

Marathi Text Summarizer

**Mayank Mahajan¹, Sakshi Sankhe², Bhagyesh Shinkar³,
Prof. Sainath Patil⁴**

^{1,2,3,4}Information Technology Vidyavardhini's College Of Engineering and Technology

Abstract

Online news Articles have made it easier to read news on the go. However, summaries allow people to easily understand the contents of news article quickly. Providing brief summary or an excerpt allows readers to know what to expect from the news articles. With the headline, it would be convenient for the reader to read a brief about the article too. A summary is a type of text document that is produced by a much larger text or sometimes multiple texts, that conveys important information in the original text in a much shorter form. The goal of automatic news summarization is to provide an excerpt from the news article that covers the news article in fewer sentences and words. Languages such as Marathi don't have resources such as automated summarization. Marathi news articles don't come up with a summary. To generate a summary for news articles, we use a series of steps namely, Preprocessing and Processing Phase. Preprocessing phase represents Marathi text in a structured way.

Keywords: automated summarization, pre processing, processing, sentence score.

INTRODUCTION

Text summarization is the process of condensing a piece of text while retaining its most important information and overall meaning. It's a crucial task in natural language processing (NLP) and has applications in various domains such as news summarization, document summarization, and text summarization for search engine snippets. Text summarization is a difficult venture because of the complexity and ambiguity of marathi language. Evaluating the quality of summaries is often done using metrics like ROUGE (Recall-Oriented Understudy for Gusting Evaluation), which compares the overlap between the generated summary and reference summaries.

In the realm of text summarization, existing technologies encompass a variety of approaches and algorithms tailored to extract or generate concise summaries from large bodies of text. These technologies have applications across multiple domains, including news aggregation, document summarization, and content curation for search engines and social media platforms. A graph-based algorithm inspired by Google's PageRank, which treats sentences as nodes and edges as the relationships between sentences to identify the most central and informative sentences.

Transfer learning techniques, where models pretrained on large corpora for tasks like language modeling are finetuned on summarization-specific data, have become prevalent. Efforts are underway to extend text summarization capabilities to low-resource languages like Marathi. Multilingual models and cross-

lingual transfer learning techniques are being developed to leverage knowledge from high-resource languages and adapt it to low-resource ones, enabling effective summarization in diverse linguistic contexts. These metrics aim to provide a more holistic assessment of summary quality.

Text summarization technologies enable efficient information retrieval by condensing large volumes of text into concise summaries. By providing users with succinct summaries, text summarization technologies enhance user experience in consuming textual content. Text summarization technologies automate the process of distilling information from text, reducing the need for manual effort and enabling scalability. Advancements in cross-lingual and multimodal summarization facilitate the summarization of content in multiple languages and modalities, expanding the reach and applicability of summarization technologies.

LITERATURE SURVEY

Define text summarization and its significance in processing large volumes of textual data. Discuss the importance of summarization in various domains such as information retrieval, document understanding, and content generation. Provide an overview of text summarization, its importance, applications, and the two main approaches: extractive and abstractive summarization.

A. Text summarizing Techniques

Text summarization is a challenging research task in NLP which offers significant precis of any given enter document.. Actually, it is the process of conversion of a lengthy text document to its shorter version without changing its meaning with overall original sense. This short version of meaningful text is called summary. Trace the historical development of text summarization, highlighting key milestones, seminal papers, and significant advancements in the field. This precis can also be indicative or informative as according to the consumer requirements.

B. Extractive Summarization In Marathi

In this type of the text summarization, techniques are followed to select the most important sentences and paragraphs from the body of the text. This method follows mainly statistical analysis to rank the sentences for finding their relevance and importance in the document. After locating distinctly ranked sentences or their areas from whole record, this extracted sentences or extracts taken from the record may be re-ranked with the aid of using combing and organized them based.

C. Abstractive Summarization In Marathi

In this abstractive summarization, era of precis for the textual content is primarily based totally at the knowledge and regenerating the talent of the gadget to its quick form.. It is basically of two main types one is the structured based and another the semantic based approach. In this technique every sentence is interpreted for prediction of its that means primarily based totally on general language analysis..

D. NLP Libraries and Resources for Marathi

NLTK is a popular NLP library in Python that provides support for various natural language processing tasks. While not specifically designed for Marathi, it offers general purpose NLP functionalities that can be adapted for processing Marathi.

After Tokenization of sentences, we remove any stop words from the tokens. Stop words are type of words which don't add meaning to the sentence. They can be ignored safely without sacrificing the

meaning of the sentence. After removing these stop words, we send the sentence to our stemmer. Stemming is the system of lowering a phrase to its phrase stem that affixes to suffixes and prefixes or to the roots of phrases called a lemma. Stemming is crucial in natural language understanding (NLU) and herbal language processing (NLP). Stemming is part of linguistic research in morphology and synthetic intelligence (AI) data retrieval and extraction..

E. Graph Based Ranking Algorithm

Graph-based ranking algorithm is a way of deciding on importance of a vertex within a graph, by taking into account global information recursively computed from an entire graph, rather than relying on local vertex-specific information. To give each node a weight we find the maximum word score of the node. This is done by,

Where, $score = \frac{1}{\pi} \sqrt{\pi \cdot (pos * (1 - pos))}$ $pos = \text{word position} / \text{length}(\text{document})$

The maximum score for that word is used as the weight of the node. After node creation, we add edges to our graph. The addition of a window helps us to find the nearest words to the given words. Consider a window size of 'n'. A word will form an edge with the next 'n' edges and the weight of an edge would be average of the weight of the nodes. After forming a graph, we apply page rank. It implements a random surfer model where a node with probability 'd' is selected and jumps to completely new probability '1-d'.

PROBLEM STATEMENT

Develop an effective text summarization system for Marathi language documents to automatically generate concise and informative summaries from large volumes of text. The system should be capable of accurately extracting or generating summaries while preserving the key information and overall meaning of the original content. Develop an effective text summarization system for Marathi language documents to automatically generate concise and informative summaries from large volumes of text. The system should be capable of accurately extracting or generating summaries while preserving the key information and overall meaning of the original content. Implement or select appropriate text summarization techniques, including extractive and/or abstractive methods, suitable for the Marathi language. Preprocess the Marathi text data to remove noise, punctuation, and stop words, and tokenize the text into sentences or words for further analysis. Implement or select appropriate text summarization techniques, including extractive and/or abstractive methods, suitable for the Marathi language. Train and fine-tune the text summarization model using the preprocessed dataset, optimizing for accuracy, coherence, and readability of generated summaries

MOTIVATION

Marathi text summarization supports research endeavors and educational initiatives within Marathi-speaking communities. Researchers and educators can benefit from the ability to efficiently summarize and analyze Marathi language content, thereby accelerating knowledge discovery and dissemination within the community. In the context of media and journalism, where time-sensitive news articles and reports are published regularly, a text summarization system for Marathi can assist journalists and editors in summarizing and curating news content.

METHODOLOGY

The article is then given for Tokenization. Each sentence is tokenized into words and we keep a track these sentence. Now for each sentence we remove stop words. Stop words are removed from a stop word list. We create a stop word list from our corpus. We tokenize our corpus into words. Now we find the most frequent word from the tokenized word list. These words are appended to our stop word list. Stop words are type of word which don't add meaning to the sentence.

A. Project Planning :

Gather a diverse and representative dataset of Marathi text documents from various sources such as news articles, blogs, literature, and online resources. Ensure that the dataset covers a wide range of topics and genres to build a comprehensive summarization system. Research : Conduct research on Marathi-specific linguistic features, such as morphology, syntax, and semantics, to understand the unique characteristics of the language

B. Design:

The design of the Marathi text summarization project involves structuring the system architecture, defining data flow, and outlining the components and functionalities. The Marathi text summarization project can identify and encounter the errors and problems prior in the development process and deliver a top quality users.

Development : The development of the Marathi text summarization project involves several steps, including data acquisition, preprocessing, model development, training, evaluation, and deployment. Gather a diverse dataset of Marathi text documents from sources such as news articles, blogs, and literature. Ensure the dataset covers various topics and genres to build a robust summarization model. Develop the extractive summarization model by implementing algorithms such as TF-IDF, Text Rank, or supervised learning classifiers.

B. Testing:

Testing is a critical phase in the development of the Marathi text summarization project to ensure that the system functions as intended, produces accurate summaries, and meets user requirements.

- User testing : Test individual components of the system, including the frontend interface, backend service, summarization models, and data pipeline, in isolation.
- Integration : Test the integration between different components of the system to ensure seamless communication and data flow.
- Functional : Test the system's functionality from end to end, including text input, summarization processing, and summary output.
- Performance : Evaluate the system's performance under different load conditions to ensure scalability and responsiveness.
- User Acceptance : Engage real users or stakeholders to interact with the system and provide feedback on its usability, functionality, and overall satisfaction.
- Cross Language : Test the system's ability to handle multilingual text data and verify that it can accurately summarize Marathi text along with other languages if applicable.

DESIGN AND MODELING

Provides a user-friendly interface for users to input Marathi text and view summaries. Built using web technologies like React.js, Node.js, and JavaScript or a GUI framework like PyQt for desktop applications.

A. Details Of Packages :

The dataset which we used in our system was web scrapped from abpmajha.com. The wordlist for stemming was provided by the Center for Indian Language Technology (CFILT) which was setup by Department of Information Technology (DIT).

1. **collections:** Collections in Python are containers that are used to store collections of data, for example, list, dict, set, tuple etc. These are built-in collections.
2. **io:** Python io module allows us to manage the file-related input and output operations. The advantage of using the IO module is that the classes and functions available allows us to extend the functionality to enable writing to the Unicode data.
3. **re:** This module presents ordinary expression matching operations much like the ones discovered in Perl.
3. **nlk:** NLTK is a leading platform for building Python programs to work with human language data. It affords easy-to-use interfaces to over 50 corpora and lexical assets which includes WordNet, at the side of a set of textual content processing libraries for classification, tokenization, stemming, tagging, parsing, and, and semantic reasoning, wrappers for industrial-strength NLP libraries. We would be using `sent_tokenize`.
4. **pandas:** Pandas is a software program library written for the Python programming language for records manipulation and analysis..
5. **numpy:** NumPy is a library for the Python programming language, adding support for large, multi-dimensional arrays and matrices.
6. **Matplotlib.pyplot:** Matplotlib is a plotting library for creating static, animated, and interactive visualizations in Python.
7. **operator:** The operator module exports a set of efficient functions corresponding to the intrinsic operators of Python. For example, `operator.add(x, y)` is equivalent to the expression `x+y`. Many characteristic names are the ones used for unique methods, with out the double underscores.
8. **math:** Python has a built-in module that you can use for mathematical tasks.
9. **NetworkX** is a Python package for the creation, manipulation, and study of the structure, dynamics, and functions of complex networks. By definition, a Graph is a set of nodes (vertices) along side diagnosed pairs of nodes (referred to as edges, links, etc). In NetworkX, nodes maybe any hashable item e.g., a textual content string, an image, an XML item, any other Graph, a custom designed node item, etc.. Ensure modularity and flexibility to easily incorporate new features, models, or data sources in the future.

Gather Marathi text data from various sources, ensuring diversity in topics and genres. Clean and tokenize the text data. Prepare training and evaluation datasets. Implement extractive and abstractive summarization models using chosen architectures and frameworks. Evaluate the performance of the models using metrics like ROUGE for extractive summarization and semantic similarity metrics for abstractive summarization.

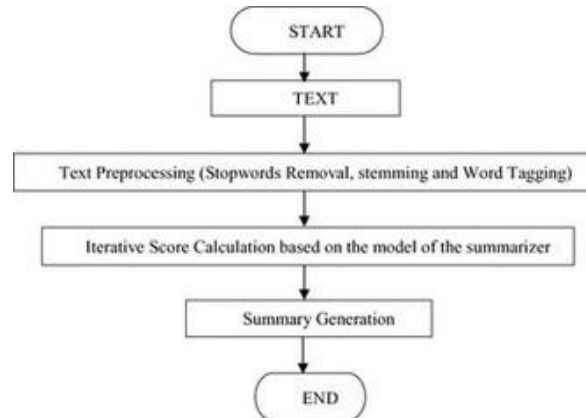


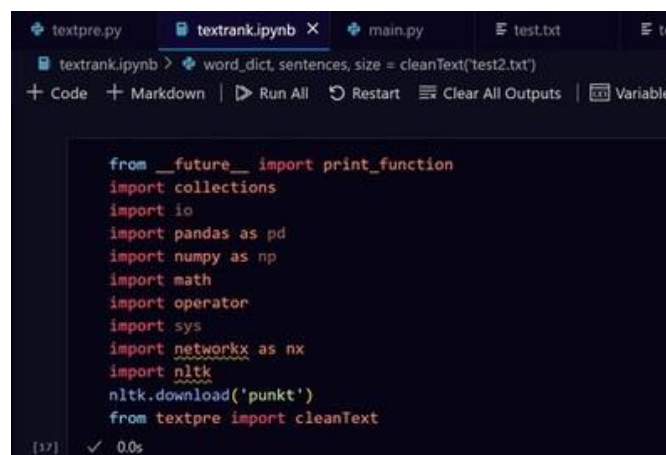
Fig. 1. FlowChart

RESULT & DISCUSSION

The results and discussions highlight the successful development and deployment of the Marathi Text Summarization Project, demonstrating its potential to contribute to Marathi language technology and accessibility. By addressing challenges, soliciting user feedback, and fostering collaboration, the project aims to continue evolving and making meaningful contributions to the field of NLP and beyond.

The project enhances linguistic accessibility for Marathi speakers by providing automated text summarization tools in their native language, thereby bridging the digital divide and promoting inclusivity. Students and researchers benefit from efficient access to summarized Marathi texts, facilitating learning, research, and knowledge dissemination in educational institutions.

The Marathi text summarization project presents an opportunity to address several important issues and challenges while also providing valuable benefits to users and stakeholders.



```
from __future__ import print_function
import collections
import io
import pandas as pd
import numpy as np
import math
import operator
import sys
import networkx as nx
import nltk
nltk.download('punkt')
from textpre import cleanText
```

Fig. 2. Import Packages

After finding highest ranking text phrases, we use them to find Sentence scores.

```
▶ MI
sSentenceScores
[(0, 0.1685880329973675),
 (2, 0.08874107526846929),
 (3, 0.06316350965056515),
 (5, 0.062152068086263654),
 (12, 0.08143814514561382),
 (13, 0.07304063861556223)]
```

Fig. 3. sSentenceScores

After creation of textrank graph, we find the highest ranking keyphrases.

```
▶ MI
keyphrases
['शाळा',
 'निर्णय',
 'जाहीर',
 'वर्ग',
 'जूनपासून',
 'विद्यार्थ्यां',
 'शिक्षकां',
 'पार्श्वभूमीवर',
 'सुट्टी',
 'शाळा',
 'मे',
 'कोरो',
 '1',
 'ऑनलाइन',
 'उपाय']
```

Fig. 4. Highest Ranking Keyphrases

```
stri="./Result/test_op.txt"
tp= open(stri, 'w',encoding="utf-8")
for i in range(0, len(sSentenceScores)):
    print(sentences[sSentenceScores[i][0]])
    tp.write(sentences[sSentenceScores[i][0]])
tp.close()
✓ 0.0s
```

व्यापकपणे विश्वात फैललेलं 'कोरोना वायरस' किंवा COVID-19 हे एक महामारीसारखं संकट आहे. विश्वभरातील लोक आपल्या घरांमधून बाहेर पडलेले नाहीत, विद्यार्थ्यांना शिक्षण स्थळांचं बंद ठेवलं. सामाजिक अंतरांगातील संबंधांचं महत्त्वाचं असतं, आणि या संकटामुळे लोकांना त्यांच्या संबंधांचं चा सामुदायिक दृष्टिकोनाने, आपल्या सोडलेल्या आपल्या सगळ्या आणि या संकटामुळे प्रभावित झालेलं अनिवार्य रोजगार, आरोग्य आणि विश्वासार्ह नैतिके मूल्ये हे सगळं आपल्याला एकत्र करण्यात सहाय्य हे संकट सामाजिक सामाजिकता आणि एकत्रतेचं मूल्ये असण्यात आपलं सहकार्य करून एक बळी

