International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 07 Issue: 05 | May 2020 www.irjet.net p-ISSN: 2395-0072

TEXT SUMMARIZATION USING NATURAL LANGUAGE PROCESSING AND GOOGLE TEXT TO SPEECH API

SUBASH VOLETI[1], CHAITAN RAJU[1], TEJA RANI[1] MUGADA SWETHA[2]

[1]EIGHT SEMESTER, DEPT OF CSE, LENDI INSTITUTE OF ENGINEERING AND TECHNOLOGY, VIZIANAGARM.
[2]ASST.PROFESSOR, DEPT OF CSE, LENDI INSTITUTE OF ENGINEERING AND TECHNOLOGY, VIZIANAGARAM.

Abstract - *Chunks of Information is available on the internet, it is most important to provide a solution to get information most efficiently and accurately. Text Summarization is the most popular problem in this modern era. The main objective of Text Summarization is extracting Summaries from Large chunks of text efficiently and accurately. Reading Time can be reduced with the use of Text Summarization. Text Summarization techniques are categorized into extractive and abstractive summarization. In this paper, we focus on the implementation of Text Summarization using Unsupervised Text Rank Algorithm and converting the summarized text into Audio File Using Google Text To Speech API. Large Chunks of Text/ web page URL link is given as input and Summary is generated and convert the Summary into Audio file Using GTTS API. In this paper, we show how to convert summarized text into Audio file Using GTTS API.*

Key Words: **Text Summarization, Text Rank Algorithm, NLTK, GTTS(Google Text To Speech) API, Extractive Text Summarization**

1. INTRODUCTION

Text Summarization is summarizing huge chunks of text into shorter form without changing semantics. Text summarization has a huge demand in this modern world. The main advantage of Text Summarization is the reading time of the user can be reduced. Text Summarization has categorized into Extractive and Abstractive Text Summarization. In this paper, we mainly focus on Extractive Text Summarization and it's an implementation using Text Rank Algorithm. Summarized Text can be converted into Audio File by using Text to Speech API. In this paper, we show how to implement how to convert summarized into Audio File Using GTTS (Google Text to speech) API. Unsupervised Text Rank Algorithm is used for implementing Text Rank which will give efficient results. Advanced NLTK Techniques are used in Text Rank Algorithm to get Efficient Results. In this Paper We will show the implementation of Text Summarizer Tool Which will provide Graphical User Interface for the End-user. Text Summarizer Tool can be created by using python's Tkinter tool kit.Large Chunks of Text can be taken as input in the user interface and summarize the text and summarized text can be converted into Audio File using Text To Speech API. Web page URL link is taken as input from End user and we Extract total text from that URL link and summarize the text and convert the summarized text into Audio File using GTTS API.

1.1 TEXT SUMMARIZATION TECHNIQUES

Extractive Text Summarization:- This Method Focuses on selecting important parts from the large chunks of text such as phrases and synonyms and combine to form meaningful summary .

Abstractive Text Summarization:- This method relies on old NLP Techniques to Extract Novel Sentences from the Large Chunks of text and produce them into a meaningful form.

1.2 Stages of Text Summarization:-

(i) Content Selection:- Choose Sentences to extract from large Chunks of Text.

(ii) Information Ordering:- Choose an order to place Summary.

(iii) Sentence Realization:- Clean up the sentences

1.3 TEXT RANK

Text Rank is a general purpose, Unsupervised Graph based Ranking Algorithm. It can be implemented by using Advanced NLTK Techniques which gives efficient results for Text Summarization.

Tasks in Text Rank:-

Text Rank include Two NLP tasks

(i) keyword Extraction Task:- The task of Keyword Extraction Algorithm is to automatically identify set of terms that best describe the document

(ii) Sentence Extraction Task:- Text rank is well suited for applications involving entire sentences, since it allows for ranking over texts units that is recursively compute based on information drawn from entire text.

Google Text to Speech API:- It is used for converting the Summarized text into Audio File.

