Grenze International Journal of Engineering and Technology, June Issue

Smart Notes and Summary Maker from Lecture Videos

Sakshi Kadam[1], Prachi Ramane[2], Leena Ragha[3] and Shruti Raorane[4]
[1-4]Department of computer engineering, Ramrao Adik Institute of Technology, Navi Mumbai, India
Email: sakshikadam1520@gmail.com, preramane@gmail.com, leena.ragha@rait.ac.in, shrutiraorane2000@gmail.com

***Abstract*—The Pandemic has altered the educational landscape, causing teaching techniques to be optimized. Schools and educational institutions have been obliged to make the transition to the internet. A new normal has emerged, and it is being taught online. Video conferencing software is required for online teaching. Teachers can also use the live lectures to record them for later use. As a result, videos have evolved into a valuable source of knowledge. The video's text contains a large number of facts and information. However, this information is not editable. It becomes easier and more effective to keep useful information if this content is transformed into readable form. Furthermore, after watching an educational video, a user might not want to watch it again because he has already seen it, and reading the important points may be sufficient for him to review the topic. The system takes the time-consuming process of manually extracting text from videos and automates it.**

***Index Terms*— Smart Notes Maker, Text Identification, Extraction, Text Summarization, Abstractive Summary, Abstractive Summary.**

I. INTRODUCTION

The coronavirus illness of 2019 (COVID-19) has wreaked havoc on countries all over the world. The epidemic of COVID-19 has altered the educational landscape. Although online education has been promoted for many years, the pandemic has accelerated its adoption. Students at all schools and institutions have been compelled to go online because they were unable to attend classes' offline. Video conferencing software are required for online teaching. Teachers can also use these applications to capture live lectures for later use. As a result, videos have evolved into a valuable source of knowledge. These videos, however, are not modifiable. The recorded video can only be listened to; it cannot be edited or marked up with crucial points. In terms of gaining a better knowledge of the subject, part of taking good notes is making sure you absorb the information as soon as possible so that it can become more consolidated in your mind with repeated exposure. In light of this issue, we devised a method that allows students to concentrate on key themes in an editable manner and revise them at any moment.

This system's primary goal is to take notes from video lectures. These notes contain a summarised version of the text that appears in the video. This used to make it easier to find content with crucial points. Because video takes up a lot of memory to retain for later use, taking notes in editable text format saves a lot of space. Also, unlike video, the information in the text form can be modified in future if it changes later or if the user wants to add any additional information or notes to the text. Once a user has watched an instructive video, it is possible that he may not watch the entire video the following time and will only watch a portion of it. As a result, he may miss any crucial information. However, in the case of text notes, reviewing the essential points may be enough for the

Grenze ID: 01.GIJET.8.2.17

user to review the topic of videos. This technology has a lot of potential in the educational field.

A. Problem Statement

Text from video input is extracted and converted into legible and editable text documents. To take notes from video lectures by making a summary of the text taken from the video. This technique can be utilised by students in the educational sector to take notes more efficiently and quickly.

II. LITERATURE SURVEY

Text recognition from video or photos, as well as automatic text summarization, is still a hot topic in AI and machine learning. Many scholars have proposed numerous methods and approaches to handle text recognition challenges. Text recognition is a large AI and machine learning domain with many subdomains. Text extraction from videos or photos, image to text conversion, text recognition, and so on are some of these subdomains. All of these research areas must be thoroughly investigated. Because extractive summarizing techniques has become a wide study field, a complete review of research publications on text summarization is also necessary. In addition, the focus of study has switched to abstractive summarization. This is due to the fact that abstractive summaries are more complex and difficult to understand than extractive summaries.

