ABSTRACT

In this new era, where tremendous information is available on the internet, it is most important to provide the improved mechanism to extract the information quickly and most efficiently . It is very difficult for human beings to manually extract the summary of a large documents of text. There are plenty of text material available on the internet. So there is a problem of searching for relevant documents from the number of documents available, and absorbing relevant information from it. In order to solve the above two problems, the automatic text summarization is very much necessary. This volume of text is an invaluable source of information and knowledge which needs to be effectively summarized to be useful.

Text summarization is the process of identifying the most important meaningful information in a document or set of related documents and compressing them into a shorter version preserving its overall meanings.

CHAPTER 1

INTRODUCTION

Before going to the Text summarization, first we, have to know that what a summary is. A summary is a text that is produced from one or more texts, that conveys important information in the original text, and it is of a shorter form. The goal of automatic text summarization is presenting the source text into a shorter version with semantics. The most important advantage of using a summary is, it reduces the reading time. Text Summarization methods can be classified into extractive and abstractive summarization. An extractive

summarization method consists of selecting important sentences, paragraphs, etc. from the original document and concatenating them into shorter form. Abstractive summarizations is an understanding of the main concepts in a document and then express those concepts in clear natural language. There are two different groups of text summarization: extractive and abstractive. Extractive summarization only represent the main idea of the text to the user. The typical length of this type of summarization is 5 to 10 percent of the main text. On the other hand, the abstractive summarization systems gives concise information of the main text. The length of abstractive summary is 20 to 30 percent of the main text .

1.1 PROBLEM STATEMENT:

The textual information on modern Internet age continues to grow. We have to summarize the text information for that information while maintaining the meaning. Text summation is a process that automatically generates natural language definitions from one entry it can issue a document while holding important points. This will help you get the information fast and easy. Majority of the work has traditionally focused on extractive approaches due to the easy of defining hard-coded rules to select important sentences than generate new ones. Also, it promises grammatically correct and coherent summary. But they often don’t summarize long and complex texts well as they are very restrictive.

1.2 OBJECTIVES OF THE PROPOSED PROJECT

Text summarization is the problem of creating a short, accurate, and fluent summary of a longer text document. The main objectives of the project is

- To improve the accuracy of the summary that is precise to the meaning of the actual text.
- To generate the context-based summary
- To reduce the human effort and reduce human errors

CHAPTER 2

SYSTEM ANALYSIS

FUNCTIONAL REQUIREMENTS

- User has to first convert the data into a .txt file. And then the .txt file should be uploaded to our tool/software.
- The file size should not exceed 32 MiB as modern text editors are not designed to handle such huge amount of data and may crash if we forcefully try to process the large data.

NON-FUNCTIONAL REQUIREMENTS

- User should have stable internet connection in order to upload the file without failing.
- The tool should be run on Google-Chrome browser to support the complete functionalities.
- The Google-Chrome browser and its plugins should all be updated to the latest versions.

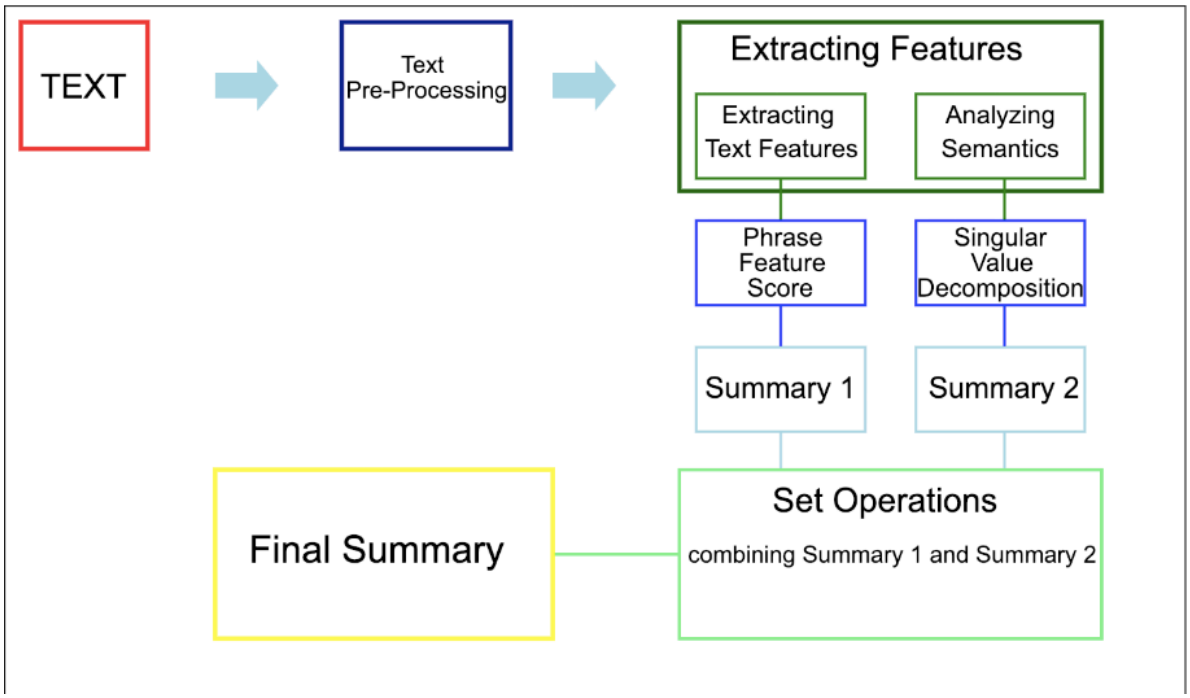
MODULES CHOSEN

- Modules chosen are numpy, pandas ,nltk ,tensorflow,keras,sklearn, and other basic ones.
- Using ‘pip' installer or use the python condo environment to do the same ,pandas and numpy will be used for the handling Data-sets and Scientific computations .
- It will also make it easier to run the bulletins for different tasks.
- nltk module will help us for the Text Rank Algorithm .
- It has various methods for doing tasks like sentence tokenization and many more.

CHAPTER 3

SYSTEM DESIGN

3.1 ARCHITECTURAL DESIGN



Here the Text is pre-processed to extract features ,further it extracts text features and analyse semantics which further they extract phrase feature score and singular value decomposition to make a summary .

The Original text is first pre-processed into word segmentation, morphological and co-reference reduction into processed text. Further the processed text is processed into phrase acquisition ,phrase refinement and phrase combination , so basically after the set operations performed to make a summary from text extraction and analysing semantics.

Both summaries are then combined together for set operations which is processed further to get the final summary.

3.2 BEHAVIOURAL DESIGN

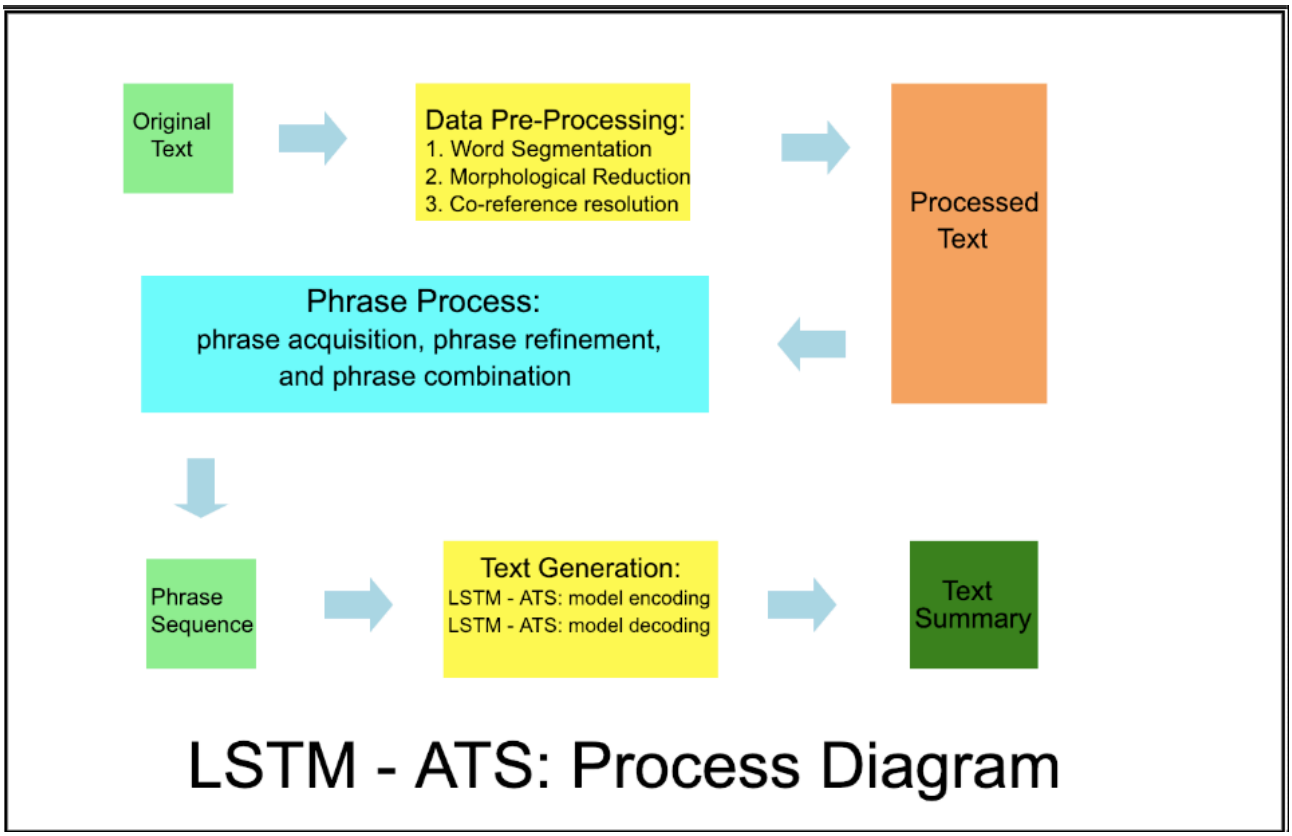


Fig. LSTM Design Model

Here the Original text is first pre-processed into word segmentation, morphological and co-reference reduction into processed text. Further the processed text is processed into phrase acquisition ,phrase refinement and phrase combination.

The phrase sequence is processed to text generation by using LSTM-ATS model encoding and decoding to obtain the most stable text summary.

It is a unit of recurrent neural network (RNN) which is composed of long term short memory(LSTM).An LSTM is composed of a cell which remembers values over arbitrary time intervals and gates of input , output and forget which regulate the flow of information into and out of the cell.

LSTM Networks are suited for classifying, processing and making predictions based on time series data.It deals with exploding and vanishing gradient problems which are encountered during the process of RNN.The main benefit of LSTM over RNN is relative insensitivity to gap length .

The peephole LSTM or the peephole connections allow the gates to access the constant error carousel (CEC), whose activation is the state of the cell.

The total error of LSTM's on a training set can be minimised by using optimisation algorithms such as like gradient descent algorithm and back propagation algorithm to compute gradients so as to change the weight of each LSTM network with respect to corresponding weight.

The main problem of using these algorithm is that error gradient vanishes quickly with size of time lag because of Weight limit n to infinity becomes zero when spectral radius is smaller than one.

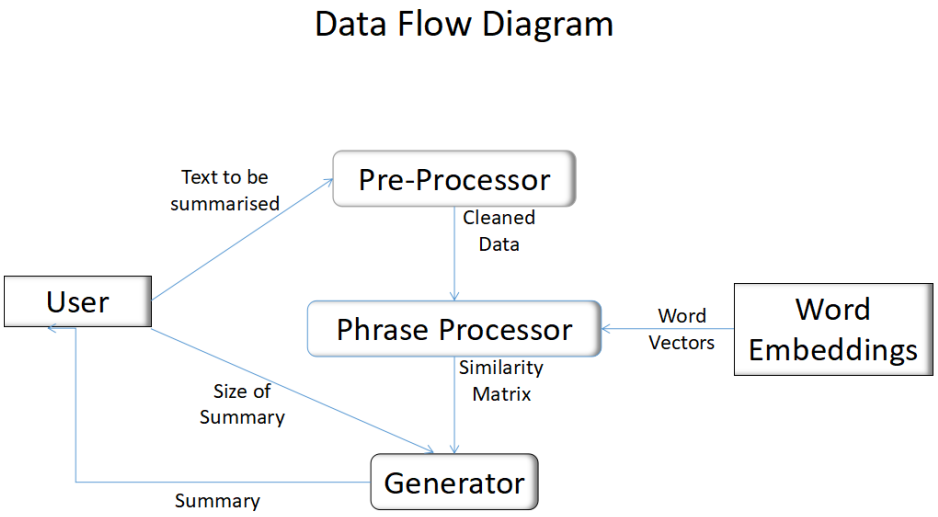
The error remains same in the LSTM cell that is the error carousel gives error to each of the LSTM gates until cut off value is obtained.

3.3 SYSTEM APPLICATIONS OF LSTM

The main applications of LSTM are,

- Recognition of Human action
- Translation of Sign Language
- Semantic parsing networks
- Prediction of Time series
- Recognition of Speech
- Music Composition
- Recognition of Handwriting
- Control mechanism of Robot
- Medical Care Predictions

3.4 DATA FLOW DIAGRAM



TextRank Algorithm uses the structure of the text and the known parts of speech for words to assign a score to words that are keywords for the text. It gives more value to nodes with bunch of connections, and gives more influence for better connected nodes, so it reinforces itself to get stable score.

First, the words are assigned parts of speech, so that only nouns and adjectives (or some other combination for different applications) are considered. Then a graph of words is created. The words are the nodes/vertices (denoted V). Each word is connected to other words that are close to it in the text. In the graph, this is represented by the connections on the graph (denoted E).

After the combination of words(articles) to form text ,it then splits to form the sentences form Vectors to which they are symbolised to Similarity matrix and then the algorithm is then run on the graph. Each node is given a weight of 1. Then the algorithm goes through the list of nodes and collects the influence of each of its inbound connections.

The value of the connected vertex (initially 1, varies periodically) and then combined to determine the new score for the node. Then these scores are normalised, the highest score becomes 1, and the rest are scaled from 0 to 1 based on that value. Each time through the algorithm gets closer to the actual value for each node, and it repeats until the values stop changing.

REFERENCES

[1] M. Esther Hannah, Dr. Saswati Mukherjee, K. Ganesh Kumar. "An Extractive Text Summarization Based on Multivariate Approach", 2010 3rd International Conference on Advanced Computer Theory and Engineering (ICACTE)

[2] Mahsa Afsharizadeh, Hossein Ebrahimpour-Komleh, Ayoub Bagheri. "Query-oriented Text Summarization using Sentence Extraction Technique", 2018 4th International Conference on Web Research (ICWR)

[3] Siya Sadashiv Naik, Manisha Naik Gaonkar. "Extractive Text Summarization By Feature-Based Sentence Extraction Using Rule-Based Concept", 2017 2nd IEEE International Conference On Recent Trends in Electronics Information & Communication Technology (RTEICT)

[4] Aditya Jain, Divij Bhatia, Manish K Thakur. "Extractive Text Summarization using Word Vector Embedding", 2017 International Conference on Machine Learning and Data Science