2. LITERATURE REVIEW

In [1] references the existed Chinese and English automatic summarization technology in domestic and foreign, and proposes a method of Tibetan automatic summarization. Experiments analysis three summarization methods based on Text Rank, based on LexRank and based on LexRank+TextRank respectively, and using the ROUGE value to evaluate the effect of summarization.

In [2] also identifies complex words from the document and substitutes simple words for the same .Sentence reconstruction is also performed to shorten long sentences to increase the scale of 'summarization. The user can take image of the article he wishes to shorten and upload it to server. The proposed application will extract the text from the image and provide the summarized version of the article to the user.

In [3] shows that summarization result not only depends on optimized function, and also depends on a similarity measure.

3.EXISTING SYSTEM:-

In existing system doesnot focus on Advanced NLP Techniques. End user does not get reliable summaries . In Existing system uses Abstractive Text Summarization Technique Which does not give reliable summary. In this Paper, we focus on improving the existing system of the user.

3.1 DrawBacks in Existing System:-

(I) summary is less accurate

(II) Time constraint is less

(III) Computational speed is more.

(IV) It uses Abstractive Summarization.

4. Proposed System:-

In the proposed system mainly focuses on providing a reliable summary. In the Proposed System we are using Extractive Text Summarization which will provide reliable summary. An unsupervised Text Rank Algorithm is used for implementing Text Summarization. Advanced NLTK Techniques is used for making Graphical User Interface. Additionally converting the summarized Text into Audio File using GTTS API. Our Project Focuses on Providing a System which will provide an Accurate Summary. In Our Project Input Can be Large Chunks of Text/web page urls and summarize the text and convert the summarized text into Audio File using GTTS API.

4.1 Advantages of Proposed System:-

(i) Our Algorithm executes with good performance because we are executing in a distributed environment.

(ii) Time Constraint is less

(iii) Computational Speed is more.

(iv) Accurate Summary

(v) It uses Extractive Summarization.

4.2 REQUIREMENTS:-

4.2.1 FUNCTIONAL REQUIREMENTS:-

- Large Chunks of Text.
- Web Page Urls.
- Text Rank Algorithm.
- Word2vec Representation(Glove Algorithm)
- Similarity Matrix(Cosine Similarity).
- GTTS API.

4.2.2 NON FUNCTIONAL REQUIREMENTS:-

- Reliability.
- Performance
- Usability.
- Platform independent
- Supportability.

4.3. SYSTEM ARCHITECTURE

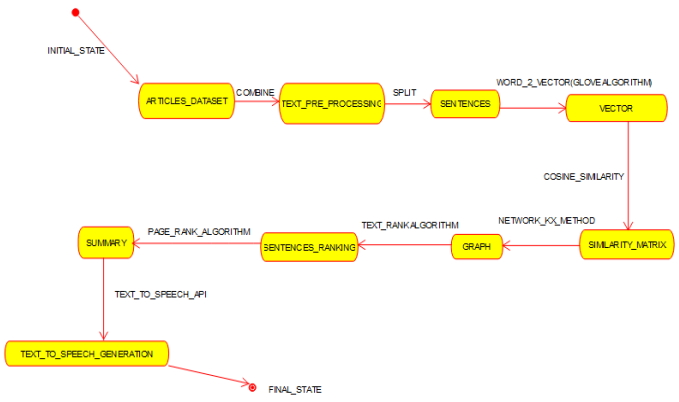

FIG-1: SYSTEM ARCHITECTURE FOR TEXT SUMMARIZATION USING NLP AND GTTS API

4.3.1 USE CASE DESIGN:-

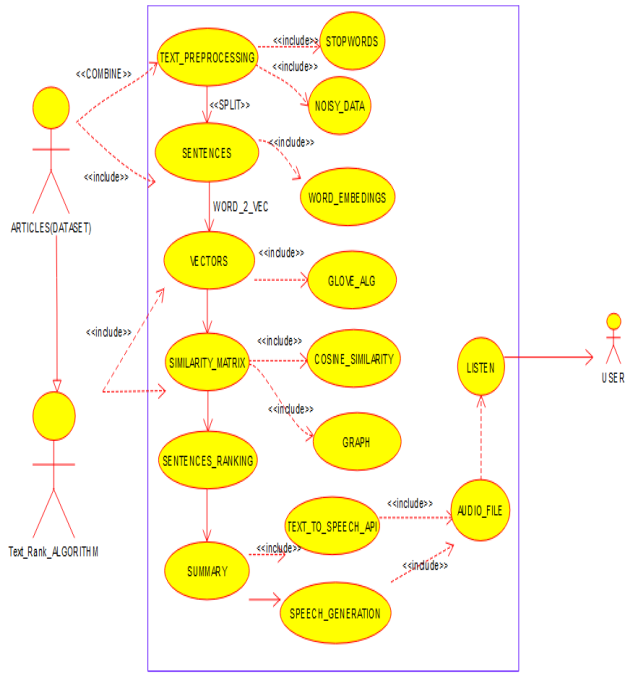

FIG-2:- USE CASE DESIGN OF TEXT SUMMARIZATION USING NLP AND GTTS API.

5. ALGORITHM SPECIFICATION:-

INPUT:- Large Chunks of Text/articles/Web Page Urls

OUTPUT:- Summarized Text and Audio File of Summarized Text

STEP1:- Concatenate all the contained in the articles

STEP2:- Entire concatenated Text is Split into individual Sentences.

STEP3:- Find Vector Representation(Word Embeddings) for each and every sentence by using Glove Algorithm.

STEP4:- Similarities Between sentence vectors are calculated and stored in a matrix using Cosine Similarity

STEP5:- Convert the similarity matrix into Graph using Page Rank Algorithm.

STEP6:- Find a certain number no top ranked sentences using page rank algorithm to form summary.

STEP7:- Convert the summarized text into audio file using Google Text To speech API.

5.1 PROJECT IMPLEMENTATION:-

/*PYTHON LIBRARIES REQUIRED*/

```
import numpy as np
import pandas as pd
import nltk
nltk.download('punkt')
import re
```

/*READ THE DATA*/

```
df = pd.read_csv("textsummarizer.csv");
```

/*STEP1:- CONCATENATE ALL THE TEXT*/

```
df['article_text'][0]
```

/*STEP2:- SPLIT TEXT INTO SENTENCES*/

We will use the sente_tokenize( ) function

```
From nltk.tokenize import sent_tokenize
sentences = []
for s in df['article_text']:
sentences.append(sent_tokenize(s))
```

/*STEP3:-WORD2VECTOR REPRESENTATION(GLOVEWORD EMBEDDINGS)*/

We will be using the pre-trained Wikipedia 2014 + Gigaword 5 GloVe vectors the size of these word embeddings is 822 MB.

/*download glove word embedding using below links */

```
!wget http://nlp.stanford.edu/data/glove.6B.zip
```

- !unzip glove*.zip

```
/*Extract Word Embeddings or Word Vectors*/
word_embeddings = {}
f = open('glove.6B.100d.txt', encoding='utf-8')
for line in f:
values = line.split() `
word = values[0]
coefs=np.asarray(values[1:], dtype='float32')
word_embeddings[word] = coefs
f.close()
```

/*STEP4:-SIMILARITYMATRIX REPRESENTATION*/

#The next step is to find similarities between the sentences, and we will use the cosine similarity approach for this challenge.

