International Journal of Engineering and Advanced Technology (IJEAT)
ISSN: 2249 – 8958 (Online), Volume-9 Issue-4, April, 2020

Text Summarization using Ml and Nlp

Manikanta K.B, Bhagavath Sai M, I Venkat, Sreekar Reddy B

Abstract*: Quantity of data produced per day is around 2.5 quintillion bytes. Right now, no one has the time to pursue each and everything. With the growth of technology and digital media, people are becoming very lazy; they are looking for everything more smartly. If they want to read any article or newspaper, they cannot go through every line that has been given. To overcome this problem, an automatic text summarizer using Machine Learning (ML) and Natural Language Processing (NLP) with the python programming language has been introduced. This automatic text summarizer will generate a concise and meaningful summary of the text from resources like textbooks, articles, messages by using a text ranking algorithm. The input text that is given will be split into sentences; these sentences are again converted into vectors. These vectors are represented as a similarity matrix and based on these similarities; matrices sentence rankings will be given. The higher ranked sentences will be the final summary of the given input text.

Keywords: NumPy, Pandas, Machine Learning (ML), Natural Language Processing (NLP), Text Ranking Algorithm.

I. INTRODUCTION

For a few decades, the summarization has become a field of study for creating an accurate, fluent and short summary for more important documents. The need for summarization is becoming a necessity with the growth of an enormous amount of data that is produced every day in online. So, to reduce the long pieces of text into a meaningful summary that can be easily understood by people we came up with a technique that summarizes the text by using Machine Learning algorithm and Natural Language Processing (NLP).

According to International Data Corporation (IDC), it is expected that the total sum of the amount of data produced around world data will reach 175 Zettabytes (ZB) by 2025. To summarize such a massive amount of data, a Machine Learning algorithm is implemented. Natural language processing (NLP) in our lives has a huge impact, and it is implemented as one application as an automatic text summarization. This summarization is spiking with the availability of a large amount of data that is produced every day. In this paper, along with the text summarization process, we will implement Text Rank algorithm and its working process, which can be performed by using the python programming language.

Revised Manuscript Received on April 25, 2020.

* Correspondence Author

Manikanta *, CSE department, GITAM University, Bangalore, India. Email: 11manikantareddy@gmail.com

Bhagavath Sai M, CSE department, GITAM University, Bangalore, India. Email: maddukuribhagavathsai@gmail.com

Venkat I, CSE department, GITAM University, Bangalore, India. Email: iskalavenkat999@gmail.com

Sreekar Reddy B, CSE department, GITAM University, Bangalore, India. Email: bsrikarreddy98@gmail.com

II. LITERATURE SURVEY

An operation that automatically removes suffixes from the words in English is done by using an algorithm called the Porter Stemming Algorithm, which is useful in the information retrieval process. The document in this information retrieval environment is in the form of a vector of words or terms. Earlier this system is developed only for a single time later, the performance of this system is improved by grouping the words and removing different suffixes like -ED, -ING, -ION, -IONS. The total number of terms can be reduced by using a suffix stiffing process and also it reduces the data size and complexity that is an advantage in this algorithm. This process removes only inflectional morphemes like declinations, conjunctions but not derivational morphemes (Parts of speech) which is a disadvantage in this algorithm. [1] To create an automatic text summarization system whose primary intention is to have a fluent and coherent summary where only the main points in the document are outlined. In this process, from the words, we will create a word frequency table that forms a dictionary. In this, we will not be using stop Words array to summarize, but instead, we will use only the words. This technique is applied to the text_string from the article or textbooks or emails. Then, the text_string is tokenized into a set of sentences by using nltk. These sentences are given score in Term Frequency Method and by fixing the average threshold score for sentences that are considered. Finally, we choose a sentence which has a high score than the average rating, and it is summarized. [2] An unbiased prediction technique that uses unsupervised learning to explore text summarization on emails in different languages like Dutch, French, English and many other languages by using a python programming language. Initially, the signatures from the emails are removed then by using the python libraries like polyglot, text blob and langdetect to identify in which language the mail is there. Then by using nltk sentence tokenizer, the emails are split into basic sentences by following some rules. The sentences in each mail must have fixed length vector representations to encode the inherent semantics and meaning of the sentence. The word embeddings for those sentences can be generated by using Skip-Gram Word2Vec method, and those sentences are clustered by using a high-dimensional vector space. The number of sentences in summary, which is equal to the number of clusters will be the square root of the total number

of sentences, and these clustered sentences are interpreted as semantically similar sentences. This summarization can also be called an extractive summarization technique. [3]

