iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 7 **Issue:** IV **Month of publication:** April 2019

DOI: https://doi.org/10.22214/ijraset.2019.4455

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 6.887
Volume 7 Issue IV, Apr 2019- Available at www.ijraset.com

Text Summarization for GRE Exam

Sumeet Deshpande[1], Shantanu Papal[2], Anant Khandalikar[3], Anish Kulkarni[4]
[1, 2, 3, 4]*Department of Computer Engineering, Vishwakarma Institute of Technology , Pune*

Abstract: ***Automatic text summarization is a process of producing a expressive summary importance of the main points from the source text document. The brief summary formed by the automatic text summarizer allows individual to read the brief summary quickly and easily, understanding the contents of the original text document into a brief summary without reading the entire document. The main reason for automatic text summarizer is to create a summary using less number of words and sentences. There are two types of text summarization techniques: Extractive and Abstractive. In extractive summarization technique, the relevant sentences from the source document are extracted in a proper order to form a meaningful summary. The relevant sentences are selected on the basis of their comparison with other sentences in the document and their score. While the abstractive summarization technique uses the natural language generation technology for generating the summary. It creates a synopsis like of a human. It is not desirable for a human to create a summary manually from the large text document. This project consists GUI in python using Tkinter library. Tkinter library helps to add GUI feature in python. It is a standard Python interface to the Tk GUI toolkit shipped with Python. Python with tkinter outputs the fastest and easiest way to create the GUI applications. Creating a GUI using tkinter is an easy task.***

Keywords: NLP, Abstractive Summary, Extractive Summary, NLTK.

I. INTRODUCTION

Text summarization is the method of filtering significant data from a basis (or sources) to produce a reduced version for a specific user (or users) and job (or tasks). Humans are usually decent at this job as we have the capacity to know the meaning of a text and abstract prominent details to summarize the papers using our own words. However, automatic systems for text summarization are vital in today's world where there is an over-abundance of information and deficiency of manpower as well as time to understand the information. There are many reasons why Automatic Text Summarization is useful:

1) Reduce reading time.
2) Makes Process Easier.
3) Text summarization improves the efficiency of indexing.
4) Automatic summarization algorithms are less unfair than human summarizers.
5) Personalized summaries are useful in question-answering systems as they provide personalized information.
6) Using automatic or semi-automatic summarization systems allows marketable abstract services to rise the number of text documents they are able to process.

II. LITERATURE REVIEW

GRE is exam which is conducted by ETS exam for Post-Graduation in countries like USA,CANADA ,Singapore etc.
GRE consist of writing section which Analytical writing section in which aspirant must type essay for given topic .In this grading which is given to aspirant is manual based and it takes 2 weeks time to come up result of it. So we come up with an solution to save time we can use text summarization technique.

A. Dharmendra Hinhu et al. [24] In this paper author practices the extractive text summarization. The author takes up the Wikipedia Articles as input to the system and finds text scoring of each word and create summary accordingly.
B. N. Moratanch et al. [26]In this paper the author grants the comprehensive review of extraction based text summarization techniques. In this paper the author delivers study on extractive summarization approach by categorized them in:Supervised learning approach and Unsupervised learning method.

III. TYPES OF TEXT SUMMARIZATION METHODS

A. Based on Input type

1) Single Document, where the input data length is very small. Many older time system, projects or software's uses summarization methods distributed with single document summarization.
2) Multi Document, where the length of text can be randomly lengthy.

B. Based on Purpose

1) Generic, where the system makes no assumptions about the specific area or technology or content of the text to be summarized and treats all inputs as similar. The mostly of the effort that has been done rotates about general summarization.

2) Domain-specific, where the model uses domain-specific knowledge to form a more precise summary. For example, if suppose machine has pre-knowledge about domain and area of document which it summarizes.

3) Query-based, where the summary only holds data which responses natural language questions about the given text.

C. Based on Output Type

1) Extractive, where significant sentences are designated from the input text to form a summary. Most summarization methods today are extractive in nature.

2) Abstractive, where the method creates its own phrases and sentences to offer a more articulate summary, like what a human would produce. This approach is definitely a more appealing, but much more difficult than extractive summarization.