Ref [1] The study by Jae-Chang Shim, et al. is analyzed and studied here. The method employed in this study creates segmented characters which can be immediately processed by an Optical Character Recognition to generate ASCII text. Experiments with approximately five thousand frames from twelve MPEG video streams reveal that the system is capable of recognizing text, although it fails. 'Text Extraction from Video Images' Using Discrete Cosine Transform, Fast Fourier Transform, and Random Forest is proposed by Nidhin Raju, et al. [2]. Text extraction from video frames of a news channel is presented in this research. Only Malayalam news video networks were examined in the projected effort. Scripts from a single Malayalam news channel called 'Mathrubhumi News' are used in the experiments. This aids in the extraction of key elements that are unique to Malayalam scripts, as well as improving accuracy and speed. We can search for multiple languages because it is only limited to Malayalam. Text Recognition, MESR, OCR, grey scale, and accuracy dependency on window dimensions were employed for 'Text recognition and extraction from video' [3]. The proposed approach for recognizing and extracting text from video is presented in this paper. The technology automates the manual process of extracting text from f, saving time and human effort in the process. The system is written in the MATLAB programming language. The method is mostly used for educational and news videos that feature textual information. The proposed approach could be improved by allowing the user to select a specific area of the screen and then extracting only the text that appears in that area. This will be beneficial in situations where the user just wants text from a specific region.

Pooja and Renu Dhir have published a work on text summarization called 'Video Text Extraction and Recognition' [4]. This paper conducts a survey of text extraction and text recognition from video input. It was challenging enough to extract frames from the videos. Text will be recovered from Punjabi text Line using word segmentation and character extraction. It was unable to build an approach for extracting text from video with a higher recognition rate because the resolution of video images is so low that detecting compound letters becomes increasingly challenging. As a result, that will be the primary focus of our paper.

III. PROPOSED WORK

As shown is Fig. 1, the proposed system can be divided into the components namely: a. Frame generation, b. Crop Image, c. Text recognition, d. Elimination of Repeated Text and e. Text summarization.

a. The frame generation module converts the input video into frames. Any image format can be used to save these frames.
b. Instance segmentation is used to detect the required area i.e., Find the regions that contain text.
c. Then, text is segmented in the regions where it is detected. For text recognition, the end result is commonly a binary image.
d. In Text recognition, text present in the video frames gets converted into ASCII characters.
e. Repeated text gets eliminated using Edit Distance in python.

At the end of this process, user will able to get notes from the videos in document format.

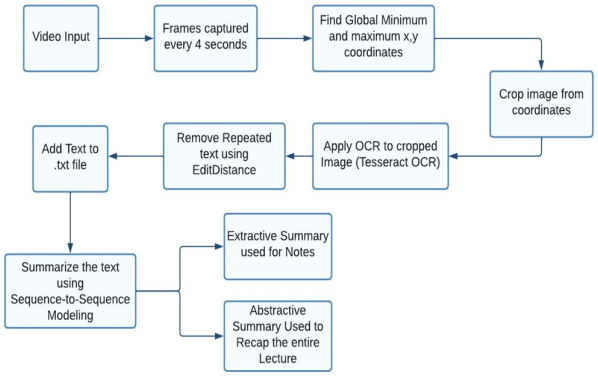

Figure 1. Flow diagram of the system

IV. METHODOLOGY

A. Text Recognition

OCR or Optical Character Recognition is used for text recognition inside images, like photos or scanned documents. It turns any image with written text into text data that may be edited. Tesseract OCR is a free and open-source OCR engine. It may be used to extract text from photos and is licenced under the Apache 2.0 licence. Tesseract uses wrappers to make it compatible with a variety of programming languages and frameworks. It may be used to read text from a huge document or in combination with an externally obtained text detector to recognise text from a single text line picture.

Tesseract 4.00 has a new neural network subsystem configured as a text line recognizer. Although Tesseract's neural network technology predates TensorFlow, it is compatible with it because TensorFlow uses the Variable Graph Specification Language to describe networks (VGSL). A Convolutional Neural Network (CNN) is used to identify a picture with a single character. The LSTM is a prominent type of RNN that is used to tackle issues like determining if a text of any length is a sequence of letters, etc.

The flow of the OCR is shown below in Fig. 2.

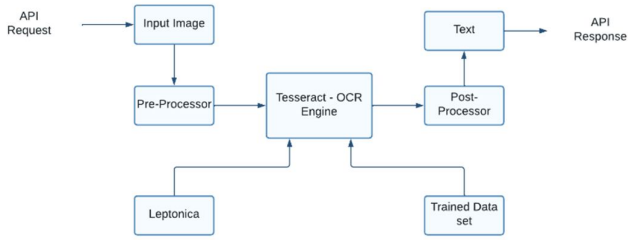

Figure 2. OCR Process Flow

B. Elimination of Repeated Text

The Edit Distance, often known as the Levenstein Distance, is a method of determining how unlike two strings are. Levenstein distance allows deletion, substitution and insertion. It is nothing but parameterizable metric. It helps in correcting OCR errors. Distance metric The Levenstein distance is calculated as follows using equation (1).

$$Lev(a,b) = \begin{cases} |a| \quad if\ |b| = 0 \\ |b| \quad if |a| = 0 \\ lev(tail(a), tail(b)) \quad If\ a|a| = b|0| \\ 1 + \min \begin{cases} lev(tail(a), b) \\ lev(a, tail(b) \\ lev(tail(a) \\ tail(b) \end{cases} \quad Otherwise \end{cases} \tag{1}$$

Where “tail” denotes the remainder of the sequence by excluding the 1st character.

Equation (1) can be explained as:

- Suppose, b is an empty string (|b|=0), then the cost is the length of a (|a|).
- Suppose, a is an empty string (|a|=0), then the cost is the length of b (|b|).
- If the first characters of strings a & b are the same, then the cost is the cost of substring tail(a) and tail(b).
- The ultimate cost is the minimum of insertion, deletion, or substitution operation. These operations are defined as:
- The deletion of a character from a is denoted by lev(tail(a), b).
- lev (a, tail(b)) denotes the addition of a character to a.
- The symbol lev(tail(a), tail(b)) denotes substitution.

C. Text Summarization

Text summarization is the process of shortening the lengthy documents, reducing the size of the text but at the same time preserving important information with the meaning of the text. Doing it manually can be expensive, difficult and time consuming. Solution to overcome this is automatic text summarization. Automatic text summarization can be accomplished in two ways. Abstractive and extractive text summarization are the two types of text summarization.

Abstractive Text summarization Approach

PEGASUS: It is an Encoder-Decoder Model which is used for sequence-to-sequence learning. In this, the encoder will first take into consideration the context of the whole input text. It is abbreviation for Pre-training with Extracted Gap-sentences for Abstractive Summarization Sequence-to-sequence models. Abstractive summarization uses the Pegasus model by Google [8]. Transformers Encoder-Decoder architecture is used in the model. PEGASUS trains a transformer encoder-decoder model using self-supervised objective Gap Sentences Generation (GSG). Important sentences in an input document are removed or masked in PEGASUS. This is then combined into a single output sequence.

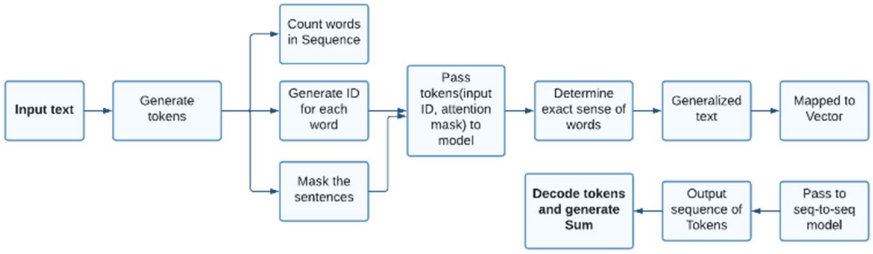

Figure 3. Flow diagram of abstractive summarization

Sequence-to-Sequence (Seq2Seq) Modelling: Seq2Seq is a machine translation and language processing system based on encoder-decoders.