#We will use Cosine Similarity to compute the similarity between a pair of sentences.

```
fromsklearn.metrics.pairwiseimport
cosine_similarity
```

/* STEP5 SIMILARITY MATRIX INTO GRAPH*/

NODES = SENTENCES

EDGES = SIMILARITY SCORES BETWEEN THE SENTENCES

On this graph, we will apply the PageRank algorithm to arrive at the sentence rankings.

```
import networkx as nx
nx_graph = nx.from_numpy_array(sim_mat)
scores = nx.pagerank(nx_graph)
```

/*STEP5:-SUMMARY EXTRACTION*/

#Finally, it's time to extract the top N sentences based on their rankings for summary generation.

```
ranked_sentences = sorted(((scores[i],s) for i,s in enumerate(sentences)), reverse=True)
```

/*STEP6:- SUMMARIZED TEXT TO AUDIO FILE*/

```
from gtts import gTTS
import os
language = 'en'
myobj = gTTS(text=mytext, lang=language, slow=False)
myobj.save("project2.mp3")
os.system("mpg321 project2.mp3").
```

5.2 TEXT SUMMARIZER TOOL:-

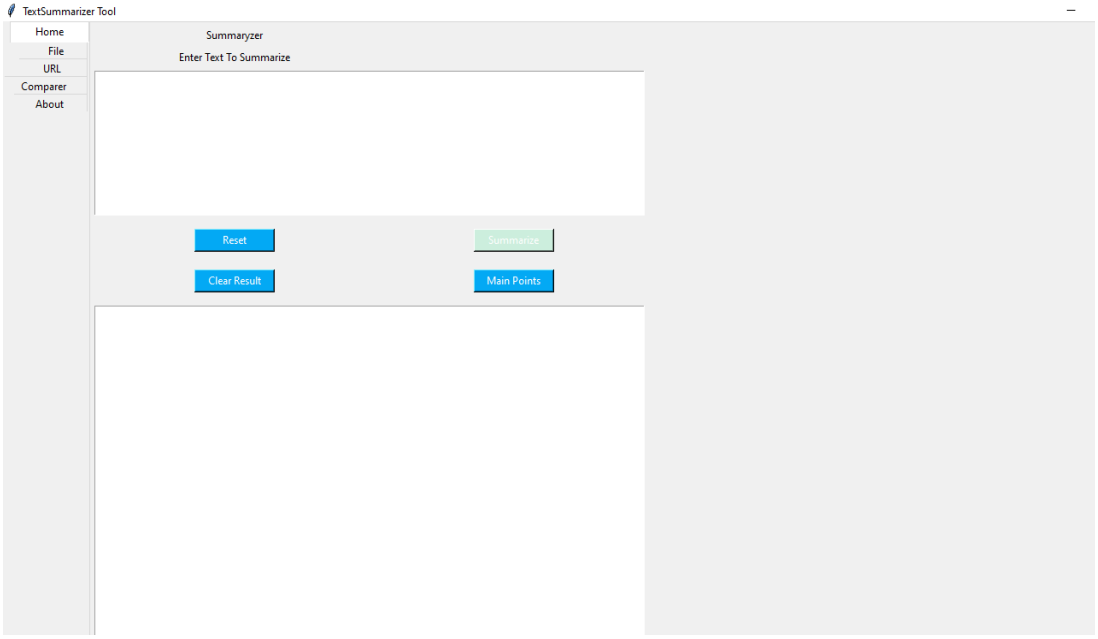

FIG-3:- TEXT SUMMARIZER TOOL

Text Summarizer tool can be created using Tkinter tool kit and Advanced NLTK python Libraries. In this tool we can copy the text which we want to summarize and click the summary button and summary is displayed.

5.3 TEST SCENARIOS:-

5.3.1 TEST SCENARIO 1:-

Input:- Large Chunks of Text

Output:- Summarized Text and It's Audio File

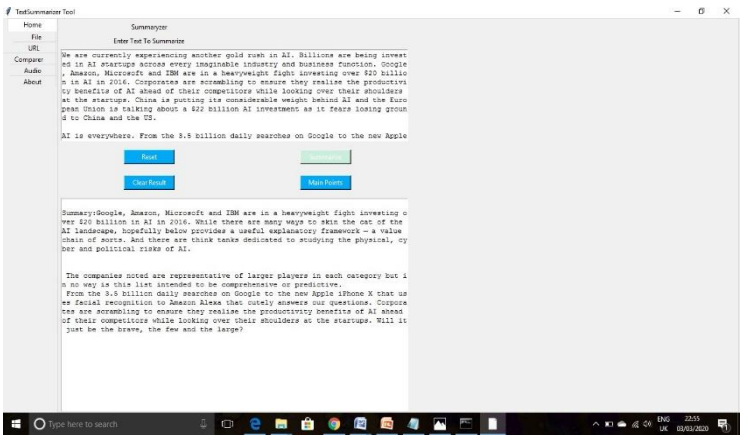

FIG-4:- DISPLAY SUMMARY

Description of Test Scenario 1:-

(i) Copy the text which you want to summarize.

