RESEARCH ARTICLE OPEN

Automatic Extraction of Text Notes from Video Tutorials

Sunil Poojari *, Abhinandan Kulkarni **, Anuradha Vishwakarma ***, Prof. Savita Lohiya ****

**Student (Department of Information Technology, SIES Graduate School of Technlogy, Navi Mumbai, Maharashtra, India*
*** Student (Department of Information Technology, SIES Graduate School of Technlogy, Navi Mumbai, Maharashtra, India*
**** Student (Department of Information Technology, SIES Graduate School of Technlogy, Navi Mumbai, Maharashtra, India*
***** Assistant Professor (Department of Information Technology, SIES Graduate School of Technlogy, Navi Mumbai, Maharashtra, India*

ABSTRACT
Today's rapidly digitizing world is a world that is redefining the very concept of interaction. One of the major strides being taken is in the ever-dynamic, ever-expanding world of education. As a result, there is an increased demand for video tutorials for study purpose, online courses and tutorials for entrance exams, as most of the videos are proved to be effective, uploaded by experts in domain and is a very effective audio video learning resource. But the videos are sometimes long and time-consuming and explained way too briefly for an average user. As a result, the user doesn't get enough time to comprehend the context of the video enough to make notes or make a proper understanding of the subject matter. Thus, we propose a system that would automatically generate text notes for the video tutorial which would contain the gist of the video tutorial. It will use the concept of text summarization that is used to summarize the video tutorials after transcribing video and would display the important contents in a notes format. In the core of this system we have used NLTK (Natural Language Toolkit) which is a python library for working on Natural Language Processing.

Keywords **–** Text Summarization, NLP, NLTK

Date of Submission: 21-07-2020 Date of Acceptance: 06-08-2020

I.INTRODUCTION

Text summarization is a technique of reducing the long version of text documents. The goal of text summarization is to create a reasonable and expressive summary having only those important points which are highlighted in the document. One of the major concept in machine learning and Natural Language Processing (NLP) is Automatic Text Summarization. In simple terms, the process of automatic summarization involves computer creating a short end version of the text. The important aim of automatic summarization is to shrink the size and volume of a source text into a short and sorted version which holds the altogether meaning and information content.

The text that would be summarized would be generated using Speech-To-Text API provided by IBM Watson. The process of text summarization is done using NLTK Library which allows sentence stemming and ranking of words. The main aim of this project is highlighted in this research paper that is to automatically create a text document from the video tutorials for future reference. The process of taking notes in handwritten fashion is quite time consuming and it also tends to break concentration while watching the video.

Google Cloud & IBM Watson provides a limited version of their Speech-To-Text API, one can use it for converting video tutorials into text, and then that text is processed and converted into a shorter version for later reference.

II. NEED OF STUDY

Students & People around the globe are moving towards video tutorials available online for learning purposes because the video tutorials more "dynamic" and "interactive" to understand compared to long lines and streams of text available in the textbook. There are numerous such videos available online, however, a huge problem arises when most of the videos are too long, banal and often the main points of the video are completely missed while taking down the notes. Hence, we come with a system which would automatically extract the useful and informative concepts into text notes and thus condensing the whole video tutorial, which priorly might have been too long and banal, into a more digestible and engaging format of bite-sized notes, which is done by the summarization process. This will save one's time and it will be efficient to use. It would also save one’s time required for taking down the notes and will also help them to concentrate on the lecture properly.

III. LITERATURE SURVEY

Young, Tom, Devamanyu Hazarika, Soujanya Poria, and Erik. [1] "Recent trends in deep learning based natural language processing." In the above mentioned paper, we review a notable deep learning related models and methods that have been employed for frequent NLP tasks and provide a go through of their evolution. We also summarize, compare and contrast the various models and put forth a brief understanding of the past, present and future of deep learning in NLP.

Hahn, Udo, and Inderjeet Mani. [2] "The challenges of automatic summarization." This paper tells us about the challenges faced while text summarization, different methods of extraction, and a brief idea of how text summarization works.

Luhn, H. P. [3] "The Automatic Creation of Literature Abstracts" This mentioned paper describes the research done in the 1950s at IBM. In this work, he stated that the frequency of each particular word in an article provides a primary measure of its significance. There are various key ideas put forth in the mentioned paper that have major role in later work on summarization process. As a foremost step, words were arising from their root forms, and stop words were deleted. Later, he then compiled a list of content words sorted by the decreasing of the frequency; the index provides a significant measure of the word. On a sentence level, a remarkable factor was derived which reflects the number of occurrences of significant words within a sentence, and the linear distance between them due to the intercession of non-significant words. All the sentences are ranked in the order of their significance factor, and the highest –ranked sentences are finally selected from an auto-abstract.

Baxendale, P. [4] “Machine-made index for technical literature” It provides the early perception on particular feature which helps to find the important parts of the documents: the sentence position. On this aim, the author examined 200 paragraphs to find that in 85.

Edmundsun [5] “New methods in automatic extracting.” described a system that automatically generate extracts. In this paper, he majorly contributed by developing a system that uses extractive summarization technique. At foremost, the author of the paper made a protocol/system for generating manual extracts, this was applied on many technical documents (400 to be precise). The two most important features that were applied are word frequency and the positional importance of the sentence. There were two other features as that were used i.e. the significance of the words and the skeleton of the documents whether the sentence is the title or heading of the document, based on these features of the sentences weights were assigned to the sentences depending on the features manually to score each and every sentence.

Krishnaveni, P., and S. R. Balasundaram. [6] “Automatic text summarization by local scoring and ranking for improving coherence.” As there is a lot of textual data found on the internet has resulted in the need for machine generated summarization. As the text documents are too large it is very difficult to perform summarization manually. So, we need an automatic text summarizer that summarizes the data automatically without human efforts. Automatic text summarization means “reducing the content of source text into smaller version while preserving its actual meaning. Although automatic text summarization was started in the year 1950’s but it’s still not been able to achieve important and meaningful summaries of the data. This approach gives the feature which provides heading wise summarization of text to increase the quality by improving the understand-ability and coherence of the summary text. The document is summarized through local ranking and local scoring. The sentences in the source text are given ranking as per heading wise and selects the top ‘n’ sentences from each heading where ‘n’ depends on the compression ratio. The heading wise summary which is generated by this approach is the summary of individual headings. There is an equal proportion of sentence from every heading in each heading wise summary. It overall improves the meaningful and understandable content in the summary text.

Madhuri, J. N., and R. Ganesh Kumar. [7] "Extractive Text Summarization Using Sentence Ranking." In this paper, she describes problems faced in today’s era because of the huge amount of data & how to tackle that problem using extractive summarization which is one of the methods of summarizing text which is based on the words frequency and sentence ranking method. First, the frequency of the words is calculated and based on that sentence ranking is done, all this is done using NLTK which is Python Library for Natural Language Processing.

IV. METHODOLOGY

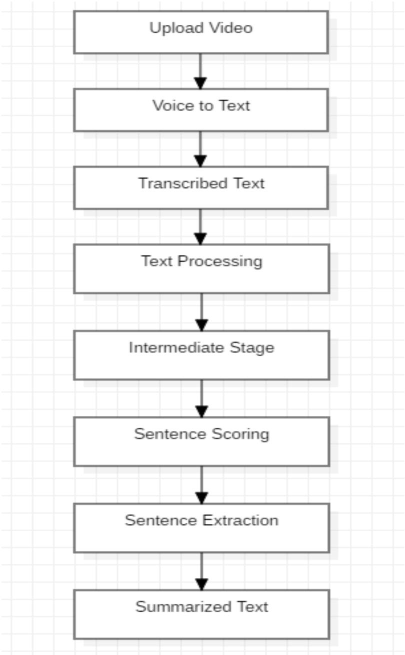


Figure 1: *Text Summarization Block Diagram*

Figure 1 represents text summarization block diagram. In the first phase, the speech or audio from the video is converted into the text format through Speech-To-Text API Further, the text format is summarized by using NLTK (Natural Language Toolkit). Firstly, source text will be passed to the program and then the frequency table gets generated, the frequency table is for determining how many times a particular word is repeated. Then sentence tokenizer is used for sentence tokenization. Once the sentence tokenization is done, then the sentence will be compared with the frequency table (that is generated by the algorithm which consist of word count of each word) and the sentence which have a greater number of keywords repeated with a higher frequency that sentence is given a weight. There is a threshold value given in the program the sentences having a value greater than that threshold value are considered for summarized text. After the summary gets generated, a text document is created in which summarized data is present. The user can download both the documents, summarized one as well as the Full-Text version which contains entire Speech-To-Text of that video

Once the summarization is done user can download the document, there are 2 formats available to download, 1st is the full text version which contains entire Speech-To-Text as it is from the video, 2nd is the summarized text which contains the summarized version that is available after text processing & summarization process.

V. DESIGN & MODELING

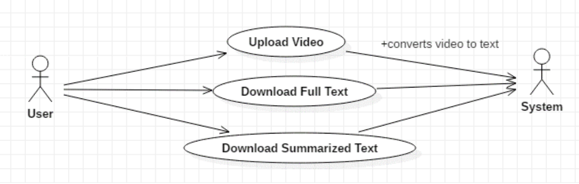


Figure 2: *Use-case Diagram*

Figure 2 represents use case diagram. The above use case diagram highlights the main function of the projects. The foremost step is whenever any video is being uploaded, the next step is the conversion of that video into text form. After this step, the summarization of the text is done. The user gets the normal text into summarized text form. One can download both summarized as well as non-summarized text

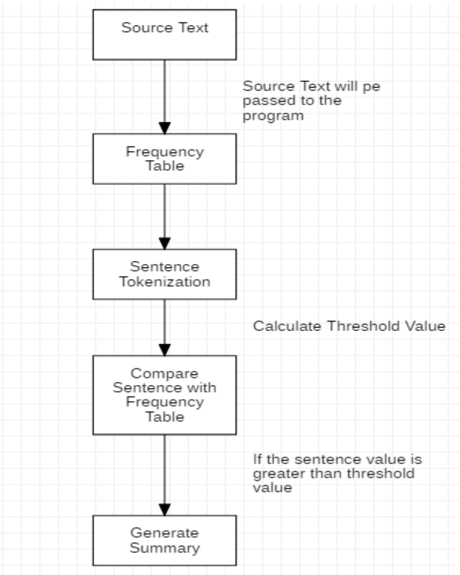

Source Text
Source Text will pe passed to the program
Frequency Table
Sentence Tokenization
Calculate Threshold Value
Compare Sentence with Frequency Table
If the sentence value is greater than threshold value
Generate Summary

Figure 3*: Flow Diagram for Text Summarization*

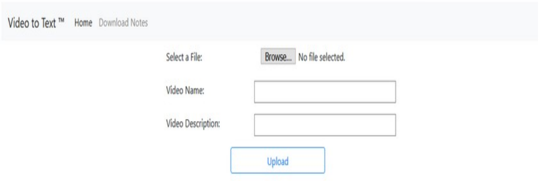
Figure 3 represents the flow diagram for text summarization. In the first phase, the speech or audio from the video is converted into the text format through Speech-To-Text API. Further, the text is summarized by using NLTK. In this process a source text will be passed to the program and then the frequency table gets generated, the frequency table is for determining how many times a particular word is repeated in the entire source text passed to the program, using this frequency the weightage of sentence is decided based on how many times the most repeated/popular words are present in the sentence. The tokenizer divides large paragraphs into sentences, tokenization is done so that we can apply weight to each sentence. Once the sentence tokenization is done & weight is assigned to each sentence, a threshold value is set in the program that will be compared with the sentence's weight. If the weight of the sentence is greater than the threshold value then the summary is generated. After the summary gets generated, a text document is created in which summarized data is present. The user can download both the documents, summarized one as well as the entire version which contains Speech-To-Text of the entire video.

TECHNOLOGY STACK

Software
1. HTML
2. Bootstrap
3. NodeJS
4. Python

Hardware Requirements:
1. A decent processor (Intel /AMD)
2. Storage

OUTPUT



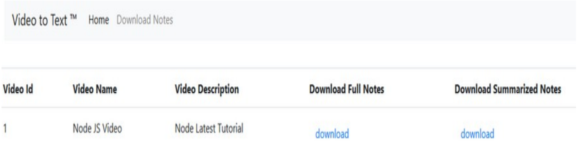
Select a File: Browse... No file selected.

Video Name:

Video Description:

Figure 4: *Upload Video Page*

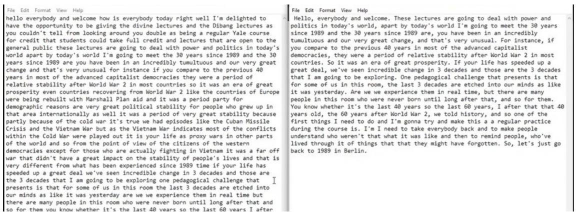
Here one can select the Video from which they want to generate notes, once they select the video, they can give a name and description to that video and after clicking on the Upload button the IBM Watson API is called and Voice to Text conversion is done and a text file is generated.



Video Id	Video Name	Video Description	Download Full Notes	Download Summarized Notes
1	Node JS Video	Node Latest Tutorial	download	download

Figure 5: *Download Page*

This is the download page from where the user can download the full text or the summarized version of the text of the video tutorial.

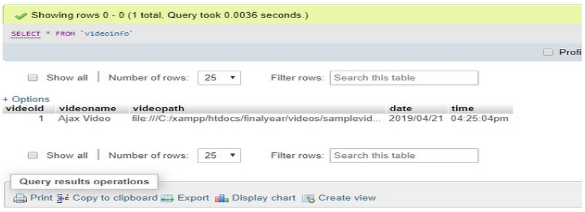


hello everybody and welcome how is everybody today right well I'm delighted to have the opportunity to be giving the divine lectures and the Dibang lectures as you couldn't tell from looking around you double as being a regular Yale course for credit that students could take full credit and lectures that are open to the general public these lectures are going to deal with power and politics in today's world apart by today's world I'm going to meet the 30 years since 1989 and the 30 years since 1989 are you have been in an incredibly tumultuous and our very great change and that's very unusual for instance if you compare to the previous 40 years in most of the advanced capitalist democracies they were a period of relative stability after World War 2 in most countries so it was an era of great prosperity even countries recovering from World War 2 like the countries of Europe were being rebuilt with Marshall Plan aid and it was a period party for demographic reasons are very great political stability for people who grew up in that area internationally as well it was a period of very great stability because partly because of the cold war it's true we had episodes like the Cuban Missile Crisis and the Vietnam War but as the Vietnam War indicates most of the conflicts within the Cold War were played out it is your life as proxy wars in other parts of the world and so from the point of view of the citizens of the western democracies except for those who are actually fighting in Vietnam it was a far off war that didn't have a great impact on the stability of people's lives and that is very different from what has been experienced since 1989 time if your life has speeded up a great deal we've seen incredible change in 3 decades and those are the 3 decades that I am going to be exploring one pedagogical challenge that presents is that for some of us in this room the last 3 decades are etched into our minds as like it was yesterday are we we experience them in real time but there are many people in this room who were never born until long after that and so for them you know whether it's the last 40 years so the last 60 years I after