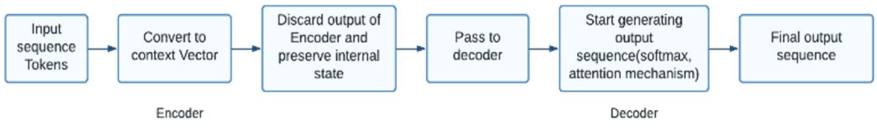

Figure 4. seq2seq model

It uses a tag and an attention value to map sequence input to sequence output. Any problem which involves sequential data can be solved using this approach. Few of the most frequent application of sequential information are Sentiment categorization, Neural Machine Translation, and Named Entity Recognition, etc.

Encoder-Decoder Architecture: The sequence-to-sequence (Seq2Seq) challenges are generally solved with the Encoder-Decoder architecture. Encoder and Decoder are the two essential components of a Seq2Seq model. The input and output sequences in this paradigm are of varied lengths. As encoder and decoder components, variations of Recurrent Neural Networks (RNNs), such as GRU - Gated Recurrent Neural Network or LSTM, are commonly used. This is due to the fact that these two components may capture long-term interdependence. Encoder-Decoder architecture may be divided into two phases: training and inference.

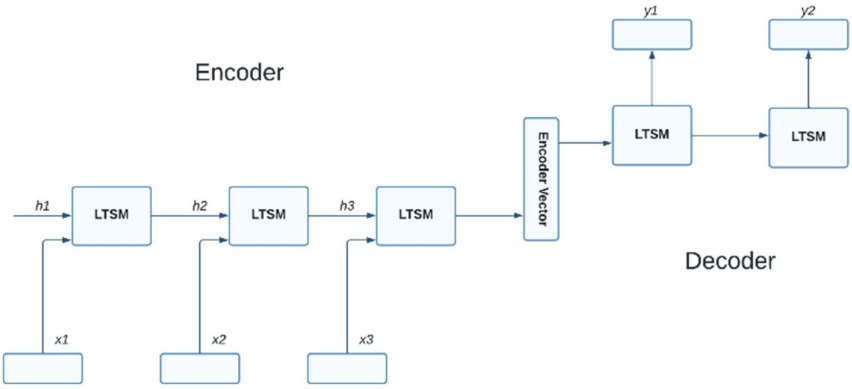

Figure 5. Encoder Decoder Architecture

Explanation of the fig. 3, fig. 4 and fig. 5 is as follows.

1. Pegasus model takes input text and generate tokens for the input. Token contains id of word and attention mask.
2. Then it counts words in each sentence, mask the sentence and pass tokens to model.
3. Model determines exact meaning of the word and generates generalized text.
4. After performing these steps, model map the generalized text to vector and pass this vector to seq2seq model.
5. Input to seq2seq model is sequence of tokens.
6. Encoder converts these tokens to context vector. The output of encoder state gets discarded. It preserves an internal state and passes internal states to the decoder.
7. Then the decoder learns to predict the summary. The task of decoder is to generate output sequence with the help of softmax activation (instead of picking single words as an output, multiple highly probable choices are retained) and attention mechanism.
8. The output of seq2seq model is output sequence of tokens. These tokens are not in human readable form.
9. Therefore, decoder decodes the tokens and generate final output summary which is in human readable form.

Extractive Text Summarization Approach

SpaCy is an advanced Natural Language Processing library in Python. SpaCy comes with pre-trained pipelines. For summarization SpaCy is used import a pre-trained NLP pipeline to help interpret the grammatical structure of the text. It helps to identify the common words that are often useful to filter out also the punctuation. It calculates the sum of normalized count for every sentence and the extract percentage of the most important sentences. This is served as the summary of the text.

V. Result and Analysis

Result

The result of the proposed system is shown below which contains three parts. The first part is shown in fig. 6. The video input is given to the system and the system gives output as text file. The output text file contains text which is present in the video frame. After OCR and Elimination of Repeated Text, the output is given to the text summarization part as an input and performed extractive and abstractive summarization on that text. Output of the both summarization approach are shown in the Fig. 7 (a) and (b).

Analysis of Keywords

The table I shows analysis of keywords. The first column contains keywords which are present in the output of text recognition part. As this text recognition part is an input to the summarization, extractive summarization approach contains some of the keywords and rest of the keywords are there in the abstractive summarization approach.

Information isn't worth much if it doesn't serve a purpose. MIS students learn how businesses use the information to improve the company's operations. Students also learn how to manage various information systems so that they best serve the needs of managers, staff and customers. MIS students learn how to create systems for finding and storing data, and they learn about computer databases, networks, computer security, and lots more. The component of an information system includes: a hardware which is used for input/output process and storage of data, software used to process data and also to instruct the hand-ware component, databases which is the location in the system where all the organization data will be automated and procedures which is a set of documents that explain the structure of that management information system.
There are various driving factors of management information system for example- Technological revolutions in all sectors make new managers need to have access to a large amount of particular information for the complex tasks and decisions. The lifespan of most product has continued getting shorter and shorter, and therefore the challenge to the manager is to design product that will take a longer shelf life, and in order to do this, the manager must be able to keep abreast of the factors that influence the organization product and services thus, management information system comes in handy in supporting the process.

Figure. 6. Extractive Summarization Output

The component of an information system includes: a hardware which is used for input/output process and storage of data, software used to process data and also to instruct the hand-ware component, databases which is the location in the system where all the organization data will be automated and procedures which is a set of documents that explain the structure of that management information system. The lifespan of most product has continued getting shorter and shorter, and therefore the challenge to the manager is to design product that will take a longer shelf life, and in order to do this, the manager must be able to keep abreast of the factors that influence the organization product and services thus, management information system comes in handy in supporting the process.

Management Information Systems (MIS) teaches students how to use information systems to improve the operations of businesses. The aim of this paper is to highlight the importance of management information system for new managers. A management information system helps the manager to keep up with the changing needs of the company.

Figure 7. (a) Extractive Summarization Output, (b) Abstractive Summarization Output

TABLE I. SUMMARIZED KEYWORDS

Text Recognition	Extractive Summarization	Abstractive Summarization
Management Information System, MIS, information, managers, networks, hardware, software, security, database, improve the operations.	Management Information System, information, managers, hardware, software, database	Management Information System, MIS, managers, improve the operations

VI. CONCLUSION

The system is to generate smart notes and summary from video lectures. The reason for this is that a written document requires a much smaller file size than a video, which saves memory and allows for faster access to information. As a result, the video's text is clipped and transformed to editable text format. The editable text format is then subjected to automatic text summarization. As a result, you will receive summary notes in editable text format as the final product. In the future work, the computation power used can be reduced by eliminating the repeated text. Also, the summarization quality can be improved by performing semantic analysis on the text received from OCR.

REFERENCES

[1] Jae-Chang Shim, C. Dorai and R. Bolle, "Automatic text extraction from video for content-based annotation and retrieval", Fourteenth International Conference on Pattern Recognition, 1998.

[2] Nidhin Raju, Anita H.B, "Text Extraction from Video Images", International Journal of Applied Engineering Research, 2017.

[3] Kiran Agre, Ankur Chheda, Sairaj Gaonkar, Mahendra Patil, "Text Recognition and Extraction from Video", International Journal of Engineering Research & Technology (IJERT) 2017.

[4] Pooja, Renu Dhir, "Video Text Extraction and Recognition: A Survey", IEEE International Conference on Wireless Communications Signal Processing and Networking (WISPNET) 2016.

[5] Ngo, CW., Chan, CK., "Video text detection and segmentation for optical character recognition", Multimedia Systems 10, 2005.

[6] Lienhart, R., Effelsberg, W., "Automatic text segmentation and text recognition for video indexing", Multimedia Systems 8, 2000.

[7] S.V. Rice, F.R. Jenkins, T.A. Nartker, "The Fourth Annual Test of OCR Accuracy", Technical Report 95-03, Information Science Research Institute, 1995.

[8] PEGASUS: Pre-training with Extracted Gap-sentences for Abstractive Summarization, [Submitted on 18 Dec 2019 (v1), last revised 10 Jul 2020 (this version, v3)]