III. PROPOSED METHODOLOGY

In this paper, we are introducing an Automatic Text Summarization system that uses Extractive Summarization Technique in which the text in the articles or books or emails or any research papers is summarized by using Text Ranking Algorithm. These Text Ranking Algorithm will be working similar to the that of Page Ranking Algorithm that impressed. [4]

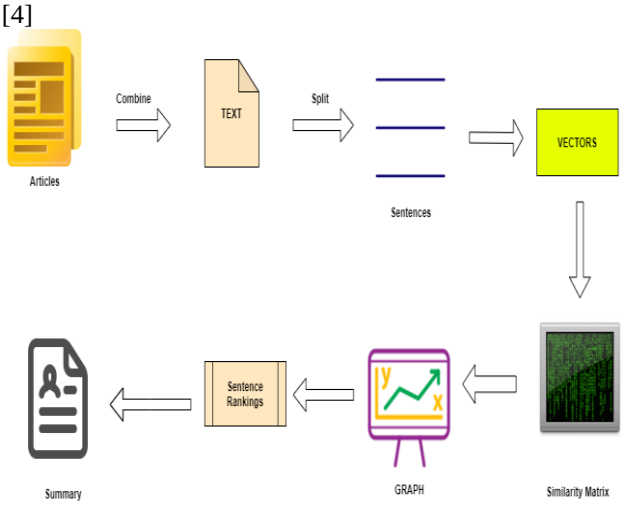

Fig 1: Text Summarization Technique

The text inside the articles is combined and split the text into sentences. Later, find the word embeddings for those sentences along with the similarity matrix. The similarity matrix is then represented in the graph format by considering the similarity scores by which we can calculate the sentence rankings. At last, the top-ranked sentences are regarded as the summarized texts.[5][6]

IV. IMPLEMENTATION

Before starting how the summarization process works, we need to install Jupyter Notebook platform for coding using python along with the dataset by which we will be training the model and then test it. Some necessary and essential libraries that are required to perform text summarization are NumPy, Pandas, nltk, re (Regular Expression). The NumPy and Pandas libraries are imported as np, pd in our python programming language. After importing that libraries, we need to read the dataset which is in CSV format and we can read this CSV file/dataset by using read_csv () function. Suppose, if we want to see the text inside the CSV file, we can read it by using an object along with the column name and row number in this format (Object Name [Column Name] [Row Number]).By using this technique, we can do either individual article summarization or all materials single summary. Nevertheless, this thing will be explained by the end of the working process. Now, the text taken from the article will be split or tokenized into sentences by using nltk library function sent_tokenize (). We use Glove Word Embedding to represent our sentences in the vector form, which is trained Wikipedia 2014+Gigaword 5glove vectors. To make the obtained data noise-free as much possible, we preprocess the text and perform the text cleaning operation. However, these texts will be having stop words like -am, -is, -the, -in etc. to eradicate that stop words we will be using nltk stop words library that has the capability of removing all the stop words from the dataset that we are using from the beginning. The function used to remove the stop words from the text is by using remove_stopwords (r. split ()) for r in clean sentences).

Now, by using Glove word Vectors, the vectors will be created to the sentences and average of those vectors are taken to consolidated vector. To these consolidated vectors the similarities between the sentences are calculated where we will be using cosine similarities for those sentences, and cosine similarity score will be given for each of these sentences and by applying the page ranking algorithm to get the ranks of those sentences. Finally, based on the rankings given, extract only top N sentences.