Hello, everybody and welcome. These lectures are going to deal with power and politics in today's world, apart by today's world I'm going to meet the 30 years since 1989 and the 30 years since 1989 are, you have been in an incredibly tumultuous and our very great change, and that's very unusual. For instance, if you compare to the previous 40 years in most of the advanced capitalist democracies, they were a period of relative stability after World War 2 in most countries. So it was an era of great prosperity. If your life has speeded up a great deal, we've seen incredible change in 3 decades and those are the 3 decades that I am going to be exploring. One pedagogical challenge that presents is that for some of us in this room, the last 3 decades are etched into our minds as like it was yesterday. Are we we experience them in real time, but there are many people in this room who were never born until long after that, and so for them. You know whether it's the last 40 years so the last 60 years, I after that that 40 years old, the 60 years after World War 2, we told history, and so one of the first things I need to do and I'm gonna try and make this a a regular practice during the course is. I'm I need to take everybody back and to make people understand who weren't that what it was like and then to remind people, who've lived through it of things that that they might have forgotten. So, let's just go back to 1989 in Berlin.

Figure 6*:Non-Summarized vs Summarized version of notes.*

This is the comparison of non-summarized vs summarized version of one of the video that we tested after converting from voice to text.


✓ Showing rows 0 - 0 (1 total, Query took 0.0036 seconds.)

```
SELECT * FROM `videoinfo`
```

□ Show all | Number of rows: 25 ▾ Filter rows: Search this table

videoid	videoname	videopath	date	time
1	Ajax Video	file:///C:/xampp/htdocs/finalyear/videos/samplevid...	2019/04/21	04:25:04pm

□ Show all | Number of rows: 25 ▾ Filter rows: Search this table

***Figure** 7: Database*

We have used MySQL as our database for storing details such as ID, Name, Upload time etc. of the video that has been uploaded on the website.

VI. CONCLUSION

In this paper, we have discussed about text summarization and how one can implement text summarization using NLTK, how text sentence tokenization and word tokenization is useful for calculating important sentences and combine them to make a short version of text documentation.

FUTURE SCOPE

Currently, it is possible to summarize notes for tutorials that are delivered in the standard English language. The future scope is to summarize tutorials for Programming Languages which contains complex syntax i.e. difficult to convert from Speech-To-Text, documents that contain mathematical and scientific formulas.

ACKNOWLEDGMENT

We extend our sincere thanks to our project guide Assistant Prof. Savita Lohiya for her constant guidance throughout the project.

REFERENCES

Journal Paper:

[1]. Young, Tom, Devamanyu Hazarika, Soujanya Poria, and Erik Cambria. "Recent trends in deep learning based natural language processing." IEEE Computational

intelligenCe magazine 13, no. 3 (2018): 55-75.

[2]. Hahn, Udo, and Inderjeet Mani. "The challenges of automatic summarization." Computer 33, no. 11 (2000): 29-36.

[3]. Luhn, H. P. (1958). The automatic creation of literature abstracts. IBM Journal of Research Development, 2(2):159-165

[4]. Baxendale, P. (1958). Machine-made index for technical literature – an experiment. IBM Journal of Research Development, 2(4):354-361

[5]. Edmundson, Harold P. "New methods in automatic extracting." Journal of the ACM (JACM) 16, no. 2 (1969): 264-285.

[6]. Krishnaveni, P., and S. R. Balasundaram. "Automatic text summarization by local scoring and ranking for improving coherence." In 2017 International Conference on Computing Methodologies and Communication (ICCMC), pp. 59-64. IEEE, 2017.

[7]. Madhuri, J. N., and R. Ganesh Kumar. "Extractive Text Summarization Using Sentence Ranking." In 2019 International Conference on Data Science and Communication (IconDSC), pp. 1-3. IEEE, 2019.

Sunil Poojari, et. al. “Automatic Extraction of Text Notes from Video Tutorials.” *International Journal of Engineering Research and Applications (IJERA)*, vol.10 (08), 2020, pp 33-37.