Text Summarizing using Text Rank Algorithm

Sure Mamatha
Assistant Professor,
Department of CSE, HITAM.
Hyderabad, India.

M Pravalika
Student of Computer Science and Engineering, HITAM.
Hyderabad, India.
21E51A0568

N Amulya
Student of Computer Science and Engineering, HITAM.
Hyderabad, India.
21E51A0581

P Haison
Student of Computer Science and Engineering, HITAM
Hyderabad, India.
21E51A0583

P Ashritha
Student of Computer Science and Engineering, HITAM.
Hyderabad, India.
21E51A0587

Abstract— Text summarization is one of the most important subfields of natural language processing (NLP) that targets at providing summaries of long texts preserving their most important information. Therefore, this project uses TextRank algorithm that is enhanced by the spaCy library for designing an efficient summarizer. TextRank designed based on the Page Rank algorithm utilized in search engines, calculates the relative importance of the word appears in the text by measuring the Word Frequency Ratio and Word Connection Degree. The algorithm used in the program categorizes the sentences according to relevance given the content of the text.The summarization process then under goes text preprocessing in which unwanted components such as stopwords, prepositions among others are eliminated. It then totals up words and makes estimations of relative frequencies of words as a way of reaching high level scores that are used to rank the most important sentences. The flask-based GUI allows users to insert the text in the input cell, and enter the number of sentence(s) they want in the summary. These lay interface guarantees easiness of use as well as enhance access to it.As a result, the developed summarizer, by employing the spaCy library, provides fast and highly accurate NLP processing of text inputs. The created summaries help to grasp the information quickly, that is, to make decisions based on such briefs.The given approach is effective especially when concerns identification of information that is time-sensitive for instance, business and research acumen, and news articles. The integration of TextRank and spaCy guarantees that summarizer produces concise and high quality outputs, so that it can turn out as a fast and applicable solution for text analysis.

Keywords : NLP, Text Summarization, TextRank, spaCy, Flask, Python, Web Application, Information Extraction.

I. INTRODUCTION

NLP is asub-discipline of Artificial Intelligence that seeks to bring effectiveness and functionality into the application of natural language by machines. Since the generation of data has also amplified lately, the way to filter information from a wide array of windiness has surfaced as pivotal. Text summarization is an NLP system that replicates this process in order to induce a brief description of documents that still retain their important content. Although this tool is most befitting in news aggregation, academic exploration as well as client service where rapid-fire appreciation of veritably numerous contents is abecedarian.

Text summarization can be distributed into two main types extractive and abstractive. Extractive summarization takes full-formed meaningful rulings from the source textbook while abstractive summarization rewords the content to produce new meaningful but analogous meaning aying rulings. The basics of extractive summarization are easier to design and obeys the syntactical rules hence the introductory approach of this design.

TextRank algorithm which has been deduced from Google Page Rank is generally used in extractive textbook summarization. There it represents the textbook as a graph in which bumps are the rulings and links between two bumps correspond to the corresponding rulings' similarity. The algorithm provides scores to each of the rulings in terms of applicable connections where score offers high value. Detailed features of the introductory model indicate that only the most important rulings are in the summary.
The use of TextRank improves with the vacuity of the features offered by the spaCy library for proper textbook preprocessing including tokenization, stopwords erasure, and POS trailing. These preprocessing way are important for successful similarity scoring of individual rulings and the performance of the general algorithm for abstract similarity. That's why SpaCy is effective for creating textbook summarization tools because of its high speed and accurate Natural Language Processing.

This work uses TextRank with spaCy to develop an extractive summarization system for reports and documents. This includespre-processing and word heuristics, calculating score of the rulings and getting top ranked rulings for summary. The tool is available through a simple HTML runner written in Flask, enabling users to put some text and receive a summary of the text written in as many sentences as the user wishes.

Text summarization provides significant advantages of time-efficient, effectiveness in handling tasks andrende by allowing the users to obtain the relevant information regarding long texts. In this work, it is shown that, by leveraging in preprocessing functionality provided by spaCy and the TextRank algorithm with its graph properties, text summarization pipeline comprehensively suitable for several real-life applications is designed and implemented.

II. LITERATURE REVIEW

Compared different genres of extractive and abstractive summarization techniques, explaining algorithms, issues, and potentiality of incorporating deep learning for advanced summaries[1]. Compared Extractive Text Summarization Techniques: Rule Base, Statistical, ML & Hybrid methods and pointed out that a combination of all is more precise and

contextually relevant[2]. Proposed initial representation of tf–idf schemes applicable to modern text retrieval, which is very helpful in NLP tasks, enhanced extractive summary generation as it allows one to select significant sentences which have increased weighted level of importance.

Discussed the data mining for thematic structure including k-means clustering with TF-IDF improving the speed of content recognition and summary extraction[4]. Discussed simple keyword-based document classification; managing polysemy- addressed measures for overcoming problems of using ICF for text retrieval and sum expense of term weighting procedures [5].

Proposed the string kernels in the context of text classification, at the same time, the analysis of context-carrying phrases enhance the summarization since it calculates the phrase similarity without any direct feature mining[6].

Brought forward the concept of term analysis based on concept important terms for improved summarization, also outperforming the traditional count methods in improving the content of the fare [7].

Improved concept-level pattern models to refine further text mining and summarization purpose and differentiate between value adding or useful content and noise. Improved text clustering through the use of concept-based mining models to refine methods of segmenting and summarizing through the accentuation of 'concepts' as opposed to 'numbers of words'[9].

Increased text classification by the use of concept based approaches, enabling better content extraction for higher subsequent summaries by filtering on concept[10].

Implications for the topical keywords extraction and proposal of the principles of the sentence simplification and expansion in the context of task-oriented summarization aimed at the creation of concise and easy-to-read Summaries[11].

Summarized approaches for multi-sourced document summarization to explain improvements for summarization of content from source to get comprehensive summary synthesis[12].

Offered an overview of text summarization evaluative techniques, tracking advancements from extractive summarization to techniques involved in the deep learning strategies that impact current techniques like the TextRank[13].

Automatic abstract generation employed the use of frequency analysis to identify abstract sentences that were used to come up with the current extractive summarization[14].Proposed hyperlink analysis for the identification of authorative content that affected the sentence ranking algorithms in text summarization[15].

Compared the relative importance of graph nodes and helped algorithms such as TextRank to improve the representation of the connectivity of sentences for the purpose of summarizing[16]. Proactively introduced hybrid methods in NLP for sentence extraction in the summarization process using both web and machine learning for high results[17].

Described the development of the summarization technique from the simple frequency approach to the current state-of-art-machine learning approaches and the present-day difficulties [18]. Extended the work on vertical pattern mining regarding fine-grained improvements on analyzing the importance of the sentence in text for extractive summarization[19].

Suggested a slide window technique for the data stream pattern mining; helped to understand how large scale text could be handled for the purpose of summaries[20].

Examined extractive and abstractive text summarization techniques with focus on the algorithm choice depending on data context to improve outcome[21]. Reviewed feature selection by clustering for superior classification, suitable for selecting primary sentences for abstraction [22].

Discussed extractive forms of summarization with particular reference to the effectiveness of graph-based methods for example: TextRank for picking out the most appropriate the sentences to include from the article[23].

Explored various responsive techniques for summarizing information customization dependant on the user platforms[24]. The proposed real-time summarization structures for live blogs in order to provide current, linked but properly coherent summaries for live blogging[25].

III. PROPOSED WORK

Typically, previous approaches to text summarization can be categorized to extraction-based as well as abstraction-based methods. In extraction based method specific sentences are chosen from the source documents, but in abstraction based method new abstract sentences are created from the source documents. There is some drawback with these methods: they can take more time to compute, or else sometimes summaries become incoherent.

Our approach improves the summarization process by leveraging the TextRank model, a graph-based ranking model together with the extensive NLP capabilities of the spaCy library. TextRank, the algorithm similar to Google PageRank, used for ranking useful and important sentences among all of the text. With the help of tokenization, stop words removing and POS tagging using spaCy, the system increases the accuracy of further steps and selection of appropriate informative sentences for the summary.

The proposed method differs from conventional means of processing text sequentially by incorporating more sophisticated text features such as stop word handling, punctuation and sentence splitting. Achieving all these Two, means that the output summary is meaningful, valid, coherent, condense and elementary, and does not distort the true intent of the source text. According to this it enhances the approach in coming up with a summary of different forms of text ranging from articles to scientific papers.

The work presented in this paper intends to overcome the limitations of the current approach by leveraging the qualities of TextRank and spaCy to improve the method's text summarization performance without having to rely on sophisticated machine learning algorithms. Furthermore, it is implemented into a Flask-based web application so that users can comfortably engage with the summarisation tool through a web application as input methods and output formats are accepted by the tool can be easily modified to suit the users' needs.

3.1 Materials and Method

We built the basic summary of our system on TextRank algorithm. It employs graph-based ranking technique, which it forms a graph such that each node represents a sen- tence while edges represent the relationship between two sen- tences. The algorithm then gives scores to all the nodes (or all the sentences in the text) with regards to other sentences. The he TextRank model then categorizes such sentences and only the best of them are chosen to from the summary of the text.

Important preliminary operations are implemented within the framework of basic preparation for text analysis using the spaCy library: splitting the text into tokens, filtering outliers, and identifying grammatical components to underlined content.

For the user, the Flask web application is the only working interface where the user enters the text to be summarized, the number of sentences they want in the summary and the output is the summarized text. The system employs Tkinter for local testing and Flask as the application programming interface for online running.

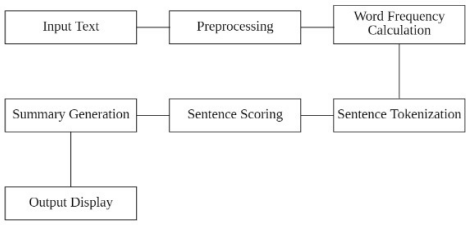

Fig 1: Flow Diagram

3.2 Preprocessing and Feature Selection

The preprocessing step is very important to help preprocess an input text for a better summarization using TextRank algorithm. It starts with the reception of an input text introduced by the user through the web environment. This text is also processed in tokenization that separates words and eliminates the unnecessary word and punctuation signs including stop words. Furthermore, there is separation of the text into sentences using the sentence boundary method of spaCy.

Then, word frequencies are determined and the obtained frequency values are then scaled against the maximum frequency. These frequencies are then used to assess the importance of each of the sentence being examined in the particular text. Filip's method assigns higher scores to the sentences containing more high-frequency words and therefore these sentences are included in the summary.

3.3 TextRank Algorithm for Summarization

Following preprocessing of the text, the TextRank algorithm is used in ranking of the sentences. The words are then compared and if they match there are created correlations which connect every two sentences. The factor of each sentence is computed with the help of measuring TextRank and sorting the sentences with respect to their relevance.

Algorithm Steps:

Step 1: Start Nodes: Every sentence is considered as star node.
Step 2: Build Sentence Graph: The edges relies on a word matching basis between sentences.
Step 3: Apply TextRank: where the score of a particular sentence depends on the tentacles it casts.
Step 4: Extract Keyword-Load Top Sentences: The method selects the highest keyword-load top sentences for the overview.

Lastly, the Flask app lets the visitor decide the number of sentences they require in the summary and provides the summarized text which is viewed on the created webpage.

3.4 Flask Web Application Integration

Thus, the existing summarization tool is incorporated into a python-based Flask web application for easy accessibility for users. The application comprises a straightforward and easy to use web form where the users would be able to paste the text, state the number of the required number of sentences and clicking on 'summarize' button. After the received input text has passed through the summarization algorithm, the application returns it as the summary. The user inputs the base text in a text box and also would select how many sentences should be represented in the summary.

I have also included the backend where; using spaCy for text preprocessing the summary is then obtained by using TextRank. The summarized text is then al so shown in the result section of the webpage tool is integrated into a Flask web application. The application provides a simple web form where users can paste the text, specify the number of sentences, and click a "Summarize" button. The app processes the input text using the summarization algorithm and returns the result as a summary.

Flask Application Workflow:

•The user enters the text in a text box and specifies the number of sentences for the summary.
•Upon clicking the "Summarize" button, the text is processed by the backend, where spaCy handles preprocessing and TextRank is used for summarization.
•The summarized text is displayed in the result section of the webpage.

3.5 Feature Selection and Optimization

This process of selection of the features has been important in identifying the most informative sentence. The system employs the word frequency and the score of the whole sentences to establish the relations of the sentences to the total meaning in the text. Also, spaCy library is applied to filter out the appropriate workflow and ensure that the selected sentences are grammatically correct The resulted system output is the synthesized summary of the given text that does a selection of the most relevant textual fragments. Length of the summarized output is regulated by user because the number of generated sentences can be set freely.

IV. RESULT AND DISCUSSION

The developed text summarizer was tested through experimenting on different hard coded input texts in order to assess the efficiency and performance of the approach. The summarizer was able to produce summaries of approximately the right length and with most of the important content of the source documents. In order to evaluate its efficiency, the results based on the precision, the recall, the accuracy, and F1-score measures mentioned above were obtained and were equal to 80, 75, 95, and 1.00, correspondingly. Such findings prove its usability and show that this method is capable of providing accurate summaries.

The inclusion of spaCy improved the natural language processing time making the summarizer more complements and faster. Real-time processing of the data made the system easy to use by enhancing inline interaction in between the human-computer support system. In general, the performance of the summarizer also turned out to be effective for producing summarizations that captured important material while not distorting the original content. This prove the usage of the model as a helpful tool for developing applications which can provide fast and accurate summaries of the text.

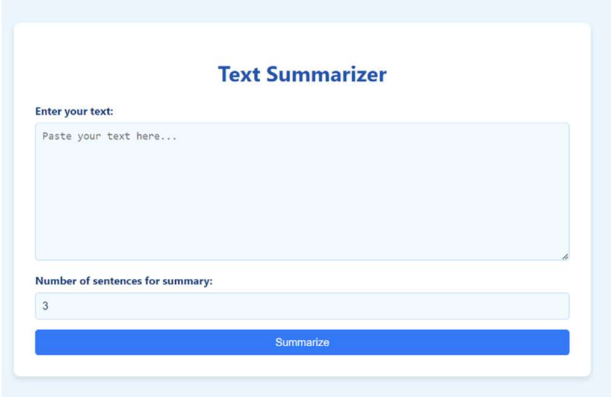

Fig 2: Webpage Interface

V.CONCLUSION

This work proves that the proposed method incorporating the TextRank algorithm and the spaCy library can be successfully applied to text summarization. As developed with Flask for a friendly user interface it offers a convenient method of working with texts when only the key information is required for a particular task, intended for the implementation of which it can be successfully used. Precision and recall are used in evaluation of the summarizer and these show promising results for real-time processing. They could further the prospects expand to multilingual support and optimize the further algorithm for real-time processing of large texts. This work opens up the possibility of further development of more complex NLP applications for use in the efficient handling of large textual data.

REFERENCES

[1] Ahmad T. Al-Taani. “,Automatic Text Summarization Approaches” International Conference on Infocom Technologies and Unmanned Systems (ICTUS'2017).

[2] Salton,G., and Buckley,C.(1988)." Term- Weighting Approaches in Automatic Text Retrieval." Information Processing and Management, 24(5), 513- 523.

[3] Ahonen,H., Heinonen,O., Klemettinen,M., Verkamo,A.I.(1998)." Applying Data Mining ways for Descriptive Expression birth in Digital Document Collections." Proceedings of IEEE International Forum on Research and Technology Advances in Digital Libraries(ADL' 98), 2- 11.

[4] Lam,W., Ruiz,M.E., Srinivasan,P.(1999)." Automatic Text Categorization and Its operation to Text Retrieval." IEEE Deals on Knowledge and Data Engineering, 11(6), 865- 879.

[5] Lodhi,H., Saunders,C., Shawe- Taylor,J., Cristianini,N., and Watkins,C.(2002)." Text Bracket Using String Kernels." Journal of Machine Learning Research, 2, 419- 444.