(ii) Click the summarize Button

(iii) Summary will be displayed

(iv) Audio will be placed on your required project folder

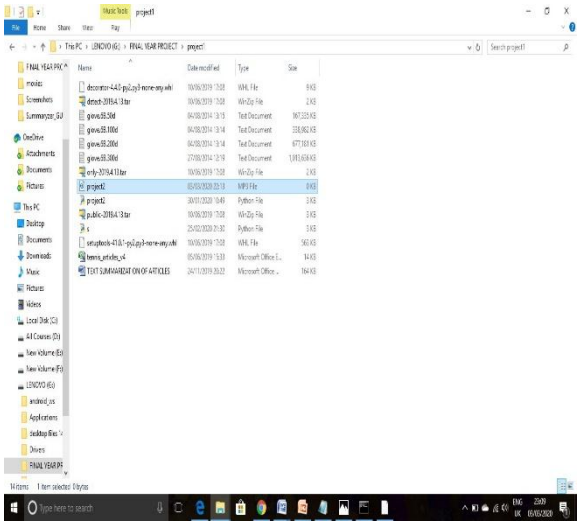

FIG:- AUDIO FILE OF SUMMARIZED TEXT.

5.3.2 TEST SCENARIO 2

INPUT:-WEB PAGE URL CONTAINS HUGE CHUNKS OF TEXT

OUTPUT:-SUMMARY OF THE WEB PAGE AND IT'S AUDIOFILE

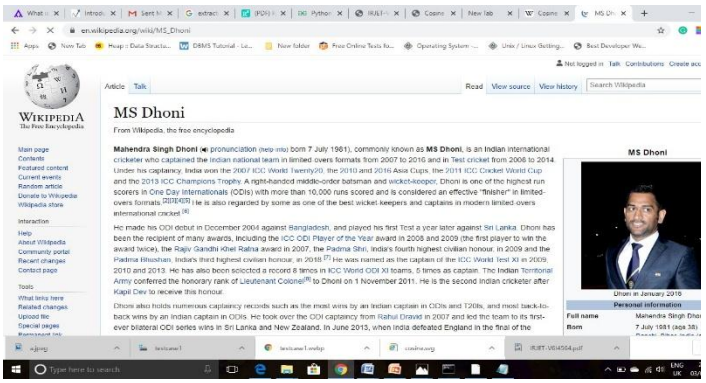

FIG:- SUMMARIZING MS.DHONI WIKIPEDIA PAGE.

DESCRIPTION OF TEST SCENARIO 2:-Summarizing web page of Ms. Dhoni Wikipedia

(i)copy the web page url of MsDhoni Wikipedia

(ii) paste it on the text summarizer tool

(iii) click "GET Text" button to get the text

(iv) click "summarize" button

(v)summary is displayed and it's Audio file on project Folder

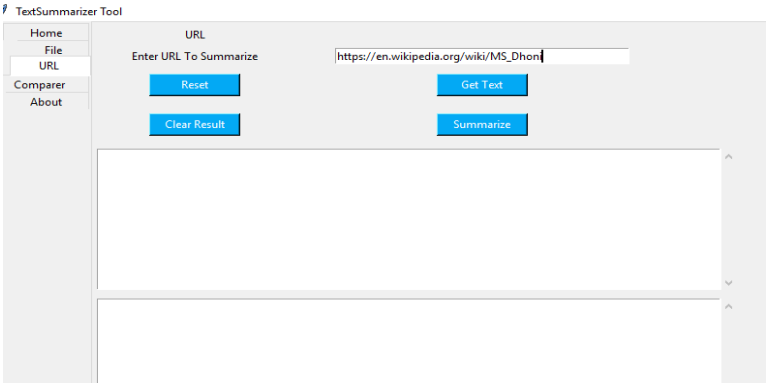

FIG:- COPY AND PASTE WEB PAGE URL OF MS.DHONI WIKIPEDIA PAGE.

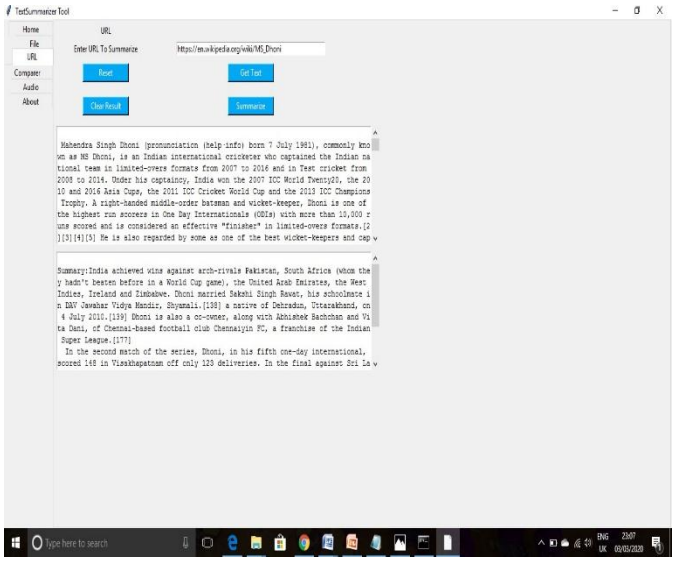

FIG:- SUMMARY OF MS.DHONI WIKIPEDIA WEB PAGE

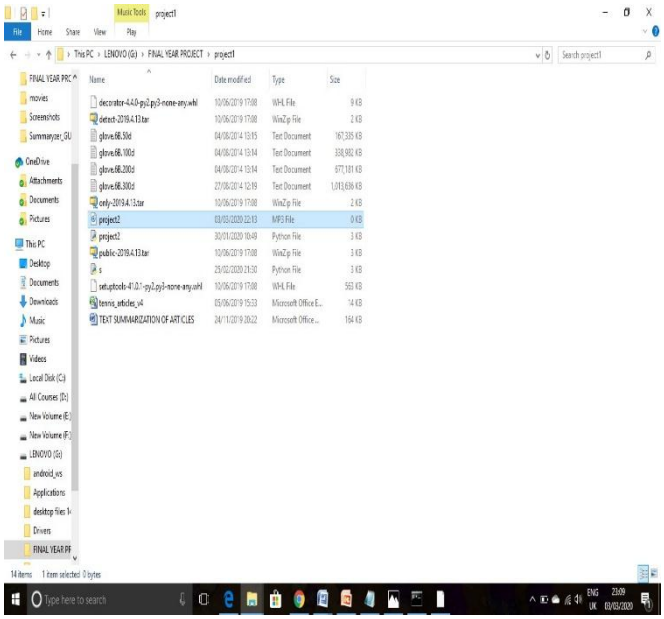

FIG:- AUDIO FILE OF MS.DHONI SUMMARIZED WEB PAGE.

TEST SCENARIO 3:-

INPUT:- TEXT FILE CONTAINING HUGE CHUNKS OF TEXT

OUTPUT:-SUMMARY OF TEXT FILE AND IT'S AUDIO FILE.

DESCRIPTION OF TEST SCENARIO3:-

(i)click "OPEN FILE" button
(ii) Text is placed and click "Summarize" button
(iii) summary and it's Audio file is placed.