D. Key Terms

1) *Tokenization:* Tokenization is the method by which huge quantity of text is divided into reduced parts called tokens.

2) Stop Words: The method of converting data to something a computer can know is referred to as **pre-processing.** One of the major methods of pre-processing is to riddle out unusable data. In natural language processing, useless words (data), are mentioned to as stop words. A stop word is a usually used word (such as “the”, “a”, “an”, “in”) that a search engine has been set to disregard, both when indexing entries for searching and when recovering them as the outcome of a search query.

3) *Stemming:* Stemming and Lemmatization are Text Normalization (or sometimes called Word Normalization**)** techniques in the field of Natural Language Processing that are used to prepare text, words and documents. Example

a) Play---play

b) Plays---play

c) Playing—play

d) Car,cars,car’s----car Root word of above all words is play so whether whatsoever verb it has fit in to above it will store as play. Likewise, for plurals also it will store unique word

4) POS Tagging: Parts of speech Tagging is accountable for reading the text in a language and transfer some specific token (Parts of Speech) to each word.

a) *Input:* Everything to allow us. Output: [('Everyone', NN), ('to', TO), ('allow', VB), ('us', PRP)]

5) *Chunking:* Chunking is used to improve extra structure to the sentence by following parts of speech (POS) tagging. It is also known as shallow parsing. The resulted group of words is called "**chunks**." In shallow parsing, there is maximum one level between roots and leaves while deep parsing comprises of more than one level. Shallow Parsing is also called light parsing or chunking. The primary usage of chunking is to make a group of "noun phrases." The parts of speech are combined with regular expressions.

IV. TEXT SUMMARIZATION MODEL PROCESS

A. Sentence Tokening

Split each document into its basic sentences using precise rules for sentence delimiters for each language. NLTK’s sentence tokenizer will do this job for us.

B. Skip Thought Encoder

We need a way to generate fixed length vector representations for each sentence in our documents. These representations should encode the inherent semantics and the meaning of the corresponding sentence. The well known Skip-Gram Word2Vec method for generating word embeddings can give us word embeddings for individual words that are present in our model’s vocabulary (some fancier approaches can also generate embeddings for words which are not in the model vocabulary using subword information).

For sentence embeddings, one easy way is to take a weighted sum of the word vectors for the words contained in the sentence. We take a weighted sum because frequently occurring words such as ‘*and*’, ‘*to*’ and ‘*the*’, provide little or no information about the sentence. Some rarely occurring words, which are unique to a few sentences have much more representative power. Hence, we take the weights as being inversely related to the frequency of word occurrence. However, these unsupervised methods do not take the sequence of words in the sentence into account. This may incur undesirable losses in model performance. To overcome this, I chose to instead train a Skip-Thought sentence encoder in a supervised manner using Wikipedia dumps as training data.

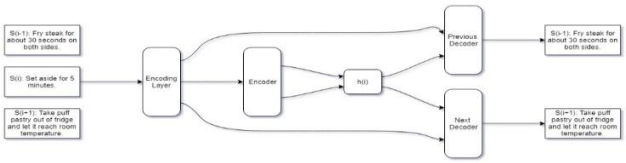
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 6.887
Volume 7 Issue IV, Apr 2019- Available at www.ijraset.com

C. Clustering

After producing sentence embeddings for each sentence in an email, the approach is to cluster these embeddings in high-dimensional vector space into a pre-defined number of clusters. The number of clusters will be equal to desired number of sentences in the summary. I chose the numbers of sentences in the summary to be equal to the square root of the total number of sentence in the email. One can also have it as being equal to, say, 30% of the total number of sentences.

D. Summarization

Each cluster of sentence embeddings can be interpreted as a set of semantically similar sentences whose meaning can be expressed by just one candidate sentence in the summary. The candidate sentence is chosen to be the sentence whose vector representation is closest to the cluster center. Candidate sentences corresponding to each cluster are then ordered to form a summary for an email. The order of the candidate sentences in the summary is determined by the positions of the sentences in their corresponding clusters in the original email. For example, a candidate sentence is chosen as the first line in the summary if most of the sentences that lie in its cluster occur at the beginning of the document.



