subtitles and it does not require a headless browser, like other selenium-based solutions do!

B. NLTK

The Natural Language Toolkit (NLTK) is a platform used for building Python programs that work with human language data for applying in statistical natural language processing (NLP). It contains processing libraries for tokenization, parsing, classification, stemming, tagging and semantic reasoning.

C. Sklearn

Scikit-learn (Sklearn) is the most useful and robust library for machine learning in Python. It provides a selection of efficient tools for machine learning and statistical modeling including classification, regression, clustering and dimensionality reduction via a consistence interface in Python. This library, which is largely written in Python, is built upon NumPy, SciPy and Matplotlib.

D. Tkinter

Tkinter is a standard library in python used for creating Graphical User Interface (GUI) for Desktop Applications. With the help of Tkinter developing desktop applications not a tough task. The primary GUI toolkit we will be using is Tk, which is Python's default GUI library. We'll access Tk from its Python interface called Tkinter (short for Tk interface).

XI. CONCLUSIONS

The increase in popularity of video content on the internet requires an efficient way of representing or managing the video. This can be done by representing the videos on the basis of their summary.

REFERENCES

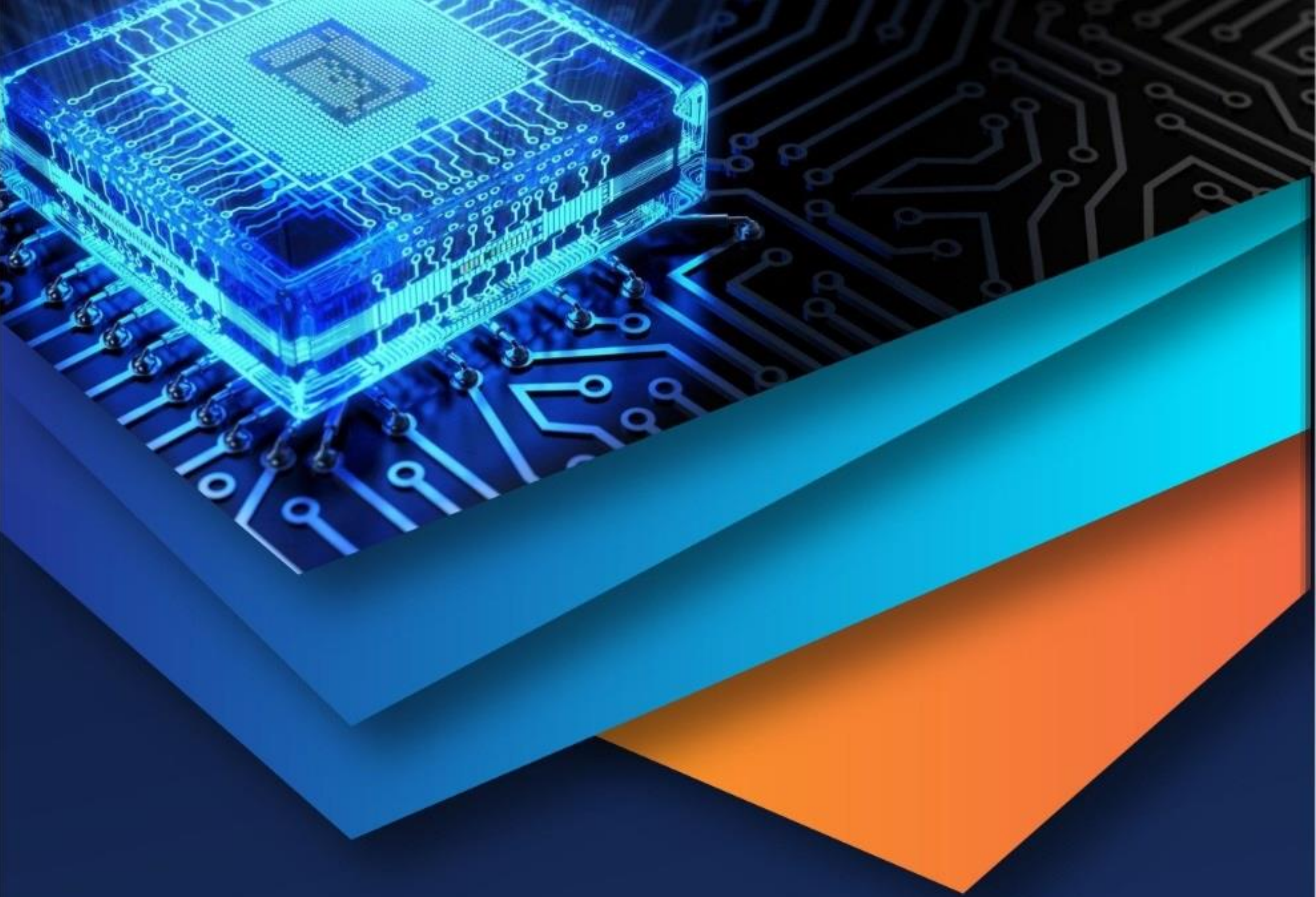
[1] K. Prudhvi, A. B. Chowdary, P. S. R. Reddy, and P.L. Prasanna, " summarization using natural language processing." in Advances in Intelligent Systems and Computing, vol. 1171, pp. 535-547, 2021, doi:10.1007/978-981-15-5400-1_54.

[2] R.Boorugu and G.Ramesh, "A survey on NLP base summarization for product reviews," in processing of the 2nd International Conference on Incentive Research in Computing Applications, ICIRCA 2020, Jul.2020, pp.352-356,doi:10.1109/ICIRCA 48905.2020.9183355.

[3] S.Sah, S.Kulhare, A. Gray, S. Venugopalan, E. Prud'hommeaux, and R.Ptucha,"Semantic text summarization of long videos," in 2017 IEEE Winter Conference on Applications of Computer Vision, WACV 2017, 2017, pp.989997,doi:10.1109/WACV.2017.115.

[4] Gousiya Begum1,Dharma Ashritha3 YOUTUBE TRANSCRIPT SUMMARIZER in international journal of creative research thoughts, vol. 10, 2022

[5] Rand Abdul wahid Albeer, Huda F. Al-Shahad, Hiba J. Aleqabie, Noor D. Al-shakarchy "Automatic summarization of YouTube video transcription" Indonesian Journal of Electrical Engineering and Computer Science Vol. 26, No. 3, June 2022, pp. 15-1519



10.22214/IJRASET

INDEX COPERNICUS

45.98

IMPACT FACTOR:
7.129

TOGETHER WE REACH THE GOAL

IMPACT FACTOR:
7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY