Published by :
https://www.ijert.org/
An International Peer-Reviewed Journal

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 10 , October - 2026

Context-Aware Hybrid Text Summarization System With Customizable Summary Length

Dhanushree K | Manya D | Spoorthi A B
4MG23CS017 | 4MG23CS032 | 4MG23CS055
Department of Computer Science and Engineering
G. Madegowda Institute of Technology
Bharathinagara, Maddur Taluk, Mandya, Karnataka, India

Mrs. Sneha M
Assistant Professor
Department of Computer Science and Engineering
G. Madegowda Institute of Technology
Bharathinagara, Maddur Taluk, Mandya, Karnataka, India

***Abstract*—** The With increased volumes of digital documents, it can be difficult to identify the key points within a document. The aim of this project is to create a context aware hybrid text summarization system with adjustable summary length. The system uses both TextRank based extractive methods and Transformer based abstractive methods. TextRank will be used to extract the most important sentences, before passing them into either a BART, PEGASUS or T5 model. During refinement, the summary should be reduced with little repetition whilst maintaining the original context. Using a Streamlit interface the user will be able to choose from short, medium or detailed summaries. From reviewing the project design, it was identified that Python would be used along with Natural Language Processing libraries and SQLite. This paper will outline the architecture, methodology and reproducible evaluation methodology using ROUGE.

***Keywords*—** Automatic Text Summarization, Natural Language Processing, TextRank, BART, PEGASUS, T5, Hybrid Summarization, Context Preservation, Removal of Redundancy, Adjustable Summary Length, ROUGE.

I. INTRODUCTION

Every day many digital platforms will publish research articles, reports, news articles, learning materials and many more texts. Whilst the contents can be useful, it can take a long time to read through every document. An automatic text summarization approach can be used to create a shorter version of the text, whilst keeping the key information.

Generally, text summarization can fall into two categories; extractive or abstractive. Extractive methods extract specific sentences or phrases from the original text. Abstractive methods generate new text to explain the meaning in the source text. TextRank is a graph-based extractive methodology used to rank sentences based on their relationship to others [1] [2]. Transformer models have helped with abstractive methodologies through their use of context and sequence to sequence learning.

Using extractive methods can help keep the original wording from the source text. However, it can lack flow due to the extraction of specific sentences. Abstractive methods can improve the flow of the text, however the generated text needs to be checked to ensure there is no unsupported information, missing information or repetition. Hybrid text summarization uses both extractive and abstractive methods to overcome these issues [3] [4].

This project aims to create a Context Aware Hybrid Text Summarization System with Adjustable Summary Length.

TextRank is used to extract important sentences, before passing them into one of the abstractive models chosen by the user; BART, PEGASUS or T5. The output will be refined to reduce any repetition.

II. RELATED WORK

Dumne et al created an automatic text summarization approach using TextRank [1]. The sentences are converted into a graph and ranked using a PageRank like approach to identify the most important sentences. This shows how useful graph-based methods can be for extractive methods.

Luthra and Maheshwary performed a survey on the advancements in text summarization. This includes extractive and abstractive methods, datasets, evaluation methods, challenges and future research [2]. This also shows how important it is to create readable, coherent, relevant, comprehensive and non repetitive summaries.

Deo and Pattnaik performed a survey on abstractive, extractive and hybrid text summarization [3]. This provides a conceptual understanding of how extractive methods can be combined with abstractive methods. The quality of the hybrid summarization depends on the information provided from the extraction and the generation.

Ashwini et al discussed different automatic text summarization methods, including feature based methods to identify important content. This project was built on these studies, using TextRank and the option to use a variety of Transformer based models to generate the summary. [4]

This project aims to combine extractive methods, alternative abstractive models, refinement of context and removing redundancy along with short, medium and detailed outputs in one application. This must be proved effective through controlled experiments.