E. Spacy

spaCy is a comparatively innovative package for "Industrial asset NLP in Python" established by Matt Honnibal at Explosion AI. It is intended with the applied data scientist in mind, meaning it does not consider the user down with decisions over what obscure systems to practice for mutual tasks and it's reckless. If you are aware with the Python data science stack, spaCy is your NumPy for NLP – it's rationally low-level, but very innate.

Spacy offers a all tasks commonly used in any NLP project, same as NLTK including:

1) Tokenisation
2) Lemmatisation
3) Part-of-speech tagging
4) Entity recognition
5) Dependency parsing
6) Sentence recognition
7) Word-to-vector transformations
8) Methods for normalising text

F. spaCy v/s NLTK

SPACY

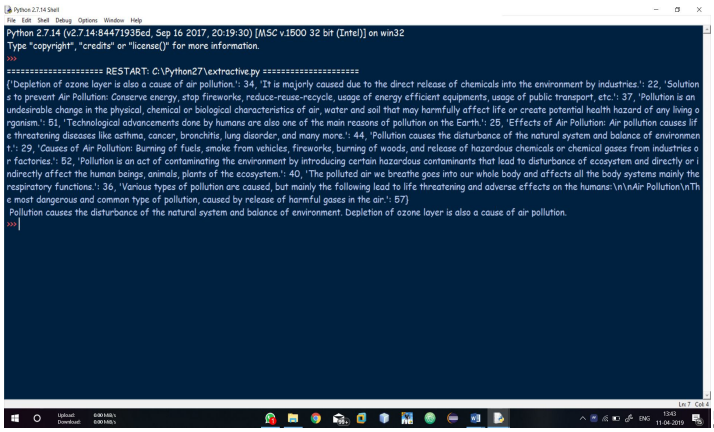
1) Over 400 times faster
2) State-of-the-art accuracy
3) Tokenizer maintains alignment
4) Powerful, concise API
5) Integrated word vectors
6) English only (at present)

G. NLTK

1) Slow
2) Low accuracy
3) Tokens do not align to original string
4) Models return lists of strings
5) No word vector support
6) Multiple languages

V. RESULTS AND CONCLUSIONS

It will generate output of given text which is stored in text file which will be in same as folder in project folder



520

VI. COMPETITIVE ANALYSIS

A. In deference to Dharmendra Hinhu et al. [24], the paper uses extraction technique for summarization. Our paper provides more relevant approach of extraction using abstraction for summarization.

B. N. Moratanch et al. [26]. In this paper the author delivers study on extractive summarization method by characterized them in Supervised learning approach and Unsupervised learning approach. This research is more inclined towards Supervised Learning to enhance the extraction technique of summarization.

VII. LIMITATIONS

A. In this summarization technique more focus is on weightage or an occurrence of particular word it gives importance to that word, but this approach may neglect important sentence from given text

B. It also decides summary only on base of higher frequency word which will not give accurate result in every situation

VIII.FUTURE SCOPE

A. Text Summarization has wide scope in field of journalism, which will help to get accurate summary of paragraph and saves labour work

B. In legal cases it will help to reduce huge chunks of papers into short summary which can be easily understandable by everyone

C. It will also help to shorten message send to customer care chat support

IX. CONCLUSION

Text Summarizer helps to reduce long, lengthy kind of data into short and easy summary which can be understand easily. It also helps avoid tedious job of summarizing text into short paragraph. This easy extraction of sentence have generated large and wide scale of application. In this type of summarizer, it calculates frequency of each word and summary is created by weightage of sentences.

X. ACKNOWLEDGEMENT

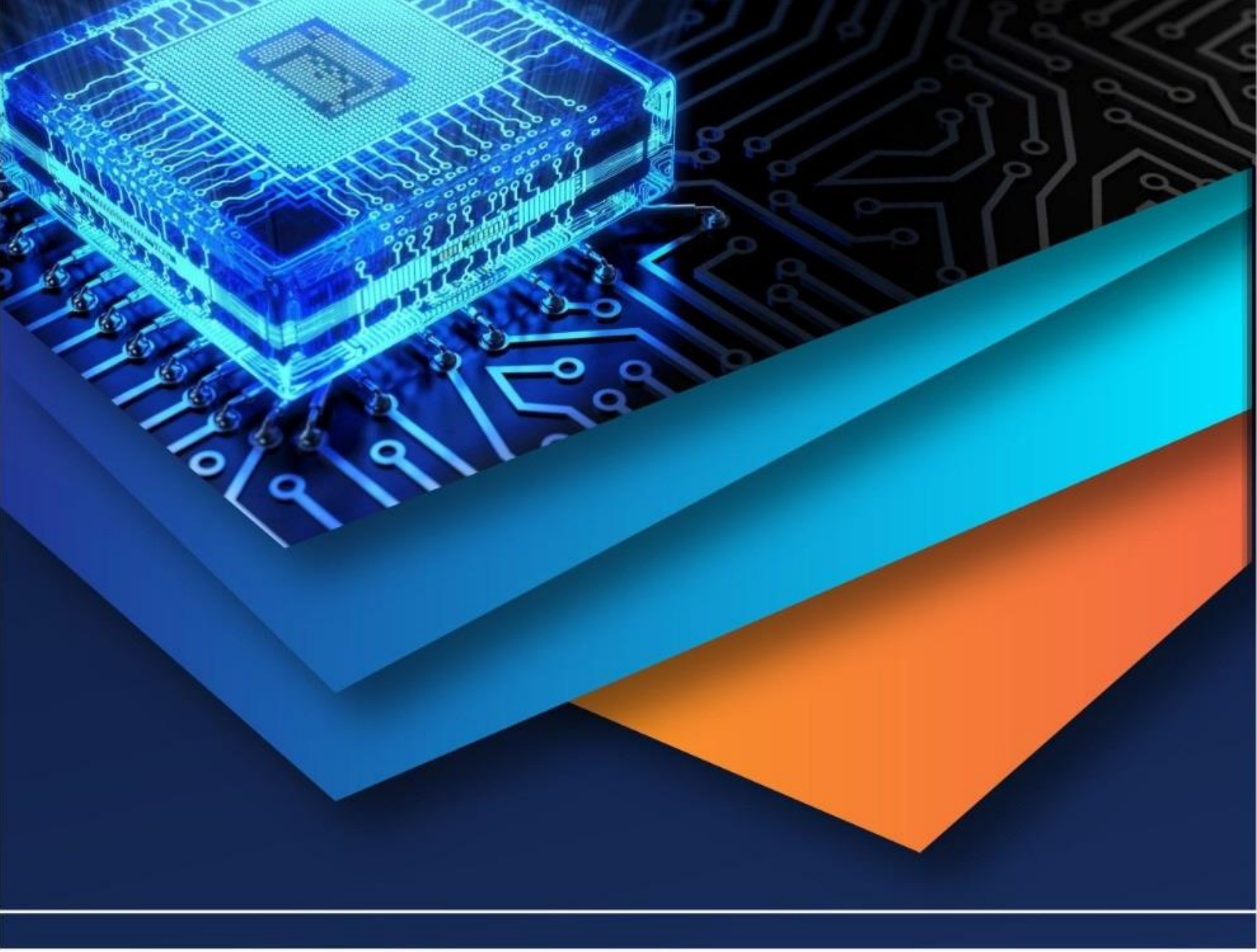
We would like to thank the respected director of Vishwakarma Institute of Technology DR. Rajesh Jalnekar sir for including Course Project of EDD (Engineering Design and Development) in our syllabus. We would also like to thank our Head Of Department of Computer Engineering Dr. Deshpande. We would also like to thanks Prof. S.H.Sutar for guiding and providing us the needful support, necessary resources and platform to represent our idea.

REFRENCES

[1] J. Hobbs. 1974. A model for natural language semantics. Part I: The model. Technical report, Yale University.

[2] A. Hulth. 2003. Improved automatic keyword extraction given more linguistic knowledge. In Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing, Japan

[3] Turney. 1999. Learning to extract keyphrases from text. Technical report, National Research Council, Institute for Information Technology



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY