Saaramsha: Leveraging NLP for Efficient Kannada Text Summarization

P. Maharshi Reddy
Department of Information Science Engineering
BMS College of Engineering
Bangalore, Karnataka, India

Lalam Aakash
Department of Information Science Engineering
BMS College of Engineering
Bangalore, Karnataka, India

Likhithraj A.
Department of Information Science Engineering
BMS College of Engineering
Bangalore, Karnataka, India

Rashmi K.B.
Department of Information Science Engineering
BMS College of Engineering
Bangalore, Karnataka, India

ABSTRACT
Summarizing involves condensing a text while retaining its key points. Extractive summarizers focus on identifying important sentences from the text to convey its message effectively. They typically operate by identifying keywords and selecting sentences containing those keywords prominently. Keyword extraction entails identifying significant words with higher frequencies, particularly emphasizing important ones. In this system, a TF (Term Frequency) model and GSS coefficients were employed to extract keywords and rank text. The algorithm automatically extracts keywords for summarizing texts in Kannada datasets.

Keywords

Kannada text summarization, Extractive text summarization, Tokenization, Stop word removal, Sentence ranking.

1. INTRODUCTION
Text summarization is a vital technique in natural language processing (NLP) that aims to distill the most essential information from a document, thereby providing a concise version that retains the core message. This process can be highly beneficial in numerous applications, including news aggregation, research synthesis, and content curation. By leveraging machine learning (ML) methods, summarization models can learn to identify and extract key sentences and concepts from large volumes of text, significantly reducing the time and effort required to comprehend lengthy documents. The effectiveness of these models is typically evaluated using metrics such as F1 score, accuracy, and recall to ensure they produce relevant and coherent summaries. In an era of digital transformation aimed at enhancing and simplifying processes, reducing paperwork, and improving user experience, the "RAILEASE-Railway Concession Automation at College" exemplifies this trend. It is a mobile application designed to streamline and modernize the process of applying for railway concessions. This report serves as comprehensive documentation of the conception, development, and evaluation of the railway concession system at college.

In the context of the Kannada language, text summarization poses unique challenges due to its rich morphology and syntactic structure. The initiative "Saaramsha: Kannada Text Summarization Using NLP and Machine Learning Methods" aims to address these challenges by developing a sophisticated system capable of processing and summarizing Kannada text. This project involves collecting a diverse set of Kannada documents, preparing them for analysis, and identifying the critical elements that contribute to effective summarization. By incorporating Kannada-specific linguistic features and using advanced NLP techniques, the system strives to generate summaries that accurately reflect the original content, thereby enhancing the accessibility and comprehensibility of Kannada literature and information.

The research project developed a comprehensive tool for summarizing Kannada text using NLP techniques. The project involved several key steps: scraping Kannada text from web sources, processing and cleaning the text data, and implementing a summarization algorithm. A frequency-based approach was utilized to score sentences, filtering out stop words and assigning scores based on word frequencies. The summarization model was designed to identify and retain the most significant sentences, producing a concise summary of the input text. The tool also includes a user-friendly interface for uploading documents, generating summaries, and displaying results. The performance of the summarization model was evaluated using metrics such as recall, F1 score, and accuracy, providing insights into its effectiveness and areas for further improvement. This project not only advances Kannada text processing but also contributes to the broader field of language technology.

2. METHODOLOGY
In this project, the goal is to condense lengthy Kannada text documents into concise summaries while enriching the dataset through web scraping. The process is as follows:

2.1 Data Collection and Web Scraping
The project begins by employing web scraping techniques to gather Kannada text from various online sources. This broadens the dataset's scope and enhances its diversity. Using advanced web scraping tools, text is extracted from websites, news portals, and other online repositories. This step is vital for building a comprehensive dataset.

2.2 Text Processing
Once we have the text, we subject it to standard text processing procedures to prepare it for summarization. This involves tokenization, where sentences are broken down into individual words or tokens using Natural Language Processing (NLP) techniques. Tokenization simplifies the text into manageable

units for further processing. After tokenization, the text is cleaned by removing punctuation marks like periods and commas, as well as eliminating stop words—common words like conjunctions and adverbs that do not contribute significantly to the text's meaning.

2.3 Frequency Analysis and Keyword Identification

After cleaning the text, we calculate the frequency of each word. This entails counting the occurrence of each word and filtering out low-frequency ones, focusing on those that contribute meaningfully to the document. By identifying high-frequency keywords, we can pinpoint the core themes and crucial information within the text. This frequency analysis is essential for identifying the most relevant parts of the text to include in the summary.

2.4 Ranking and Sentence Selection

The final step involves ranking and selecting sentences for the summary. We utilize the Google Suggestion Score (GSS) coefficient, derived from a pre-trained model, to rank the processed text based on contextual relevance. The GSS coefficient helps evaluate the importance of keywords within the document's context. Sentences containing high-frequency keywords are then selected for inclusion in the summary. This ensures that the essence of the text is preserved while minimizing redundancy and verbosity. By focusing on sentences with high contextual relevance, we create summaries that effectively convey the key information from the original documents. By following this methodology, we aim to produce high-quality summaries that capture the essential content of Kannada text documents. This approach leverages advanced NLP techniques and ensures a diverse and rich dataset through web scraping, enhancing the summarization process's effectiveness and reliability.

This figure 1 flowchart illustrates the process of extracting and summarizing Kannada text, starting from user inputs via URL or text file, preprocessing the text by discarding stop words and computing word occurrences, and finally generating a summarized output using a Generalized Similarity Score (GSS).This meticulous approach allows us to create summaries that effectively convey the crucial information from Kannada text documents. By focusing on sentences with high contextual relevance, we ensure that the summaries are concise yet comprehensive. Our methodology leverages advanced NLP techniques and incorporates web scraping to gather a diverse and rich dataset, enhancing the summarization process's accuracy and reliability.

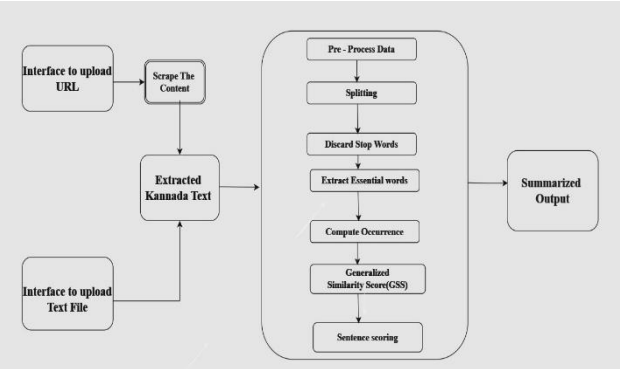

Figure 1: System Architecture

3. RESULTS

The planned project is executed using Python 3.6.4 and incorporates libraries like nltk, pandas, and other necessary modules. The summarization process, which relies on word frequency analysis, is carried out on a dataset containing news articles spanning various subjects. Each input document undergoes experimental evaluation, resulting in a condensed version that encompasses the key details of the entire article. This iterative approach guarantees that the summarized output effectively encapsulates the main ideas and essential information from the original text, thereby improving the accessibility and usefulness of Kannada news content for a wide audience.

Figure 2: The display page for the text summarization

This figure 2 image shows a user interface for a Kannada text summarization tool, featuring fields to input a URL or upload a text file, display the extracted text, and show the summarized output, with buttons for various actions like uploading, generating summary, and clearing text.

The below figure 3 shows the user interface of a web scraping tool for Kannada text, where users can enter a URL in the provided text field and click a button to scrape and summarize the content from the URL.

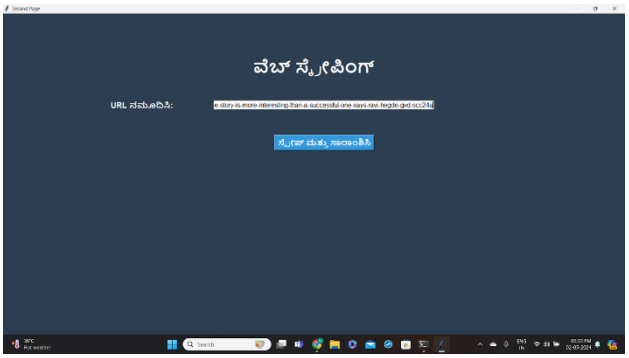

Figure 3: Scraping Text from URL for summarization

This below figure 4 image shows the user interface of a Kannada text conversion tool, featuring fields to upload a text file, display the original text, and show the converted text output, with buttons for various actions like uploading, converting, and clearing text.

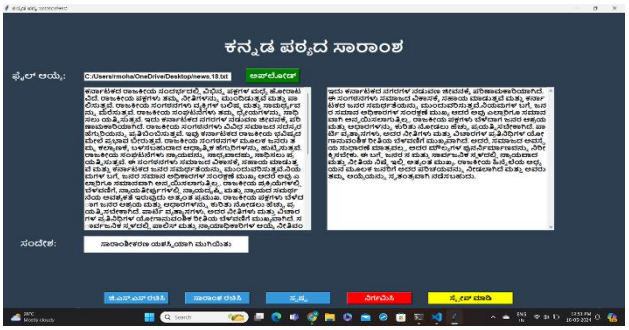

Figure 4: Normal text and summarized text

4. CONCLUSION

In conclusion, our approach will provide a robust solution for summarizing Kannada news articles efficiently, condensing extensive content into concise passages while retaining essential information. By using Natural Language Processing techniques and word frequency analysis, we generate accurate summaries, facilitating information retrieval and comprehension. The integration of web scraping techniques enriches the dataset, contributing to more comprehensive summaries. This methodology offers journalists and news reporters a streamlined approach to news processing and dissemination, enabling faster research and story development, and enhancing audience engagement through timely updates.

5. FUTURE ENHANCEMENTS

Future work aims to enhance the Kannada text summarization application by incorporating abstractive summarization techniques. Unlike extractive methods, abstractive summarization can generate new sentences that more accurately capture the essence of the original text, resulting in more coherent and human-like summaries. This can be achieved by leveraging advanced neural network models such as transformers and sequence-to-sequence architectures, which have demonstrated significant improvements in natural language generation tasks.

Additionally, plans include deepening web scraping techniques to handle more complex and dynamic web page structures. By employing more sophisticated scraping frameworks and machine learning algorithms, the extraction of relevant content from diverse web sources can be improved, ensuring that the scraped text is both comprehensive and high-quality. These enhancements will collectively make the application more robust, versatile, and capable of generating superior summaries from a wide range of inputs.

6. REFERENCES

[1] J.N.Madhuri, Ganesh Kumar.R, “Extractive Text Summarization Using Sentence Ranking,” International Conference on Data Science and Communication (IconDSC), 2019.

[2] Mahsa Afsharizadeh, Hossein Ebrahimpour-Komleh, Ayoub Bagheri, “Query-oriented Text Summarization using Sentence,” 4th International Conference on Web Research (ICWR), 2018.

[3] Ping Chen, Fei Wu, Tong Wang, Wei Ding “A Semantic QA-Based Approach for Text Summarization Evaluation,” The Thirty-Second AAAI Conference on Artificial Intelligence (AAAI-18), 2019.

[4] Md Ashraful Islam Talukder, Sheikh Abujar, Abu Kaisar Mohammad Masum, Fahad Faisal, Syed Akhter Hossain, “Bengali abstractive text summarization using sequence to sequence RNNs,” 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT), 2019.

[5] Scott A. Crossley, “Automated Summarization Evaluation (ASE) Using Natural Language Processing Tools,” International Conference on Ar‹t#if›icial Intelligence in Education, 2019.

[6] Rupali Hande, “Two-Level Text Summarization with Natural Language Processing,” International Conference on Computer Networks and Inventive Communication Technologies, 2019.

[7] Avinash Payak, “Automatic Text Summarization and Keyword Extraction using Natural Language Processing,” International Conference on Electronics and Sustainable Communication Systems, 2020.

[8] Vishal Soni, “Text Summarization: An Extractive Approach,” Advances in Intelligent Systems and Computing, 2020.

[9] K. Janaki Raman, “Automatic Text Summarization of Article (NEWS) Using Lexical Chains and WordNet,” Artificial Intelligence Techniques for Advanced Computing Applications, 2020.

[10] Sagarika Pattnaik, “Automatic Text Summarization for Odia Language: A Novel Approach,” Intelligent and Cloud Computing, Smart Innovation, Systems and Technologies, 2021.

[11] M. Lu and F. Li, "Survey on lie group machine learning," in Big Data Mining and Analytics, vol. 3, no. 4, pp. 235-258, Dec. 2020, doi: 10.26599/BDMA.2020.9020011.

[12] Jayashree. R, Srikanta Murthy.K and Sunny.K, “Keyword Extraction based Summarization of Categorized Kannada Text Documents”, International journal on soft computing (IJSC), Vol.2, N0.4, November 2011.

[13] Letian Wang, Fang Li, SJTULT LAB: “ Chunk Based Method for Keyphrase Extraction”, Proceedings of the 5th International Workshop on Semantic Evaluation, ACL 2010,pp 158– 161,Uppsala, Sweden,15-16 July 2010.

[14] You Ouyang Wenjie Li Renxian Zhang,'273. Task 5. “Key phrase Extraction Based on Core Word Identification and Word Expansion”, Proceedings of the 5th International Workshop on Semantic Evaluation, ACL 2010, pp 142–145, Uppsala, Sweden, 15-16 July 2010.

[15] Rupali Hande, “Two-Level Text Summarization with Natural Language Processing,” International Conference on Computer Networks and Inventive Communication Technologies, 2019.

[16] Y. Ko, et al., “Automatic text categorization using the importance of sentences,” in Proceedings of the 19th International Conference on Computational Linguistics, Vol. 1, 2022, pp. 1-7.

[17] A. Kolcz, et al., “Summarization as feature selection for text categorization,” in Proceedings of the 10th International Conference on Information and Knowledge Management, 2021, pp. 365-370.

[18] D. Shen, et al., “Web-page classification through summarization,” in Proceedings of the 27th Annual International Conference on Research and Development in Information Retrieval, 2020, pp. 242-249.

[19] A. McCallum and K. Nigam, “A comparison of event models for Naïve Bayes text classification,” in Proceedings of AAAI Workshop on Learning for Text Categorization, 2020, pp. 41-48.

[20] R. R. Yager, “An extension of the naïve Bayesian classifier,” Information Sciences, Vol. 176, 2006, pp. 577-588.

[21] T. Joachims, “A probabilistic analysis of the Rocchio algorithm with TFIDF for text categorization,” in Proceedings of the 14th International Conference on Machine Learning, 1997, pp. 143-151.

[22] I. Rahal and W. Perrizo, “An optimized approach for KNN text categorization using P-trees,” in Proceedings of ACM

Symposium on Applied Computing, 2004, pp. 613-617.

[23] E. Gabrilovich and S. Markovitch "Text categorization with many redundant features: using aggressive feature selection to make SVMs competitive with C4.5," in Proceedings of the 21st International Conference on Machine Learning, 2004, pp. 321-328.