V. RESULTS

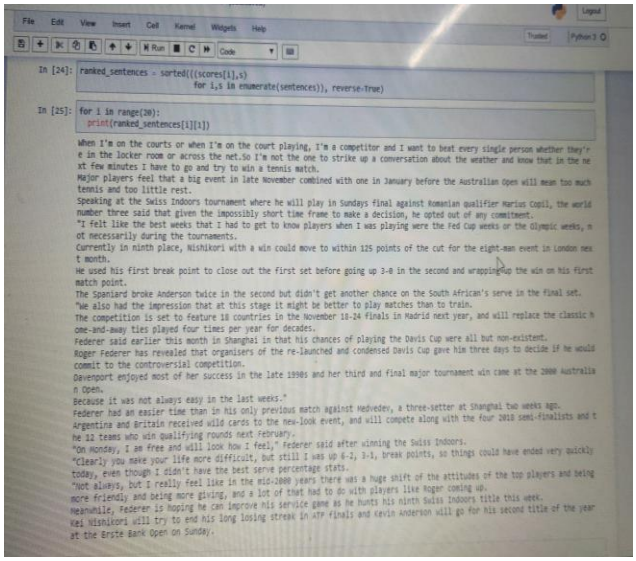

Fig 2: Summarized Sentences from the CSV File

VI. CONCLUSION

Automatic Text Summarization is a successful application which can be used for boosting up your business. A large amount of text can be summarized within a fraction of seconds. The proposed system will be trained and tested by using the dataset. The cosine similarity will be used for ranking the sentences based on the similarities in the sentences. In the future, we can implement the automatic text summarization process by using RNNs (Recurrent Neural Network), LSTM (Long Short-Term Memory), Reinforcement Learning and Generative Adversarial Networks (GANs).

Retrieval Number: D7278049420/2020©BEIESP
DOI: 10.35940/ijeat.D7278.049420
Journal Website: www.ijeat.org

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

REFERENCES

1. M F Porter, “An Algorithm for Suffix Stripping”, No.03, PP 130-137, July 1980.
2. Akash Panchal, “Text Summarization in 5 steps using NLTK: Word Frequency Algorithm”, Medium, Jan 22, 2019.
3. Aishwarya Padmakumar, Akanksha Saran, “Unsupervised Text Summarization Using Sentence Embeddings”.
4. H P Luhn, “The Automatic Creation of Literature Abstracts”, IRE National Convention, March 24, 1958.
5. H P Edmundson, “New methods in Automatic Extracting”, Vol 16, No.02, April 1969.
6. Prateek Joshi, “An Introduction to text summarization using the text rank algorithm (with python implementation)”, Analytics Vidhya, Nov 1, 2018.

AUTHORS PROFILE

Manikanta K.B, currently working as an assistant professor in GITAM University, Bengaluru. He is having 4.5 years of experience in teaching field and 2 years’ experience in IT field. Previously he was working in BITIT. He worked as a data analyst and system admin in Manastha solutions and dimension data respectively. He has published 5 research papers in UGC journal. He is very much interested to research in areas like cloud computing, IOT and machine learning.

Bhagavath Sai M, is a student pursuing his B. Tech Final year in computer science and engineering in GITAM (Deemed to be a university) Bangalore, Karnataka, India. Bhagavath has developed an app called Alert app in thinkable software which mainly designed for women safety purposes. He is also a GUSAC club member at GITAM University And he has done MTA certification in machine learning using python. And He is currently working in Wipro and he is working on AWS.

Venkat I, is a student pursuing his B. Tech Final year in computer science and engineering in GITAM (Deemed to be a university) Bangalore, Karnataka, India. Venkat has done internship in Bangalore based on Machine Learning with Python. And worked with WebTech Labs in Kolkata and developed An E-commerce Application where there will be Clothing, Accessories. He is also a GUSAC club member in GITAM University. And using course in Android Application developed an Application Alert App Which used Thinkable Platform the main purpose of app is to save women life.

Sreekar Reddy B, is a student pursuing his B.tech Final year in computer science and engineering in GITAM(Deemed to be a university)Bangalore, Karnataka, India. Sreekar has developed An app called Alert app in thunkable software which mainly designed for woman safety purpose. He is also a GUSAC club member in Gitam university And he has done Java certification from HCL Technologies and Penetration Testing.