VISVESVARAYA TECHNOLOGICAL UNIVERSITY

BELAGAVI - 590018

Project Report

on

“Text Summarization using NLP”

Submitted in partial fulfilment of the requirements for the VIII Semester

Bachelor of Engineering

in

COMPUTER SCIENCE AND ENGINEERING

For the Academic Year

2020-2021

BY

Rishikesh Patil	1PE16CS198
Akanksha Makkar	1PE17CS011
B T Amith Kumar	1PE17CS033
Pratheek G	1PE17CS111

UNDER THE GUIDANCE OF

Prof. Nivedita Kasturi

Assistant Professor, Dept. of CSE, PESIT-BSC

PES
INSTITUTIONS

Department of Computer Science and Engineering

PESIT - BANGALORE SOUTH CAMPUS

Hosur Road, Bengaluru - 560100

PESIT - BANGALORE SOUTH CAMPUS

HOSUR ROAD, BENGALURU - 560100

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the project work entitled “**Text Summarization using NLP**” carried out by “***Rishikesh Patil, Akanksha Makkar, B T Amith Kumar, Pratheek G*** bearing USN’s ***1PE16CS198, 1PE17CS011 , 1PE17CS033, 1PE17CS111***” respectively in partial fulfillment for the award of Degree of Bachelors (Bachelors of Engineering) in Computer Science and Engineering of Visvesvaraya Technological University, Belagavi during the year 2020-2021. It is certified that all corrections/ suggestions indicated for internal assessment have been incorporated in the report. The project report has been approved as it satisfies the academic requirements in respect of project work prescribed for the said Degree.

Signature of the Guide	Signature of the HOD	Signature of the Principal
Prof. Nivedita Kasturi	**Dr. Annapurna D.**	**Dr. Subhash Kulkarni**
Asst Professor, CSE	**HOD, CSE**	**Principal, PESIT-BSC**

External Viva

Name of the Examiners **Signature with Date**

1.

2.

DECLARATION

We, Rishikesh Patil (1PE16CS198), Akansha Makkar (1PE17CS011), B T Amith Kumar (1PE17CS033), and G Prateek (1PE17CS111) hereby declare that the dissertation entitled, **‘Text Summarization using NLP’**, is an original work done by us under the guidance of Nivedita Kasturi, Assistant Professor, Department of Computer Science and Engineering, PESIT-BSC, Bengaluru, is being submitted in partial fulfillment for the award of B.E. in Computer Science and Engineering, VTU of the requirements for 8th semester.

Date: Signature of the candidates

Place: Bangalore

Rishikesh Patil Rishi

Akanksha Makkar Akanksha

B T Amith Kumar B.T. Amith Kumar

Pratheek G

Acknowledgements

The satisfaction and euphoria that accompany the successful completion of our task would be incomplete without the mention people who made it possible, whose constant guidance and encouragement crowned our effort with success.

We are indebted to our guide, Prof. Nivedita Kasturi, Assistant Professor, Department of Computer Science and Engineering PESIT Bangalore South Campus, who has not only coordinated our work but also given suggestions from time to time.

We are grateful to Dr. Annapurna D, Professor and HOD of Computer Science and Engineering Department who has seen me doing well in the project work.

We would like to express our immense gratitude to Dr. Subhash Kulkarni, Principal, PESIT Bangalore South Campus, for providing with excellent infrastructure.

We gratefully acknowledge the help lent out to me by all faculty members of the Department of Computer Science and Engineering. PESIT Bangalore South Campus, at difficult times. We would also take this opportunity to thank our college management for the facilities provided during the course of the project.

Date: :

Place: Bangalore

ABSTRACT

Summarization is the task of condensing a piece of text to a shorter version, reducing the size of the initial text while at the same time preserving key informational elements and the meaning of content. Since manual text summarization is a time expensive and generally laborious task, the automatization of the task is gaining increasing popularity and therefore constitutes a strong motivation for academic research.In the big data era, there has been an explosion in the amount of text data from a variety of sources. This volume of text is an inestimable source of information which needs to be effectively summarized to be useful. This increasing availability of documents has demanded exhaustive research in the NLP area for automatic text summarization.

Keywords:

Text Summarization, big data, Natural Language Processing(NLP),
Neural Networks

Contents

List of Figures

Chapter 1

Introduction

1.1 Introduction

1.1.1 Purpose

- Abstractive Text Summarizers have been used extensively by researchers and developers but, has the following limitations:
- Training of the Text Summarizer is extremely complex, and is still a problem for all models present.
- At present, the existing summarizers lack accuracy and efficiency to convert the meaning of the sentences
- Our work aims to improve Abstractive Text Summarization methods and provide an accurate summarization.

1.1.2 Scope

In recent times text summarization has gained its importance due to the data overflowing on the web. This information overload increases in great demand for more capable and dynamic text summarizers. It finds the importance because of its variety of applications like summaries of newspaper articles, book, magazine, stories on the same topic, event, scientific paper, weather forecast, stock market, News, resume, books, music, plays, film and speech

1.1.3 Definitions, Acronyms, Abbreviations

1.1.3.1 Definitions

- ROUGE (Recall-Oriented Understanding for Gisting Evaluation):- ROUGE is basically a recall- oriented measure that works by comparing the number of machine-generated words that are a part of the reference sentence with respect to the total number of words in the reference sentence.

  This metric is more intuitive in the sense that every time we add a reference to the pool, we expand our space of alternating summaries. Hence, this metricshould be preferred when we have multiple references.
- Due to this reason, we chose not to go for BLEU score evaluation.

  We can directly use the ROUGE-N implementation using the python library ROUGE.

  Latent Semantic Indexing is also implemented through this library.
- Tokenizer :- A tokenizer builds the vocabulary and converts a word sequence to an integer sequence.
- LSTM:- Long short-term memory is an artificial recurrent neural network architecture used in the field of deep learning. Unlike standard feedforward neural networks, LSTM has feedback connections. It can not only process single data points, but also entire sequences of data.
- Seq2Seq:- Seq2Seq is a method of encoder-decoder based machine translation that maps an input of sequence to an output of sequence with a tag and attention value. The idea is to use 2 RNN that will work together with a special token and trying to predict the next state sequence from the previous sequence.

  •NER model:- Stanford NER is a Named Entity Recognizer implemented in Java. It provides a default trained model for recognizing chiefly entities like Organization, Person and Location. Apart from this, various models trained for different languages and circumstances are also available.

1.1.3.2 Acronyms/ Abbreviations

NER model:- Stanford NER is a Named Entity Recognizer.

LSTM:- Long short-term memory.

ROUGE (Recall-Oriented Understanding for Gisting Evaluation **)**

BERT:-Bidirectional Encoder Representations from Transformer

API: Application Programming Interface

OS: Operating System

1.2 Literature Survey

- Ankit Kumar, Zixin Luo, Ming Xu: Text Summarization using Natural Language Processing Expands on the sentence extraction technique by implementing a more generalised technique
- Josef Steinberger, KarelJežek: Using Latent Semantic Analysis in Text Summarization and Summary Evaluation Paper on semantic analysis for text summarization which also proposes evaluation methods for summary accuracy.
- Sunitha C., Dr. A. Jaya, Amal Ganesh: A Study on Abstractive Summarization Semantic graph technology is used, studies on summaries based on indian languages
- Jen-Yuan Yeh, HaoRenKe, Wei- Pang Yang, I- HengMeng: Text summarization using a trainable summarizer and latent semantic analysis.

1.3 Existing System

An Encoder Long Short Term Memory model (**LSTM**) reads the entire input sequence wherein, at each timestep, one word is fed into the encoder. It then processes the information at every timestep and captures the contextual information present in the input sequence, which has its drawbacks like LSTMs are inclined to overfitting and it is hard to apply the dropout calculation to control this issue. Dropout is a regularization strategy where input and intermittent associations with LSTM units are probabilistically prohibited from actuation and weight refreshes while preparing an organization.

1.4 Proposed System

In the proposed system. BERT (Bidirectional transformer) is a transformer used to conquer the impediments of RNN and other neural organizations as Long haul conditions. It is a pre-prepared model that is normally bidirectional. This pre-prepared model can be tuned to effectively play out the NLP errands as determined, Synopsis for our situation.

1.5 Statement of Problem and objective

"*I don't want a full report, just give me a summary of the results", this* is the most common situation faced by everyone. Instead of reading the whole document people are interested in the summary which saves a lot of time.

Manually converting the report to a summarized version is time consuming, so we came up with a solution to summarize large reports using Natural Language Processing(NLP) and Recurrent Neural Networks (RNN).

1.6 Summary

This chapter gives us the detailed description of the user interaction with the system. To describe this we have used interaction models such as use case diagrams, data flow diagrams and activity diagrams. It also provides the overall system architecture along with the assumptions to be made while developing the system and its constraints.

Chapter 2

Hardware and Software Requirements Specification

2.1 Software Requirements Specifications

Programming prerequisites details joins the important necessities for a specific task or framework. The best objective of having these prerequisites is to notice the normal outcomes and necessities prior to beginning with the execution. This segment manages the different sorts of programming, equipment and other useful and non-useful prerequisites of the undertaking. The applications, benefits and the clients for this venture are likewise referenced.

2.2 Operating Environment

This section gives a brief about the hardware and software prerequisites for the project.

2.2.1 Hardware Requirements

- 1.6GHz or faster processor
- Minimum of 8 GB RAM
- Minimum 250GB of available hard disk space

2.2.2 Software Requirements

- IDE-Visual Studio Code- Overall development and User Interface
- Python IDLE- Prediction model development
- Jupyter Notebooks and Colab for implementation of model
- NER model
- DistilBERT API
- Python Libraries: Pandas, Numpy, Seaborn, Matplotlib, Tensorflow,
- ROGUE-N

2.3 Functional Requirements

Functional requirements are a formal way of expressing the idea behavior of a project or the system. Functional Requirements for our project are as follows:

- The framework ought to give text parser capacities which can take the entire content and separate into sentences, passages and words.
- The framework ought to give text ¬to¬ include work which can take the essential part and get a include vector.
- The framework ought to give a well¬ prepared Auto encoder to produce better contributions for classifier.
- The framework needs a classifier which is well¬ prepared to choose synopsis sentences.

2.4 Non-Functional Requirements

Nonfunctional requirements are the various capabilities offered by the system. They do not bother about the expected results but focus about the efficiency and how well the result has been achieved.

- Usability- The User Interface has to be simple enough to understand with proper documentation
- Performance- The time for preprocessing and extracting the summary should be low
- Reliability:-This software will be developed with machine learning, feature engineering and deep learning techniques. So, in this step there is no certain reliable percentage that is measurable.
- Accuracy:- The System should be accurate in the translation of meaning of the sentence from source document to the Summarized document

2.5 User Types

The users associated with the system are as follows:

Text synopsis can be utilized by close to home or concentrated partners, that would be a definitive use case. Aside from that it tends to be utilized for some customized gadgets or applications. Mail customers, report age, news source and so on It could happen that you would take a gander at an article or blog and your collaborator could give you a synopsis of the blog.

2.6 Application of System

Media monitoring:- The issue of data over-burden and "content shock" has been generally talked about. Programmed rundown presents a chance to gather the constant downpour of data into more modest snippets of data.

Newsletters;- Numerous week after week bulletins appear as a presentation followed by a curated choice of significant articles. Outline would permit associations to additionally improve bulletins with a surge of rundowns (versus a rundown of connections), which can be an especially advantageous organization in portable.

Monetary examination:-Venture banking firms go through a lot of cash procuring data to drive their dynamic, including mechanized stock exchanging. At the point when you are a monetary expert seeing business sector reports and news ordinary, you will unavoidably reach a stopping point and will not have the option to understand everything. Outline frameworks custom fitted to monetary records like acquiring reports and monetary news can assist investigators with getting market signals from content.

2.7 Advantages of System

- Summing up lessens scrutinizing time
- While exploring reports, traces make the assurance technique less complex
- Outline works on the ampleness of requesting
- Outline computations are less uneven than human summarizers
- Customized rundowns are helpful being referred to noting frameworks as they give customized data
- Using modified or Outline systems engage business hypothetical organizations to assemble the quantity of content chronicles they can measure

2.8 Summary

This part talks about the different prerequisites – utilitarian and non-practical, that are found in building this task. Least equipment that is expected to fabricate the model and the product including outsider administrations is likewise been examined. Aside from this, it additionally examines who utilizes our framework, and the pertinent utilization of this framework. It finishes by expressing the benefits of this framework and why it is liked over other existing systems.

Chapter 3

High Level Design

The architecture that will be utilised to construct this project is described in high-level design (HLD). The architecture diagram shows the overall system and identifies the primary components that will be built as well as their interfaces. The HLD employs non-technical to somewhat technical terminology that are easily understood by the user. Low-level design, on the other hand, shows programmers and developers using our API the logical precise design of each of these pieces. A high-level architectural diagram describing the system's major components and interfaces is generally included in high-level design.

3.1 Block Diagram

Before attempting to construct a complete design solution, this section discusses the difficulties that must be addressed or overcome. Preprocessing entails taking the EDA of the dataset/source document and breaking it down to identify the individual segments that will be provided to the model; Analysis of the Semantic Model — the NER model recognises repetitive vocabulary and allocates it to an n-dimensional matrix that the encoder can accept; Map of Text Relationships Construction — the NER model's hashed data and the n-dimensional matrix are used to find patterns in the text, which the LSTM model then uses to produce predictions at each step and draw connections between sentences. Finally, a summary document is created from the data summaries obtained for each sentence.

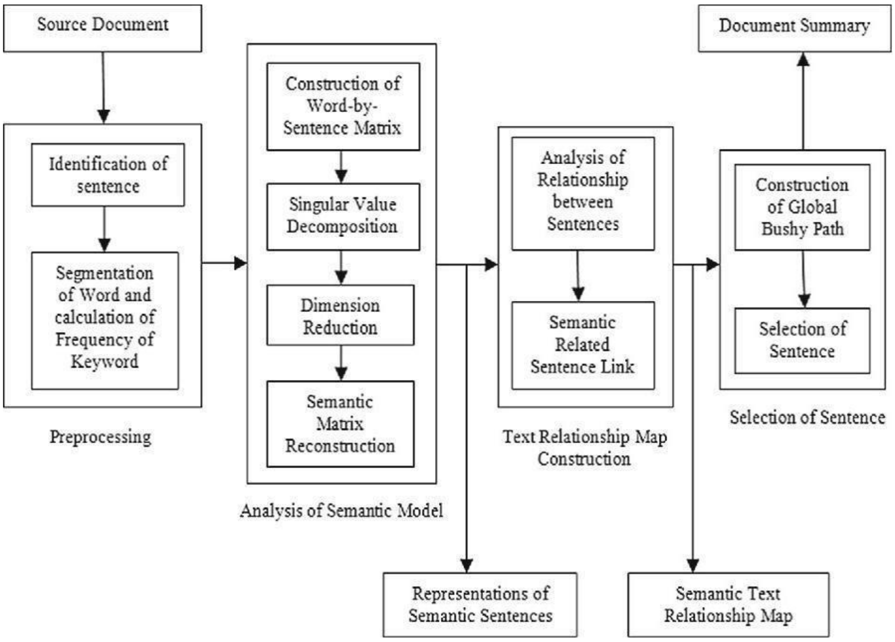

Source Document
Identification of sentence
Segmentation of Word and calculation of Frequency of Keyword
Preprocessing
Construction of Word-by-Sentence Matrix
Singular Value Decomposition
Dimension Reduction
Semantic Matrix Reconstruction
Analysis of Semantic Model
Analysis of Relationship between Sentences
Semantic Related Sentence Link
Text Relationship Map Construction
Document Summary
Construction of Global Bushy Path
Selection of Sentence
Selection of Sentence
Representations of Semantic Sentences
Semantic Text Relationship Map

Figure 3.1: Block diagram

3.2 Activity Diagram

An activity diagram is a flowchart that depicts the movement of information from one action to the next. The action can be described as a system operation. The primary goal of activity diagrams is to depict the system's dynamic behaviour. Object-oriented flowchart is another name for it. To verify and assess the model's correctness, the activity diagram focuses on the activities done on the document set, which is separated into training and test data.The model takes the training data and generates an abstractive summary of each individual phrase, which is compared to the model's ideal summary. These assertions are all integrated, and the semantic meaning of each one is determined at each stage to help the predictions at the next. Finally, the summary of the paragraph or extract is checked to the test results to confirm correctness. These are all integrated to provide a comprehensive document summary.

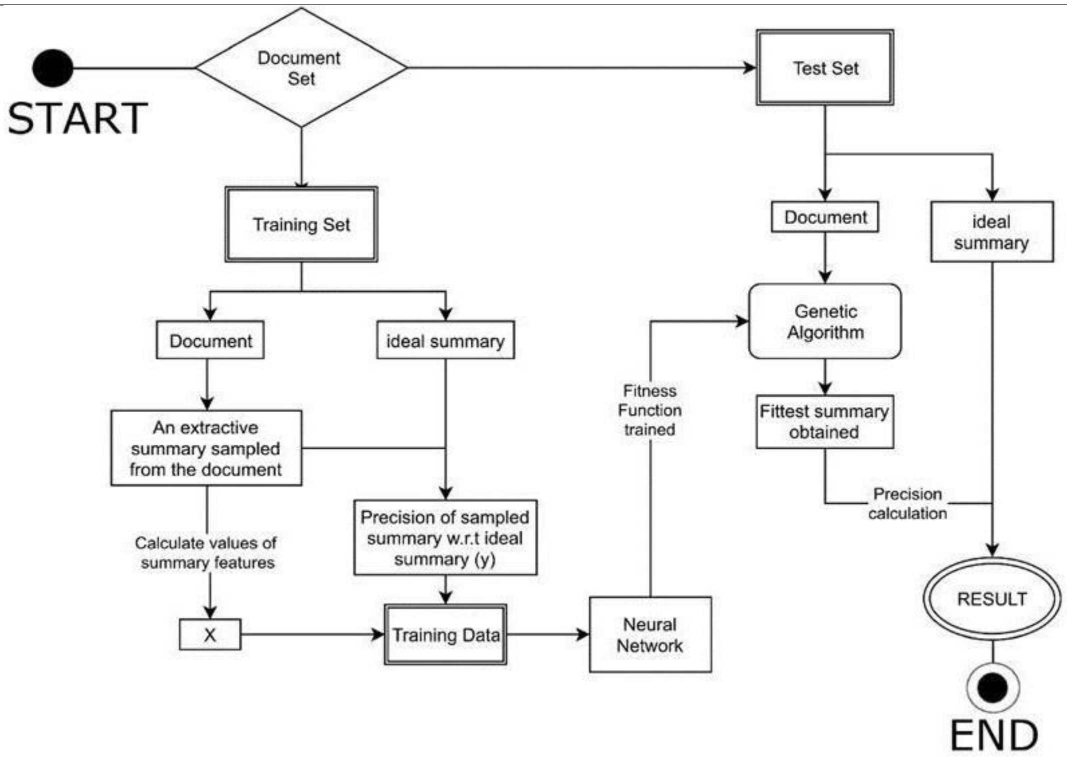


Figure 3.2: Activity Diagram

3.3 Use case diagram

At its most basic level, a use case diagram depicts a user's engagement with the system by illustrating the relationship between the user and the many use cases in which the user is involved. The users and the summarization model are the actors in this issue statement. The different input documents given by the user, as well as the extracts presented, are the various use cases involved in the same. It is necessary to choose between abstractive and extractive summarising techniques. Individual lines play a critical part in determining the following phrase prediction in the document, which is read in segments. The summarizer model takes care of gathering all of the data and showing it as a final summary document, which is then given to the user for further usage.

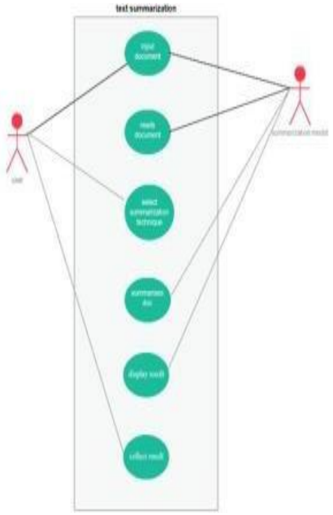


Figure 3.3: Use case diagram

3.4 System Architecture

A sequence diagram simply illustrates the order in which things interact, or the order in which these interactions occur. A sequence diagram can also be referred to as an event diagram or an event scenario. Sequence diagrams show how and in what sequence the components of a system work together. The model trainer completes the training process by accepting the dataset for summary as input. The next word to be predicted is completed and sent to the predictor, who then saves it to assure the segment's correctness. The evaluator uses the ROGUE python package to assess the aggregate summary for semantic correction and prediction accuracy. The needed modifications are broadcast over the network so that the model may fix them, and the process continues until the entire document has been processed.

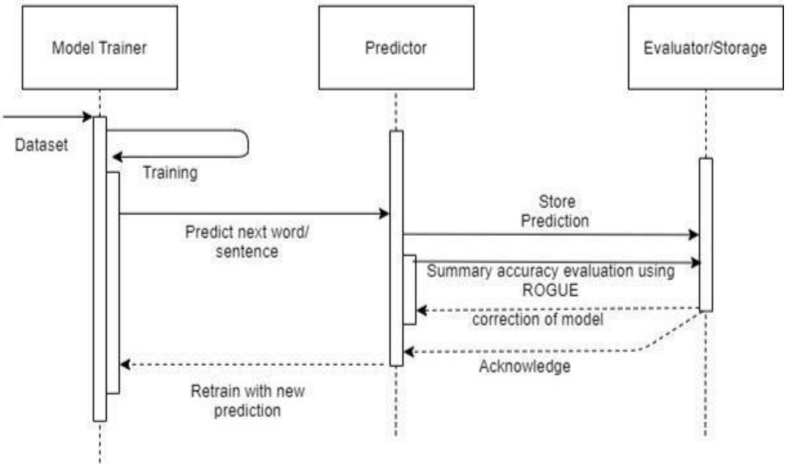


Figure 3.4: Sequence diagram

3.5 State transition diagram

State-transition diagrams show all of the states that an item may have, as well as the events that cause it to change state (transitions), the criteria that must be met before the transition can take place (guards), and the activities that occur over the object's lifetime (actions). State-transition diagrams are extremely effective for explaining the behaviour of particular objects over a broad range of use scenarios. State-transition diagrams are ineffective at portraying the collaboration between the items that produce transitions.

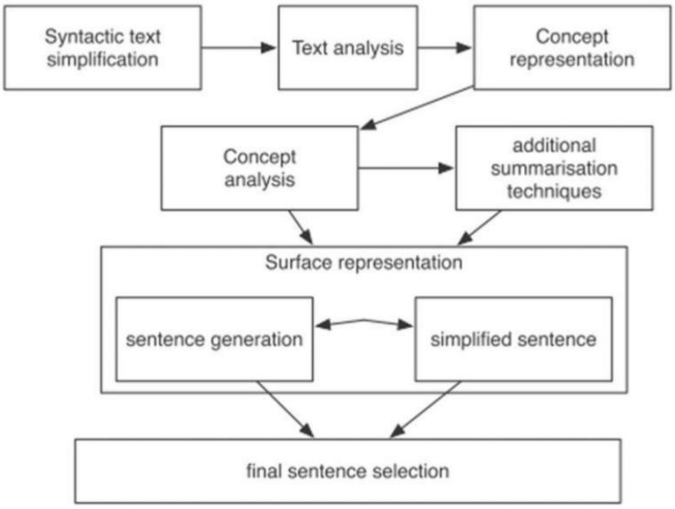


Figure 3.5: State transition diagram

3.6 Summary

This chapter describes the user's interaction with the system in great detail. We utilised interaction models including use case diagrams, data flow diagrams, and activity diagrams to express this. It also includes the general system design, as well as the assumptions and restrictions to be made when creating the system.

Chapter 4

Detailed Design

4.1Purpose

The purpose of this chapter is having an understanding of the two approaches used to solve this problem, draw a comparison between them and finally conclude which is the better one.

4.2Module 1: Data Acquisition and Preprocessing

4.2.1 LSTM Dataset

The dataset used to train and develop the first model was Amazon Fine Food Reviews dataset that was taken from Kaggle, it comprises of revies about food items written in short descriptions of about 100-300 words. The data obtained has been processed and changed to lowercase, remove the HTML tags, Contraction Mapping, removal of text inside parenthesis, elimination of punctuations and stop words. The dataset comprises of user details, the review written and the summary of that text. The summary is relatively short – less than 100 words.

The dataset contains 568,454 food reviews Amazon users left from October 1999 to October 2012.

4.2.2 BERT Dataset

The dataset used for training in the second phase of implementation was the Spacy dataset, which was taken from the Spacy – python’s inbuilt library itself. It comprised of text and their summaries taken from various articles, blogs and text excerpts. The text was varying from a few ten lines to hundreds. The summary was accordingly scaled down to a few tens of lines. The default model is small English spaCy model (en_core_web_sm, 11Mb) and is installed automaticaly with this package. The Neuralcoref library was used along with Spacy for preprocessing. The dataset consists of some unwanted things, need to remove all the unwanted symbols, characters, etc. From the text that do not affect the objective of our problem. The noise by using different preprocessing techniques. After removing the unwanted things using nltk, re and bs4 python libraries, final preprocessed data is obtained.

4.3Module 2: Model Training

4.3.1 LSTM Model Training

We initially implemented the LSTM model for an extractive summary of the amazon food reviews dataset. This data was passed through a word to index mapping segment and followed by the word embeddings to an n-dim matrix, which were done using an NER model. The approach used to build and train this LSTM model is Seq2Seqwhich comprises of an encoder and decoder. The Seq2Seq model is able to identify the previous relations between the embedded words and apply the same while predicting the next set of appropriate words in the summary. Keras was used to build the models and train through about 50 epochs. A single headed attention layer was used to extract the contextual embeddings of words in the text.

4.3.2 BERT Model Training

The next phase of implementation was for the larger dataset – the articles and blogs from Spacy. This dataset then went through tokenization of the input words. Next, the tokenized inputs were passed through the 2 phases on the BERT model – pretraining. This process is done to aid the BERT model to understand the language. Fine tune is the next phase which is done to learn a specific task - summarize the text. In order to aid the understanding of the language and context, the positional embeddings and contextual embeddings are important and they are done using the Masked Language modelling phase of the Pretraining process. The next sentence prediction follows it in order to ensure that context is maintained across 2 consecutive sentences. The model carries out both left-to right and right to left training to include the context of the text. This has been done on Google Colaboratory, which is a platform for training machine learning models. It provides us with free GPU for our model training.

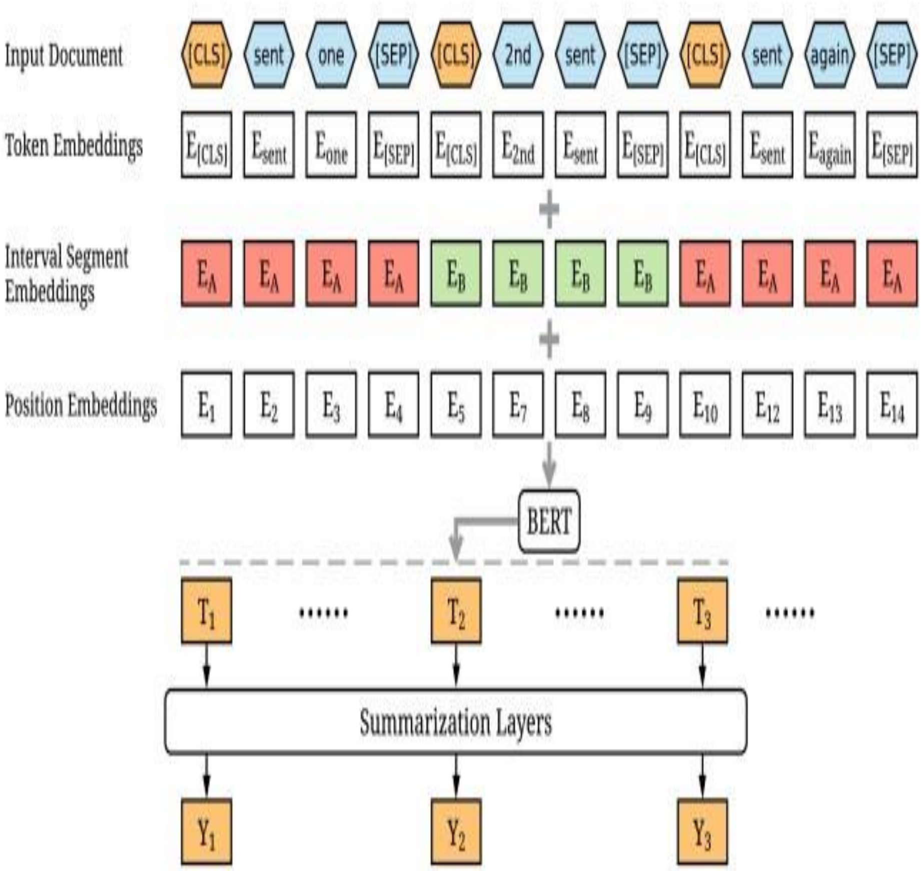

Input Document
[CLS]
sent
one
[SEP]
[CLS]
2nd
sent
[SEP]
[CLS]
sent
again
[SEP]
Token Embeddings
$E_{[CLS]}$ E_{sent} E_{one} $E_{[SEP]}$ $E_{[CLS]}$ E_{2nd} E_{sent} $E_{[SEP]}$ $E_{[CLS]}$ E_{sent} E_{again} $E_{[SEP]}$
Interval Segment Embeddings
E_A E_A E_A E_A E_B E_B E_B E_B E_A E_A E_A E_A
Position Embeddings
E_1 E_2 E_3 E_4 E_5 E_7 E_8 E_9 E_{10} E_{12} E_{13} E_{14}
BERT
T_1 T_2 T_3
Summarization Layers
Y_1 Y_2 Y_3

Fig 4.1 BERT Model

4.4 Module 3: LSTM and its limitations

The LSTM approach was initially utilized as they are easy to train and effective with summarizing smaller pieces of text. However, they are extremely slow to train. It could take a few hours to train the model even for smaller texts and this increases exponentially with larger datasets. They are not truly bidirectional as they can cannot retain the contextual meaning very well. This is so as they do not implement any complex mechanisms for such a comparison, they simply concatenate the previous layers' outcome with the current one. Further, they face the problem of vanishing gradients with long sequences. Although we were able to train smaller datasets well, several issues were faced with larger, thus not aligning with our initial goals.

4.5 Module 4: Transformer Neural Networks

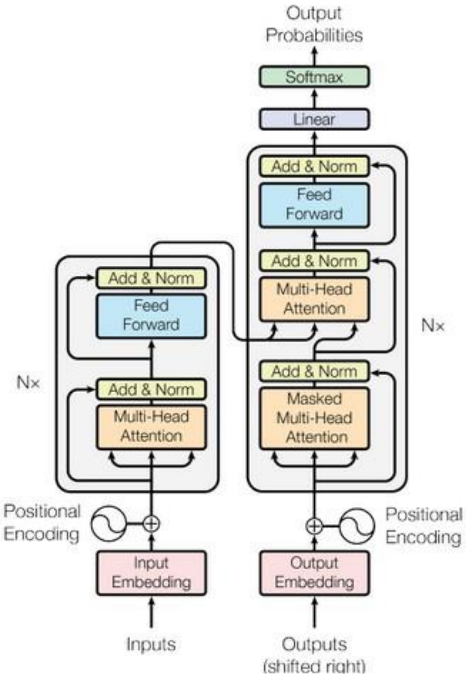


Fig 4.2 Transformer Neural

To overcome the limitations of the LSTM model, a TNN (Transformer Neural Network) was used. These work well with larger datasets and are able to work well for the task at hand – extractive summary of blogs, articles and reports. A transformer is a deep learning model that adopts the mechanism of attention, weighing the influence of different parts of the input data. This focuses primarily on the attention layers for contextual meaning retention. Multiple attention heads are used, due to this, a GPU is required for training and parallel processing of the sentences.

The key components of this neural network are: Encoder: this encoder is both a combination of the embedding vector as well as the positional encoder. This effective in providing an embedding of the words with contextual data. Multi-headed Attention: Can give significance to multiple parts of the embedded sentence at a time and its parallel functionality improves the performance as well. Generates information as a combination of attention vectors and blocks. Feed forward NN: used to transform them into vectors that can be processed by the encoder and decoder block. Batch norm and Layer norm: These are used after ever stage to normalize the features and smoothen them so that it's easier to optimize using the learning rates. These are deeply bidirectional and are thus able to capture the contextual, semantic and syntactic meaning of the text.

BERT's key specialized development is applying the bidirectional preparing of Transformer, a mainstream consideration model, to language demonstrating. This is rather than past endeavors which took a gander at a book arrangement either from left to right or joined left- to-right and right-to-left preparing. The paper's outcomes show that a language model which is bidirectionally prepared can have a more profound feeling of language setting and stream than single-heading language models.

Instead of directional models, which read the content information successively (left-to-right or right-to-left), the Transformer encoder peruses the whole arrangement of words on the double. Along these lines it is considered bidirectional, however it would be more precise to say that it's non-directional. This trademark permits the model to gain proficiency with the setting of a word dependent on the entirety of its environmental elements (left and right of the word).

4.6 Module 5: Attention Layer

The attention mechanism is used when the input or the context vector passed from the encoder to the decoder is too large to generate an accurate output. -The input is divided into several parts and then fed in such cases.

The main intuition of using attention is to allow the model to focus/pay attention on the most important part of the input text.

Helps to avoid the vanishing gradient problem as well

Different types of attention mechanisms -we use dot product here

In this architecture, instead of directly using the output of last encoder’s hidden state, we are also feeding the weighted combination of all the encoder hidden states. This helps the model to pay attention to important words across long sequences.

Algorithm:

1. ***Attention scores*** *are first calculated by computing the dot product of the encoder(h) and decoder(s) hidden state*

2. *These attention scores are converted to a* ***distribution(α)*** *by passing them through the Softmax layer.*

3. *Then the* ***weighted sum of the hidden states (z)*** *is computed.*

4. *This z is then concatenated with s and fed through the softmax layer to generate the words using **'Greedy Algorithm'** (by computing argmax)*

4.7 Module 7: Comparison Between LSTM and BERT

Through comparison of the 2 models implemented, few observations were made. It was clear that BERT not only performed well with respect to the time taken for training, but also performed better for summarization of the data. The positional and contextual embeddings of the data done by the model enabled more accurate summaries to be generated. This model could also work better with larger datasets, unlike LSTM. It also proved to avoid the vanishing gradient problem for long sequences observed on LSTM model and was able to retain the memory for longer durations. Dur to the presence of the multi headed attention layers – focus on the most significant words was given to the text and model generated summaries that were much more semantically, syntactically and contextually sound. Finally, the model was proven to have a true bidirectional nature in comparison to the LSTM model.

4.8 Module 8: Model Evaluation

Predictive accuracy and F1 score were all measured for the model validation and evaluation process.

4.9 Summary

This chapter presents a detailed overview of the various modules of the system.

Chapter 5

Implementation

The implementation phase of the development life cycle is crucial. This phase involves converting system needs and specifications into a functioning model that can be used to provide real-time services. The design stage's key functionalities are translated into functions that may be executed using appropriate programming languages.

As a result, the implementation phase is always preceded by critical decisions such as language selection, platform selection, and so on. Various variables influence these decisions, including the desired response time, security issues, and data management considerations, among others. These choices have an impact on how well the finished product works.

5.1 Programming Language Selection

In the implementation phase, the programming language plays an important role. The programming language chosen should be based on the requirements at hand. Given the simplicity of the syntax and the ease with which applications can be constructed using Python, java was an obvious choice for our project. Python programming is ideal for machine learning applications since it has a large library of libraries that make duplicate work much easier. Python was used to build the backend of the project, which included the logic for the predictive tool. To deliver a user-friendly experience, the front end is required. This was built with flask frame work using HTML and CSS for the development of the web page.

5.1.1 Overview of Python

Python is dynamic, object oriented and multipurpose programming language. It has a huge and inclusive standard library. It helps its user express concepts in a fewer lines of code which is not possible for languages such as in C++ and java. It is extended to create WEB applications as well. Due the project we're working on is primarily about machine learning and data analysis, python is one of the greatest options we have because of its dynamic nature.

FEATURES OF PYTHON:

- Since python has few keywords, easy syntax and simple structure so it is learnt easily.
- This language is portable as it can run in many hardware platforms.
- The code in python is easily maintained.
- Graphical user interface [GUI] is supported by python.
- It supports dynamic type checking.

5.1.2 Overview of HTML

Html is hypertext markup language. It is language that helps in creation of web pages in the internet. We have used HTML and CSS language to create GUI for user interactions, where users input the values for specified features and obtain their results.

FEATURES OF HTML:

- HTML is highly flexible.
- HTML is supported on every browser and it is user friendly.
- HTML coding is easily understandable.
- It can use wide ranges of colors, objects and layouts.

5.1.3 Overview of CSS

CSS is known as cascading style sheets. They create a uniform look across several pages of a website. It is used to describe how html elements are to be displayed on the screen. It plays a key role in the web designing.

FEATURES OF CSS:

- CSS separates content and presentation in the HTML.

- Since the complexity in the web pages is reduced the pages are loaded with faster speed.
- There is flexibility in CSS code once it is written can be applied to several other groups of HTML too.

5.2 PLATFORM SELECTION

Our project was developed in Windows 10. This OS was our choice given that our data needed a lot of fine tuning and visualizations which could be easily provided in the MS Excel software. Windows provide GUI for its computers. Windows minimizes the need for commands to operate OS by simply implementing mouse which navigates through menus, tabs, dialog boxes and icons. Windows was given this name due to its dynamic nature to allow multiple tasks to run at the same time. We have chosen this platform since its user friendly, since our project required more of analytics rather than focus on a very powerful development environment.

5.3 GRAPHICAL USER INTERFACE

The success of any project depends on how well it serves the end users. We require a friendly user interface including text boxes, buttons, labels etc. The web page is developed in HTML and styles with CSS. Users are provided with a text box where they can enter the original text which has to be summarized.

5.3.1 Overview of Flask

Flask is a web framework coded in python it uses template jinja2. Flask is used to build WEB applications such as web pages, blogs, or a website. Our first choice of frame work was flask given its syntax that is very similar to python itself, and the ease to learn the framework. Flask has it dependencies as follows:

- WSGI which is a utility library.
- Jinja2 which is template engine.

5.4 PYTHON LIBRARIES

- **Scikit-learn:** scikit learn is a wide python library which practices machine learning, cross validation of data, and preprocessing of data. It is built on NumPy and SciPy.

- **Pandas:** Pandas is a python package. It provides a designed data structure with fast expressive and flexible features. It makes the data work easier and intuitive. For many different kinds of data pandas is well suited such as tabular, matrix, ordered or unordered data.

- **Numpy:** Numpy's main objective is an multidimensional array, it holds an array data structure. NumPy is a core python library which contains a collection of tools and techniques. One of these tools is multi dimensional object.

- **Tensorflow:** Tensorflow can be used across different tasks but focuses on training and inference of deep neural networks. It is a math library based on dataflow and differentiable programming.

- **Nltk:** Nltk is Natural Language Toolkit, is a set of libraries and programs for statistical natural language processing for English written in Python.

    - **Spacy:** It supports deep learning workflow in neural networks in parts of speech, dependency parsing, named entity recognition.

5.5 CODING CONVENTIONS

The developed product may not be maintained by the developer throughout its lifetime. Thus it is highly important to make the code comprehensible to any new reader as well. Making the code more readable helps in changing and maintaining the code even in the future. Besides writing the code in python that is relatively easy to understand, two other practices are also employed.

- **Naming variables and constants:** The naming is done such that the name indicates the

variable's/ constant's obvious meaning without too much ambiguity. However some of the features used in the dataset are medical terms which may require a little bit of background study. Example height indicates the height of the pregnant woman.

Comments: Appropriate comments are provided in the code to make the confusing parts of code easy to understand. Comments are provided as complete sentences to avoid any sort of ambiguity.

5.6 IMPLEMENTATION STEPS

1. Used food review dataset collected from kaggle for abstractive text summarization using LSTM model.

2. Define a module for loading the .csv file. Check for rows with a few missing values and fill them with the mode value of each feature. Divide the dataset into training data and testing data.

3. Build the LSTM model using three stacked LSTM including encoder and decoder. Using this model we train the data of train dataset. Later in this model can be used for testing data.
4. For better results we used BERT model for extractive text summarization. Spacy library was used for data preprocessing such as tokenization.

5. advantage of BERT model is that its fast compared to LSTM as multiple words can be sent to the input layer so the time taken for training is quick compared to LSTM model where one word at a time is sent to the input layer.

6. Using Flask library a rest API is created which is used for sending data for text summarization. POST method is used for adding the original text by the user in the API call.

7. Develop a graphical user interface to enhance the experience of the end users in obtaining the results.

8. The user interface contains two web pages, the home page consists of an empty text box where a user can add original text. After clicking the submit button the result page appears which consists of the summarized text.

5.7 SUMMARY

This chapter covered the many strategies utilized in the project's development, beginning with language and platform selection and ending with an explanation of the whole process of implementation processes.

Chapter 6

Testing

6.1 Software testing

Before deploying any piece of software for use, testing is an important aspect of the software development cycle. Through the testing process, we can verify and validate that the system is working according to the requirements specification and is meeting the highest quality standards. If any problems are detected during this stage, the system is worked upon for improvement. The process of assessing the system as well as each of its sub-components with the objective of finding any errors/bugs is called software testing.

Testing can be a costly process in terms of time, hence a good way to reduce this cost is to test the system in parallel to its development. The idea is to break down the system into independent parts and iteratively create each part and test it to ensure it is performing the associated task. Essentially, it can be done in every stage of the life cycle with the aim of refining the stage.

Three major phases of testing include:

- Debugging
- Verification
- Validation

6.2 Levels of Testing

The typical testing process involves two main stages:

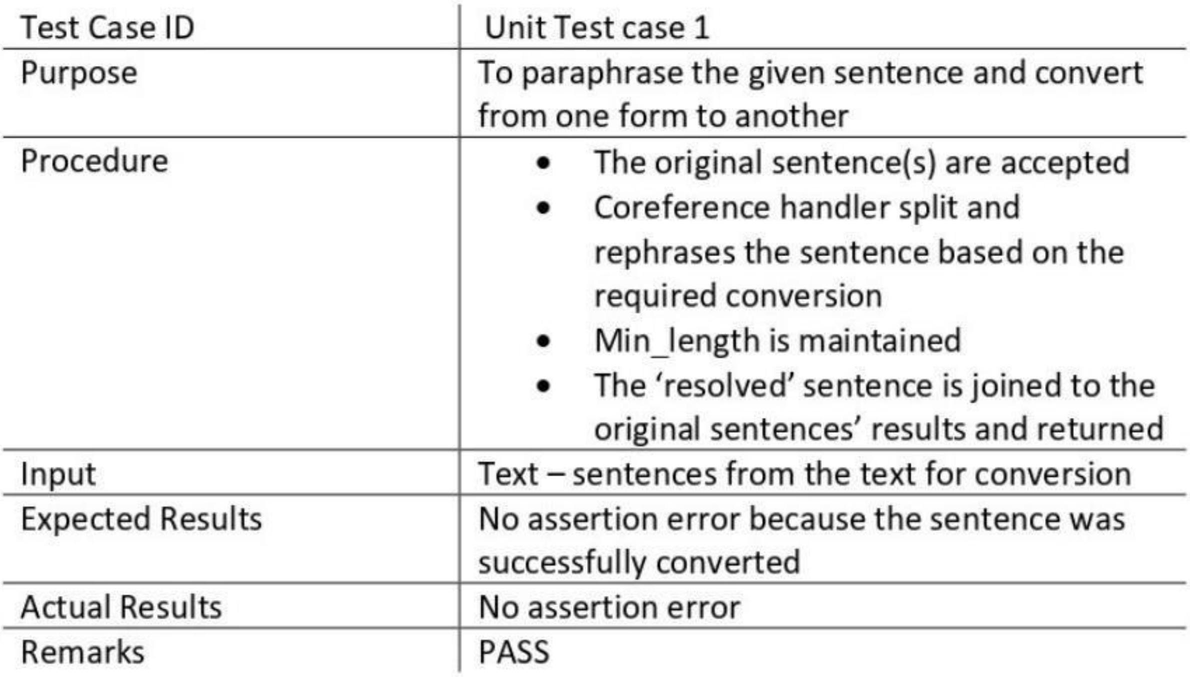
1. **Unit testing**: In this form of testing, each unit/function/component of the sys- tem is isolated and tested to ensure it works as desired. This is performed in parallel to the development of the component to prevent failure of the system as a whole when these multiple components are integrated. In our project, we per- formed unit testing on the custom simulation environment, i.e., the

functions of the Ad class and also the functions of the AdServerEnv class.

2. **Integration testing**: While in unit testing, we test individual components, in integration testing, we put these developed components together and test how they work with each other in conjunction. In this case, the system is tested as a whole. During our project, integration between the agents and the environment was tested to ensure the right type of data is flowing between the functions resulting in optimal outputs.

6.3 Unit Testing of Main Modules

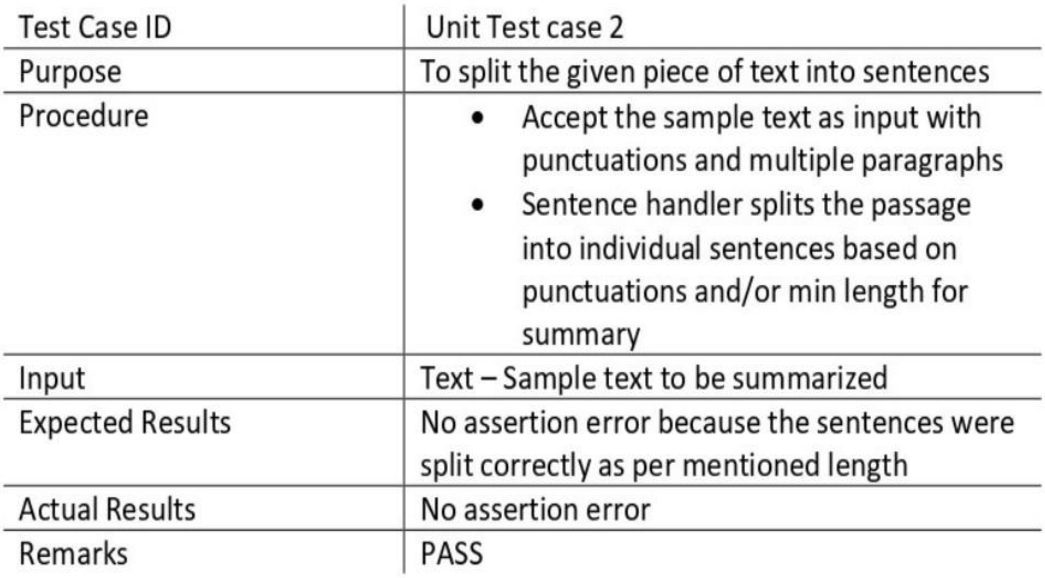
6.3.1 Testing the Coreference handler



Test Case ID	Unit Test case 1
Purpose	To paraphrase the given sentence and convert from one form to another
Procedure	• The original sentence(s) are accepted • Coreference handler split and rephrases the sentence based on the required conversion • Min_length is maintained • The 'resolved' sentence is joined to the original sentences' results and returned
Input	Text – sentences from the text for conversion
Expected Results	No assertion error because the sentence was successfully converted
Actual Results	No assertion error
Remarks	PASS

Figure 6.1: Unit test case 1

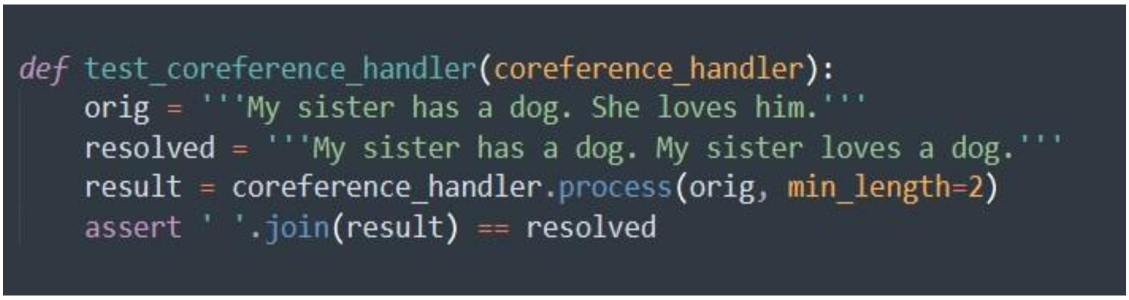
6.3.2 Testing the Sentence handler



Test Case ID	Unit Test case 2
Purpose	To split the given piece of text into sentences
Procedure	• Accept the sample text as input with punctuations and multiple paragraphs • Sentence handler splits the passage into individual sentences based on punctuations and/or min length for summary
Input	Text – Sample text to be summarized
Expected Results	No assertion error because the sentences were split correctly as per mentioned length
Actual Results	No assertion error
Remarks	PASS

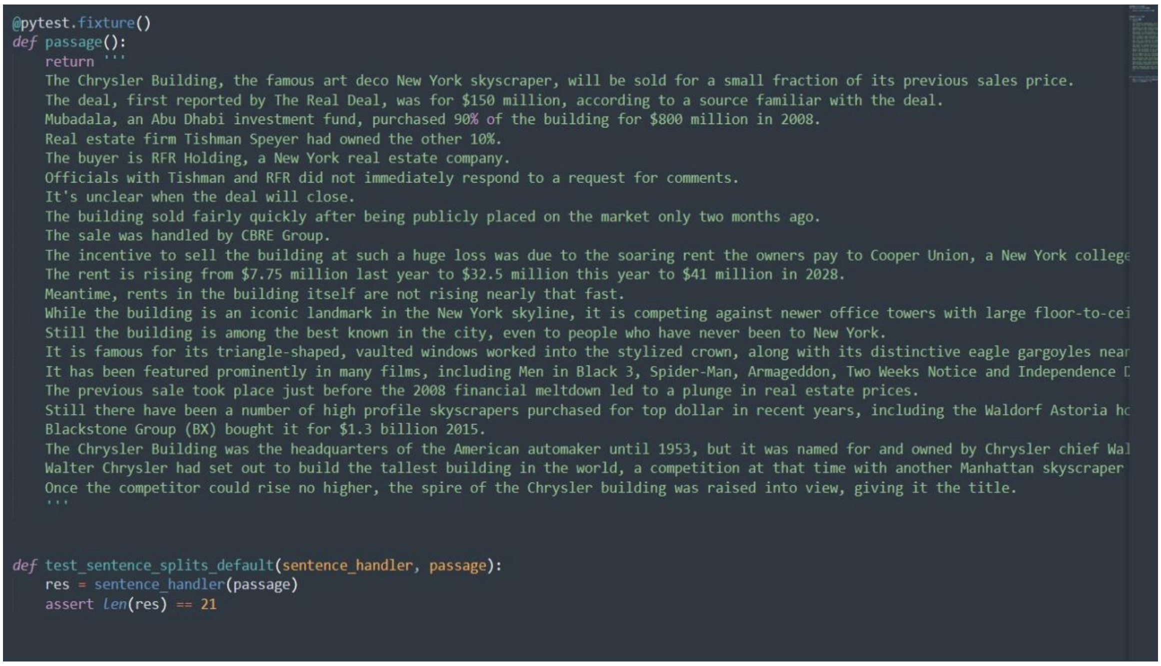
Figure 6.2: Unit test case 1

6.4 Testing Snapshots



```
def test_coreference_handler(coreference_handler):
    orig = '''My sister has a dog. She loves him.'''
    resolved = '''My sister has a dog. My sister loves a dog.'''
    result = coreference_handler.process(orig, min_length=2)
    assert ' '.join(result) == resolved
```

Figure 6.3: Function for test case 1



```
@pytest.fixture()
def passage():
    return '''
    The Chrysler Building, the famous art deco New York skyscraper, will be sold for a small fraction of its previous sales price.
    The deal, first reported by The Real Deal, was for $150 million, according to a source familiar with the deal.
    Mubadala, an Abu Dhabi investment fund, purchased 90% of the building for $800 million in 2008.
    Real estate firm Tishman Speyer had owned the other 10%.
    The buyer is RFR Holding, a New York real estate company.
    Officials with Tishman and RFR did not immediately respond to a request for comments.
    It's unclear when the deal will close.
    The building sold fairly quickly after being publicly placed on the market only two months ago.
    The sale was handled by CBRE Group.
    The incentive to sell the building at such a huge loss was due to the soaring rent the owners pay to Cooper Union, a New York college
    The rent is rising from $7.75 million last year to $32.5 million this year to $41 million in 2028.
    Meantime, rents in the building itself are not rising nearly that fast.
    While the building is an iconic landmark in the New York skyline, it is competing against newer office towers with large floor-to-cei
    Still the building is among the best known in the city, even to people who have never been to New York.
    It is famous for its triangle-shaped, vaulted windows worked into the stylized crown, along with its distinctive eagle gargoyles near
    It has been featured prominently in many films, including Men in Black 3, Spider-Man, Armageddon, Two Weeks Notice and Independence
    The previous sale took place just before the 2008 financial meltdown led to a plunge in real estate prices.
    Still there have been a number of high profile skyscrapers purchased for top dollar in recent years, including the Waldorf Astoria ho
    Blackstone Group (BX) bought it for $1.3 billion 2015.
    The Chrysler Building was the headquarters of the American automaker until 1953, but it was named for and owned by Chrysler chief Wal
    Walter Chrysler had set out to build the tallest building in the world, a competition at that time with another Manhattan skyscraper
    Once the competitor could rise no higher, the spire of the Chrysler building was raised into view, giving it the title.
    '''


def test_sentence_splits_default(sentence_handler, passage):
    res = sentence_handler(passage)
    assert len(res) == 21
```

Figure 6.4: Function for test case: 2

6.5 Summary

This chapter is concerned with the levels of testing performed in all the modules/sub- components of the system as well as the testing of the project as a whole. The project is observed to suitably satisfy the requirements. The different testing levels performed have also been explained briefly.

Chapter 7

Results

Two approaches used for text summarization are abstractive method using LSTM model and extractive method using BERT model. It is observed that BERT is a better model compared to LSTM as it is faster in training and execution. User interface consists of two web pages, home page where a user can enter the original text and the result page where the summarized result is obtained. For LSTM we used Jupyter notebook for execution where food review dataset was used. In BERT model any paragraph can be used as the original text which can be summarized so that the content is reduced to 60% of the original content.

7.1 Results of original text converted to summarized text

Original Text

While most definitions of health focus on the fact that how weak or strong one’s body physically is and how immune they are to illnesses and injuries, a new conversation has been stirring about the wellbeing of your mental health. If we go through the pages of human history, the greatest assets identified for all individuals are good health and a sound mind. What is noteworthy is the strong conjunction and interdependence these two variables have on each other.

One who does not have good mental health fails to maintain good physical strength and stamina. Only a stress-free headspace and the right frame of mind can lead to strong physical health. This combination makes it possible for us to see and feel the true eternal happiness which is intrinsic and arises from within. The endless and tiring pursuit of happiness often makes one forget that they also have to take care of their physical as well as mental health.

As kids, we’re often told that maintaining a good lifestyle can lead to the best of health.

But only as we grow older, do we realise that introspection, self-realisation and reflection of one’s own thoughts is also a crucial aspect of maintaining good mental health which perhaps leads to a fit and fine body.The lack of this approach often leads to this gloomy thought process at times, where one fails to understand what is troubling them and what particularly are they lacking.

Maintaining good health hence becomes vital for the overall core development of our personality and perspective towards life.

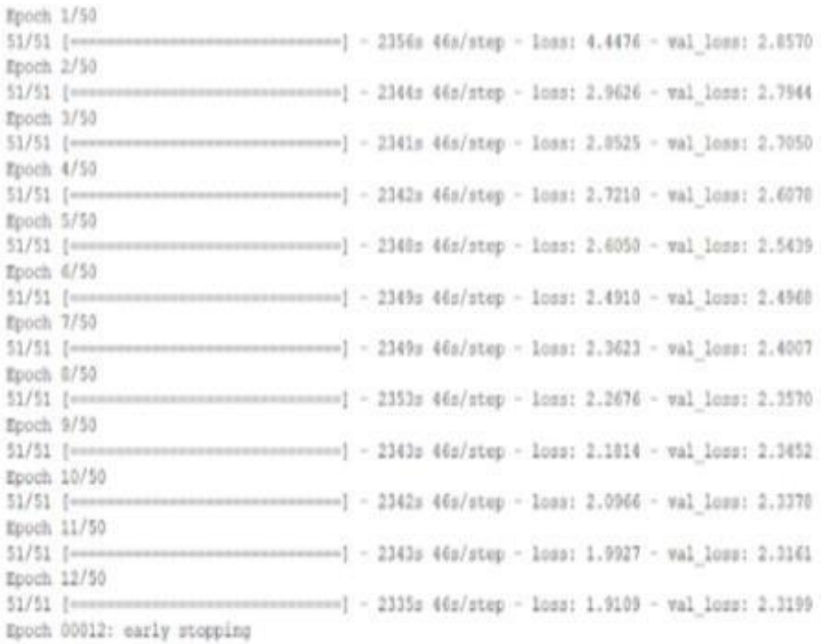
Summarized Text

While most definitions of health focus on the fact that how weak or strong one’s body physically is and how immune they are to illnesses and injuries, a new conversation has been stirring about the wellbeing of your mental health. If we go through the pages of human history, the greatest assets identified for all individuals are good health and a sound mind. What is noteworthy is the strong conjunction and interdependence these two variables have on each other. One who does not have good mental health fails to maintain good physical strength and stamina. This combination makes it possible for us to see and feel the true eternal happiness which is intrinsic and arises from within.

7.2 Snapshots :

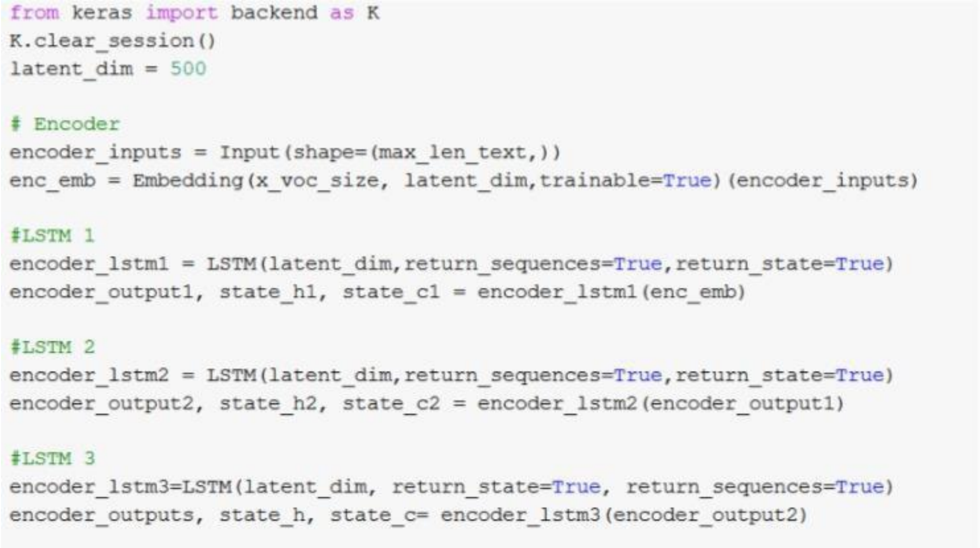
Results for LSTM:

We used lstm first But moved on to bert because it was better for larger datasets.



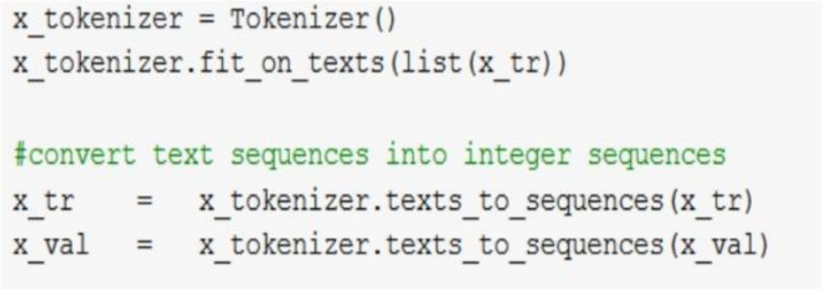
```
Epoch 1/50
51/51 [==============================] - 2356s 46s/step - loss: 4.4476 - val_loss: 2.8570
Epoch 2/50
51/51 [==============================] - 2344s 46s/step - loss: 2.9626 - val_loss: 2.7944
Epoch 3/50
51/51 [==============================] - 2341s 46s/step - loss: 2.8525 - val_loss: 2.7050
Epoch 4/50
51/51 [==============================] - 2342s 46s/step - loss: 2.7210 - val_loss: 2.6078
Epoch 5/50
51/51 [==============================] - 2348s 46s/step - loss: 2.6050 - val_loss: 2.5439
Epoch 6/50
51/51 [==============================] - 2349s 46s/step - loss: 2.4910 - val_loss: 2.4968
Epoch 7/50
51/51 [==============================] - 2349s 46s/step - loss: 2.3623 - val_loss: 2.4007
Epoch 8/50
51/51 [==============================] - 2353s 46s/step - loss: 2.2676 - val_loss: 2.3570
Epoch 9/50
51/51 [==============================] - 2343s 46s/step - loss: 2.1814 - val_loss: 2.3452
Epoch 10/50
51/51 [==============================] - 2342s 46s/step - loss: 2.0966 - val_loss: 2.3378
Epoch 11/50
51/51 [==============================] - 2343s 46s/step - loss: 1.9927 - val_loss: 2.3161
Epoch 12/50
51/51 [==============================] - 2335s 46s/step - loss: 1.9109 - val_loss: 2.3199
Epoch 00012: early stopping
```

Fig 7.2.1 : Training results for the epochs



```
from keras import backend as K
K.clear_session()
latent_dim = 500

# Encoder
encoder_inputs = Input(shape=(max_len_text,))
enc_emb = Embedding(x_voc_size, latent_dim,trainable=True)(encoder_inputs)

#LSTM 1
encoder_lstm1 = LSTM(latent_dim,return_sequences=True,return_state=True)
encoder_output1, state_h1, state_c1 = encoder_lstm1(enc_emb)

#LSTM 2
encoder_lstm2 = LSTM(latent_dim,return_sequences=True,return_state=True)
encoder_output2, state_h2, state_c2 = encoder_lstm2(encoder_output1)

#LSTM 3
encoder_lstm3=LSTM(latent_dim, return_state=True, return_sequences=True)
encoder_outputs, state_h, state_c= encoder_lstm3(encoder_output2)
```

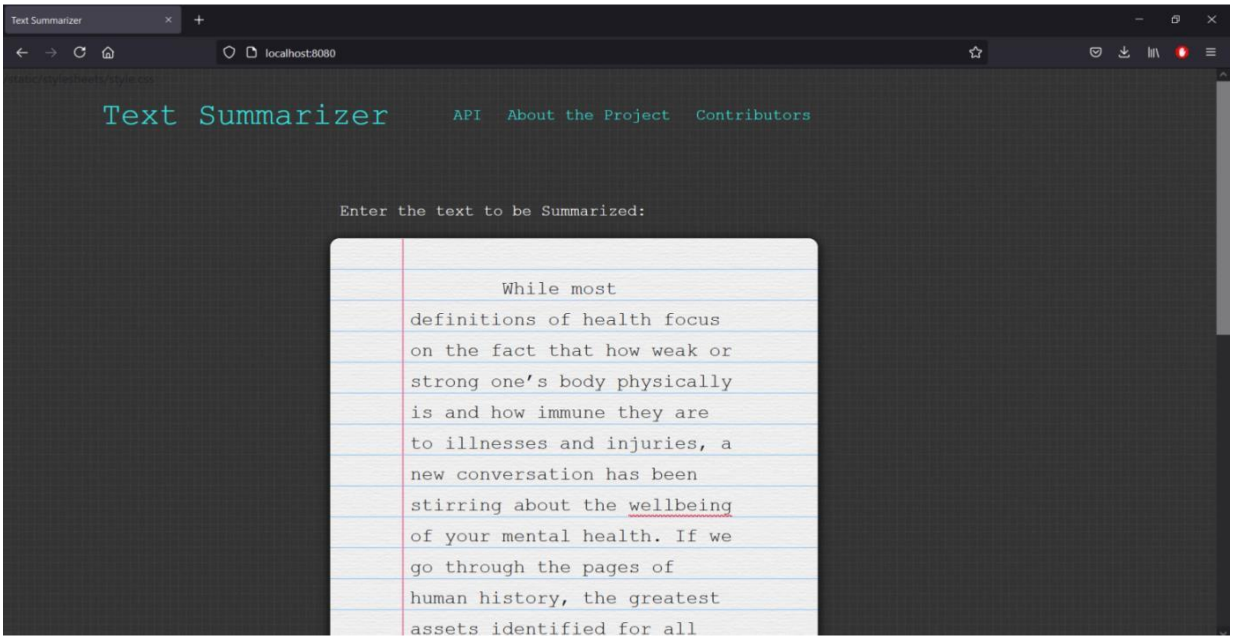
Fig 7.2.2 : LSTM layers



```
x_tokenizer = Tokenizer()
x_tokenizer.fit_on_texts(list(x_tr))

#convert text sequences into integer sequences
x_tr    =   x_tokenizer.texts_to_sequences(x_tr)
x_val   =   x_tokenizer.texts_to_sequences(x_val)
```

Fig 7.2.3 : Tokenizer

Home page where the user can enter the original text

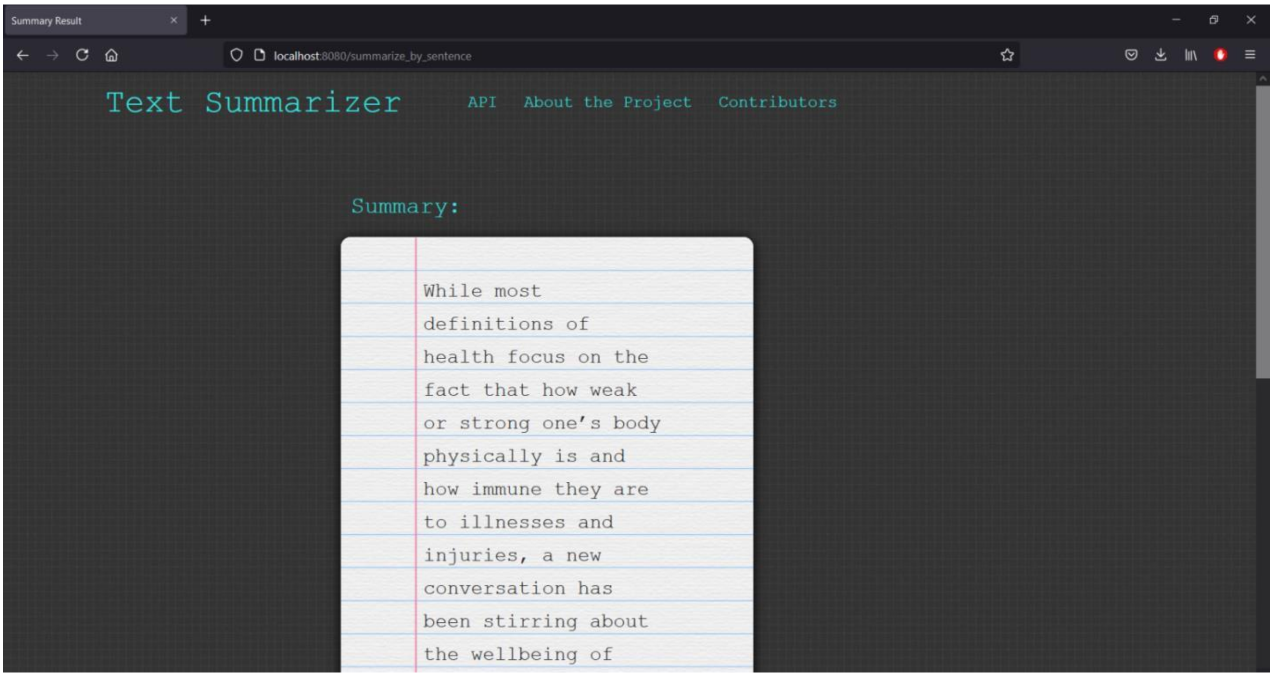


Enter the text to be Summarized:

While most definitions of health focus on the fact that how weak or strong one's body physically is and how immune they are to illnesses and injuries, a new conversation has been stirring about the wellbeing of your mental health. If we go through the pages of human history, the greatest assets identified for all

Fig 7.2.4: Home page

Result page which contains the summarized text

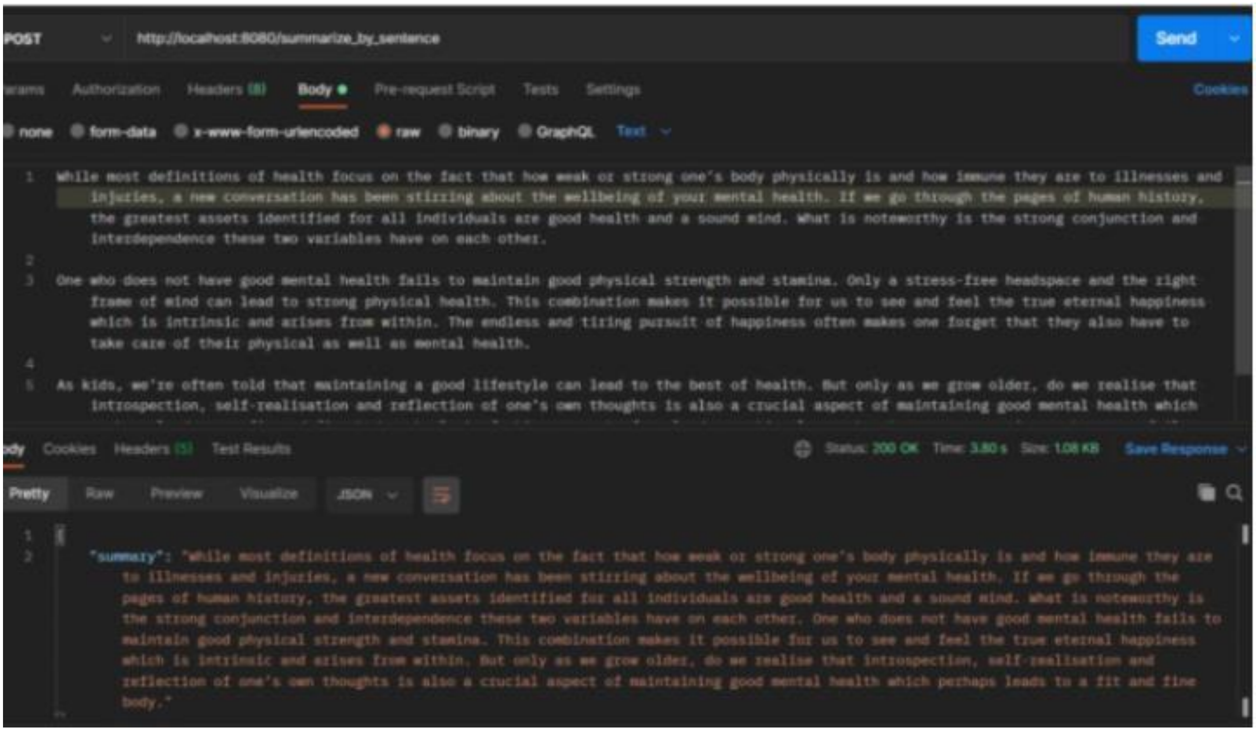


Summary:

While most definitions of health focus on the fact that how weak or strong one's body physically is and how immune they are to illnesses and injuries, a new conversation has been stirring about the wellbeing of

Fig 7.2.5: Result Page

Screenshot of Post Request, request body contains original text and the response contain summarized text.



http://localhost:8080/summarize_by_sentence

```
While most definitions of health focus on the fact that how weak or strong one's body physically is and how immune they are to illnesses and injuries, a new conversation has been stirring about the wellbeing of your mental health. If we go through the pages of human history, the greatest assets identified for all individuals are good health and a sound mind. What is noteworthy is the strong conjunction and interdependence these two variables have on each other.

One who does not have good mental health fails to maintain good physical strength and stamina. Only a stress-free headspace and the right frame of mind can lead to strong physical health. This combination makes it possible for us to see and feel the true eternal happiness which is intrinsic and arises from within. The endless and tiring pursuit of happiness often makes one forget that they also have to take care of their physical as well as mental health.

As kids, we're often told that maintaining a good lifestyle can lead to the best of health. But only as we grow older, do we realise that introspection, self-realisation and reflection of one's own thoughts is also a crucial aspect of maintaining good mental health which
```

Status: 200 OK Time: 3.80 s Size: 1.08 KB

```
{
    "summary": "While most definitions of health focus on the fact that how weak or strong one's body physically is and how immune they are to illnesses and injuries, a new conversation has been stirring about the wellbeing of your mental health. If we go through the pages of human history, the greatest assets identified for all individuals are good health and a sound mind. What is noteworthy is the strong conjunction and interdependence these two variables have on each other. One who does not have good mental health fails to maintain good physical strength and stamina. This combination makes it possible for us to see and feel the true eternal happiness which is intrinsic and arises from within. But only as we grow older, do we realise that introspection, self-realisation and reflection of one's own thoughts is also a crucial aspect of maintaining good mental health which perhaps leads to a fit and fine body."
```

Fig 7.2.6: Postman result

Chapter 8

Conclusion

8.1 Conclusion

Automatic text summarization is an old challenge however the flow research course redirects towards arising patterns in biomedicine, item survey, instruction areas, messages and sites. This is because of the reality that there is data over-burden here, particularly on the Overall Web. Automated rundown is a significant region in NLP (Natural Language Processing) research. It comprises of consequently making a rundown of at least one writings. The motivation behind extractive record rundown is to naturally choose a number of demonstrative sentences, sections, or passages from the first record. Text rundown approaches dependent on Neural Networks, Graph Theory and Clusters have, to a degree, prevailed in making a powerful synopsis of a document. Both extractive and abstractive strategies have been investigated. Most synopsis procedures depend on extractive strategies. Abstractive technique is like outlines made by people. Abstractive outline as of now requires large equipment for language generation and is hard to reproduce into the domain explicit regions.

8.2 Limitations of the Project

- Our project doesn’t convert the input file type into text automatically before summarization.

8.3 Future Scope

- For model future upgrades, one system could be to adjust the model on Udacity addresses, since the current model is the default pre-prepared model from Kaggle.

- The other improvement is fill in the holes for missing setting from the rundown, and consequently decide the best number of sentences to address the talk. This could be conceivably done through the amount of squares with bunching.

- The data set would ultimately should be changed over to a more perpetual arrangement over SQLite. Likewise, having logins where people could oversee their own synopses would be another helpful component

References

1. Ankit Kumar, Zixin Luo, Ming Xu: Text Summarization using Natural Language Processing.
2. Aravind Pai: Comprehensive Guide to Text Summarization using Deep Learning inPy https://www.analyticsvidhya.com/blog/2019/06/comprehensive-guide-text-summarization-using-deep-learning-python/
3. Sunitha C., Dr. A. Jaya, Amal Ganesh: A Study on Abstractive Summarization
4. https://medium.com/analytics-vidhya/text-summarization-using-bert-gpt2-xlnet-5ee80608e961
5. https://iq.opengenus.org/bert-for-text-summarization/
6. Derek Miller: Leveraging BERT for Extractive Text Summarization on Lectures

TEAM MEMBERS

1.

Name: Rishikesh Patil

USN: 1PE16CS198

Phone No.:8277144824

Email ID: rishikesh.patil10@gmail.com

Address:10-2/118,”Neelkanth”,
Mahant Layout, Anand Nagar,
Gulbarga - 585103

2.

Name: Akansha Makkar

USN: lPEl7CS011

Phone No.: +91 98868 01110

Email ID: akanksha.azure@gmail.com

Address: 3143, Prestige Notting Hill,
Kalena Agrahara,Bannerghatta Road
Bangalore – 560076

3.

Name: B T Amith Kumar

USN: lPEl 7CS033

Phone No.: 7348826881

Email ID: 311amith@gmail.com

Address: Plot no 8 Pragatools
colony, jeedimetla village, pet
basheerabad, medchal dist.
Hyderabad, Telangana 500055

4.

Name: Pratheek G

USN: 1PE17CS111

Phone No.: 9110285284

Email ID: pratheekg248@gmail.com

Address: Ajantha Royal Apartment 32/1
SF-10 B-Block , Lavakush nagar, Beretena
Agrahara, near Metro Whole-sale ,
Bangalore - 560100