III. PROPOSED SYSTEM

The source text will be inputted into the application. The text will be preprocessed and TextRank will be used to rank the sentences based on their importance. Once the most important sentences have been extracted, this will be inputted into one of the abstractive models chosen by the user (BART, PEGASUS or T5). The output will be refined to reduce any repetition or highly similar text. The summary will be adjusted to meet the desired output preference. The final

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 10 , October - 2026

summary will be shown in the web interface.Overall the text goes through the following: Input Document —>Preprocess —>TextRank —>Important Content —> (BART / PEGASUS / T5) —>Context Refinement —> Remove Redundancy —>Adjustable Summary Length —> Final Summary. The three Transformer models are alternative methods and for a fair comparison the same input documents, extraction and evaluation methods will be used.

IV. METHODOLOGY

A. Input

The user will input the source text. The project includes inputting documents. If any additional file types (i.e. PDF/DOCX) can be uploaded, this must be mentioned..

B. Preprocessing

The text will be normalized. Unwanted formatting and extra white space will be removed. The text will then be broken down into sentences. This list will be inputted into the extractive method. If stop words have been removed or stemming has been used, this must be done consistently.

C. TextRank Sentence Extraction

The text will be represented as a graph. Each sentence will be represented as a node. An edge will connect sentences that match the criteria based on their similarity. The importance of each sentence will be calculated iteratively using a PageRank like approach. This can be represented as:

$$S(V_i) = (1 - d) + d \sum_{j \in In(V_i)} [w_{ji} S(V_j) / \sum_{k \in Out(V_j)} w_{jk}]$$

Where S(Vi) is the score of sentence i, d is the damping factor and w is the similarity weighting between sentences. Once converged the sentences with the highest scores will be extracted. This will depend on the extraction ratio or maximum sentences. The sentences will be put back into the original text order.

D. Abstractive Summary

The important content extracted by TextRank will be inputted into one of the alternative Transformer models. BART uses denoising sequence to sequence pre-training. PEGASUS uses a pre-training objective designed for abstractive summarization. T5 uses text to text transformations for language tasks. The summary will be generated using the model chosen by the user. Each model will be tested individually.

E. Context Refinement and Removal of Redundancy

The summary will be checked to ensure there is no repetition. In practice this could involve checking the similarity of sentences and removing any sentences that have a lot of repeated information. The context refinement should ensure any entities, key facts, relationships and logical order from the source text. As the post-processing is automated there is no guarantee the information is accurate, so the source text should be checked against the summary.

F. Adjustable Summary Length

There are three options for the output; Short, Medium and Detailed. This will be done by adjusting the generation parameters and/or limiting the post-processing. The exact number of words/sentences should be mentioned.

G. Implementation

The project will be implemented using Python. The web interface will use Streamlit. Text will be processed using Natural Language Processing libraries (i.e. NLTK). Local data will be stored using SQLite. The development environment will use Visual Studio Code. The Transformer models can be loaded using a library (i.e. Hugging Face Transformers) with a supported Deep Learning backend. The final implementation should state what packages, model checkpoints and versions were used.

V. SYSTEM ARCHITECTURE

The following Fig. 1 shows the proposed methodology. The input document will be preprocessed. TextRank will be used to rank the sentences. The important content will be sent to one of the alternative abstractive models. Context refinement and removal of redundancy will be performed. The summary length will be adjusted to meet the desired output..

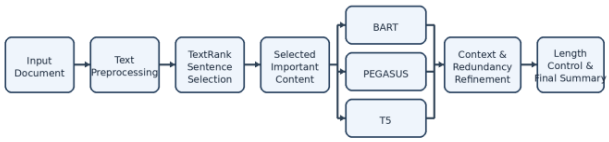


Fig. 1. Proposed architecture of the context-aware hybrid text summarization system.

VI. EXPERIMENTAL SETUP AND EVALUATION

The evaluation should be reproducible. This will involve having a set of source documents and reference summaries written by humans. The name of the dataset, number of documents, domain, language, training/test and any preprocessing rules should be mentioned. Each document will be processed using the same TextRank settings. Each document will then be processed using BART, PEGASUS and T5. The summary lengths will be set to Short, Medium and Detailed.

ROUGE will be used to evaluate the generated summaries compared to the reference summaries . ROUGE-1 calculates the unigram overlap, ROUGE-2 calculates the bigram overlap and ROUGE-L calculates the longest common subsequence. The precision, recall and f1 scores should be mentioned for each metric. How the scores were averaged should be mentioned. ROUGE scores only calculate the lexical overlap, so human evaluation should also be done to check the factual consistency, comprehensiveness, coherence and readability.

No benchmark dataset, reference summaries, complete ROUGE scores, timings or comparison outputs were provided with the project materials. So no numerical values can be mentioned. This evaluation should be used as a protocol for future testing.

VII. RESULTS AND DISCUSSION

The project presentation and application screenshot indicate a Streamlit-based interface containing controls for summary type, AI model, summary length, and output language, together with document statistics and a generated-summary area. These interface elements demonstrate the intended user interaction and output presentation.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 10 , October - 2026

The Hybrid Text Summarization system evaluates the quality of generated summaries using ROUGE-1, ROUGE-2, and ROUGE-L metrics by comparing them with reference summaries. It also calculates and displays the execution time required for summary generation. The system supports three summarization approaches: Extractive, Abstractive, and Hybrid, using three transformer models: BART, PEGASUS, and T5. Users can evaluate different models and summary-length settings by analyzing ROUGE scores, processing time, and generated outputs. These results are stored in the system's history, allowing users to review and compare previous summarization results..

VIII. LIMITATIONS

The quality of the output will depend on the quality and length of the input text, TextRank extraction ratio, chosen Transformer checkpoint and length of input for the model. Long documents may need to be chunked or split into sections. The abstractive models may miss out facts or include unsupported information. The removal of redundancy based on the similarity may also remove useful repetition. Fact checking will be needed. The system should be systematically evaluated. If multilingual outputs and additional file types can be handled this should be mentioned

.IX. CONCLUSION AND FUTURE WORK

This paper proposes a context aware hybrid text summarization system using TextRank for sentence extraction and alternative abstractive models (BART, PEGASUS or T5). The methodology involves preprocessing the text, extracting important content, context refinement, removing redundancy and adjusting the output to Short, Medium or Detailed based on the users choice. The system uses Streamlit to provide a practical interface to interact with the summarization pipeline. The aim is to make long

documents easier to review and give the user control over the amount of information.

Future work will include completing benchmarking using ROUGE and human evaluation. Improving the factual consistency. Chunking to handle longer documents. Adding multilingual capabilities that have been tested. Additional document types can be accepted. Word limits can be set by the user. A larger and varied dataset will help to compare the three abstractive models fairly.

REFERENCES

[1] R. Dumne, V. N. Waghmare, N. L. Gavankar, and M. M. Bokare, “Automatic Text Summarization Using Text Rank Algorithm,” in Proc. 2024 3rd International Conference for Advancement in Technology (ICONAT),2024,doi:10.1109/ICONAT61936.2024.10775241.

[2] J. Luthra and P. Maheshwary, “A Survey of Advances in Text Summarization Methods,” in Proc. 2024 International Conference on Advances in Computing Research on Science Engineering and Technology (ACROSET),2024,doi:10.1109/ACROSET62108.2024.10743404.

[3] S. Deo and P. K. Pattnaik, “A Short Survey of Abstractive, Extractive and Hybrid Text Summarization,” in Proc. 2025 International Conference on Cognitive, Green and Ubiquitous Computing (IC-CGU), 2025, doi: 10.1109/IC-CGU67042.2025.11337962.

[4] A. K. M. Ashwini, S. S. Shreyas, and S. K. Shreyas, “Text Summarization Using Automatic Text Summarization,” in Proc. 2025 9th International Conference on Computational System and Information Technology for Sustainable Solutions (CSITSS), 2025, doi: 10.1109/CSITSS67709.2025.1129558

(This work is licensed under a Creative Commons Attribution 4.0 International License.)