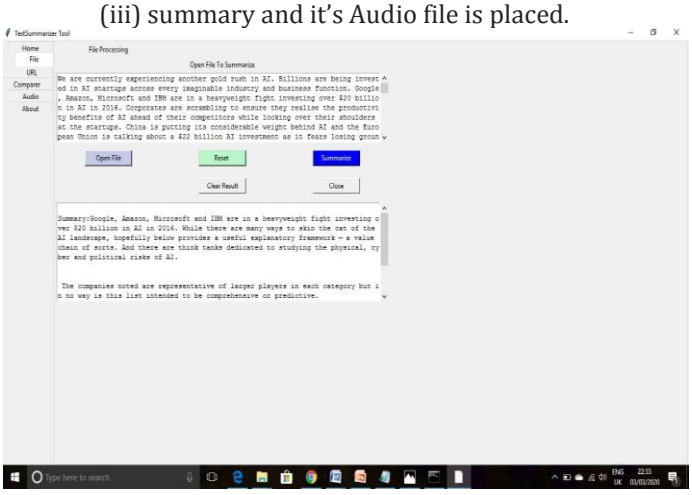

FIG:-SUMMARIZATION OF A TEXT FILE

CONCLUSION:-

This paper mainly focused on creating a system that gets concise summaries of articles or large chunks of text or any blog past. These implications of this would mean that knowledge gathering would be easier and time saving. This project will reduce reading time and provides concise summary. On implementing Unsupervised Text Rank Algorithm Efficient results will be takes place. Access time for Information searching will be improved. Converting the Summarized text into Audio File Which is used at various Real Time Scenarios. On implementing Unsupervised Text Rank Algorithm Efficient results will be takes place.In this paper implementation of GTTS API has been shown.

ACKNOWLEDGEMENT

We would like to thank our Assistant Professor Mr.M.Swetha. Under her guidance we are able to complete this journal.

REFERENCES

[1] Li, Ailin, et al. "The Mixture of Text rank and Lexrank Techniques of Single Document Automatic Summarization Research in Tibetan." 2016 8th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC). Vol. 1. IEEE, 2016.

[2] Histogram Summarization of Long Text Extracted from Article Images By Integrating Extractive and Abstractive Text Summarization Methods.

[3] A new Sentence similarity measure and sentence based extractive technique on Automatic Text summarization.

[4] Ta Hahn, Udo, and Inderjeet Mani. "The challenges of automatic summarization." Computer 33.11 (2000): 29-36. 3.

[5] Salton, Gerard, et al. "Automatic text structuring and summarization." Information processing & management 33.2 (1997): 193-207.

[6] Chen, Fang, Kesong Han, and Guilin Chen. "An approach to sentence-selection-based text summarization." 2012 IEEE Region 10 Conference on Computers, Communications, Control and Power Engineering. TENCOM'02. Proceedings.. Vol. 1. IEEE, 2012.

[7] Abuobieda, Albaraa, et al. "Text summarization features selection method using pseudo genetic-based

model."2012 International Conference on Information Retrieval & Knowledge Management. IEEE, 2012.

BIOGRAPHIES

Muguda swetha is Assistant Professor in Dept of CSE since 2011. She has total 9 years experience of teaching. Her major expertise is Data Analytics, Natural Language Processing and Networks.

SUBASH VOLETI is Pursuing his B.Tech in Lendi Institute of Engineering and Technology, Andhra Pradesh. He worked his final year project in the stream of Natural Language Processing. He has participated in many workshops regarding Natural Language Processing, Machine Learning. He was an CSI Member in 2017-2020. His Areas of streams are NLP,ML.

CHAITAN RAJU is pursuing his B in CSE in Lendi Institute of Engine and Technology, Vizianagaram, Ar Pradesh. He worked his final project in the stream of Na Language Processing. He was ar Member in the year 2017-2020 areas of Streams include AI and M

Teja Rani is pursuing her B.Tech ir in Lendi Institute of Engineering Technology, vizianagaram, Ar Pradesh. She worked her final project in the stream of Natual Lang processing. She was an CSI Memb the year 2017-2020. Her area of Str include NLP and ML.