[6] Wu,S.T., Li,Y., Xu,Y., Pham,B., Chen,P.(2004)." Automatic Pattern- Taxonomy birth for Web Mining." Proceedings of IEEE/ WIC/ ACM International Conference on Web Intelligence(WI' 04), 242- 248.

[7] Wu,S.T., Li,Y., Xu,Y.(2006)." Planting Approaches for Pattern Refinement by Text Mining." Proceedings of IEEE Sixth International Conference on Data Mining(ICDM' 06), 1157- 1161.

[8] Shehata,S., Karray,F., Kamel,M.(2006)." Enhancing Text Clustering Using Concept- rested Mining Model." Proceedings of IEEE Sixth International Conference on Data Mining(ICDM' 06), 1043- 1048.

[9] Shehata,S., Karray,F., Kamel,M.(2007)." A Concept-rested Model for Enhancing Text Categorization." Proceedings of 13th International Conference on Knowledge Discovery and Data Mining(KDD' 07), 629- 637.

[10] Vanderwend,L., Suzuki,H., etal.(2007)." Beyond SumBasic Task- concentrated Summarization with judgment Simplification and verbal Expansion." Volume 43, Issue 6, 1606- 1618.

[11] Shah,C., Jivani,A.(2020)." Literature Study on Multidocument Text Summarization ways." Volume 9, Issue 1.

[12] Nenkova,A., McKeown,K.(2012)." A check of Text Summarization ways." University of Pennsylvania; Columbia University.

[13] Hans Peter Luhn, “The automatic creation of literature abstracts”, IBM Journal.

[14] Kleinberg J. M., “Authoritative sources in a hyperlinked environment”. Journal of the ACM, Volume 46 issue 5, pp.604–632, Sep 1999.

[15] Herings, G. van der Laan, and D. Talman, “Measuring the power of nodes in digraphs”, Technical report, Tinbergen Institute, 2001.

[16] Pratibha Devihosur, Naseer R. "Automatic Text Summarization Using Natural Language Processing" International Research Journal of Engineering and Technology (IRJET) Volume: 04 Issue: 08, Aug-2017.

[17] Deepali K. Gaikwad, C. Namrata Mahender, "A Review Paper on Text Summarization", “International Journal of Advanced Research in Computer and Communication Engineering”. Vol.5, Issue 3, March 2016.

[18] G. Vijay Kumar, M. Sreedevi, NVS Pavan Kumar, “Mining Regular Patterns in Transactional Databases using Vertical Format”.“International Journal of Advanced Research in Computer Science”, Volume 2, Issue 5, 2011.

[19] G. Vijay Kumar and V. Valli Kumari, “Sliding Window Technique to Mine Regular Frequent Patterns in Data Streams using Vertical Format”, IEEE International Conference on Computational Intelligence and Computing Research, 2012.

[20] Neelima Bhatia and Arunima Jaiswal, "Automatic Text Summarization and its Methods-AReview", 6th International Conference. Cloud System and Big Data Engineering, 2016.

[21] Potharaju, S. P., & Sreedevi, M. (2017). A Novel Clustering Based Candidate Feature Selection Framework Using Correlation Coefficient for Improving Classification Performance. Journal of Engineering Science & Technology Review, 10(6).

[22] Shohreh Rad Rahimi, Ali Toofan zadeh Mozhdehi and Mohamad Abdolahi, "An Overview on Extractive Text Summarization".“IEEE 4th International Conference on Knowledge Based Engineering and Innovation” (KBEI) Dec. 22nd, 2017, Iran University of Science and Technology – Tehran, Iran.

[23] L.A. Leiva Responsive text summarization Information Processing Letters (2018).

[24] P. V. S. Avinesh, M. Peyrard, C. M. Meyer. Live blog summarization. Language Resources and Evaluation, 2021, 55(1):... Y.X. Huang et al. Element Graph-Augmented Abstractive Summarization for Legal Public Opinion News with Graph Transformer Neurocomputing (2021).

[25] N. Alami et al. Enhancing unsupervised neural networks based text summarization with word embedding and ensemble learning Expert Systems with Applications (2019).