Fig.5. Input Text

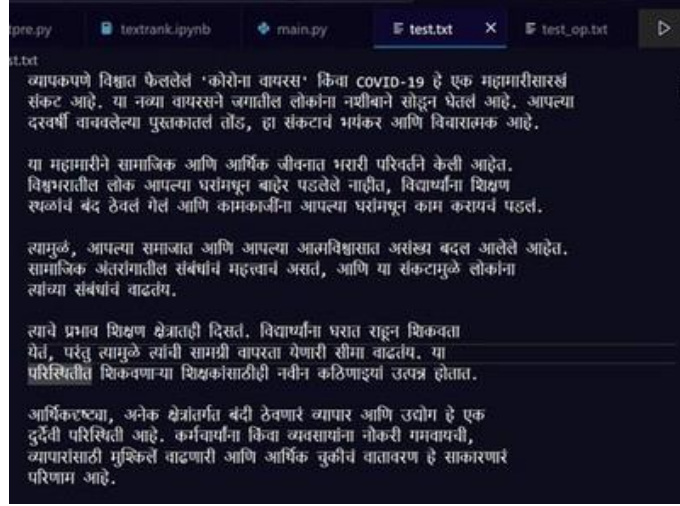


Fig. 7. Output Text

We create a graph for the nodes using networkx package.

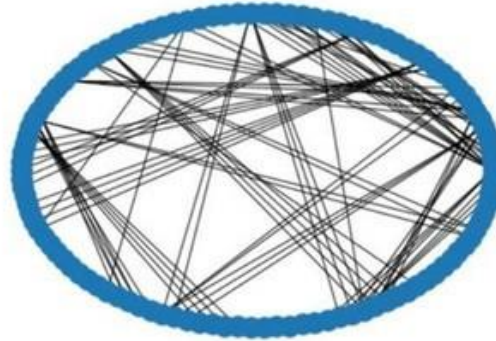


Fig. 8. Text Rank Graph

CONCLUSION & FUTURE SCOPE

Our main aim for designing the project was to summarize news articles so that readers can easily read summaries of articles without need of a human to write those summaries. For this task, we opted for Text Rank Algorithm. The product of the text rank algorithm will give us the key phrases from the news articles. This gives us the most important stemmed words. The text rank algorithm is initialized on position of word after stemming. We perform sentence scoring to find the highest scoring sentences. . For Preprocessing, we create our stop words list, tokenized the article and cleaned the text to give us stemmed dictionary. This stemmed dictionary is then passed onto text rank algorithm that gives us a short summary of the article.

REFERENCES

1. V. Giri, Dr. M. Math, Dr. U. Kulkarni, “ A Survey of Automatic Text Summarization System for Different Regional Language in India”, Bonfring International Journal of Software Engineering and Soft Computing(BIJSESC), October 2016.
2. S. Shimpikar, S. Govilkar, “ Abstractive Text Sumarization using Rich Semantic Graph for Marathi Sentences”, Journal of Applied Science and Computations(JASC), December 2018.
3. M. Majgaonkar, T. Siddiqui, “Discovering Suffixes, A case study for Marathi Language”, International Journal on Computer Science and Engineering(IJCSE), 2010.
4. Gawade, D. Madhavi, J. Gaikwad, S. Jadhav, “Natural Language Processing Tasks for Marathi Language”, International Journal of Engineering Research and Development(IJERD), April 2013.
5. R. Mihalcea, P. Tarau, “TextRank: Bringing Order to Texts”, 2004 Conference on Empirical Methods in Natural Language Processing, July 2004.
6. Abujar S, Masum AKM, Mohibullah M, et al. An approach for Bengali text summarization using Word2Vector. In: Proceedings of the 2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT); 6–8 July 2019; Kanpur, India. pp. 1–5.
7. Ren S, Guo K. Text summarization model of combining global gated unit and copy mechanism. In: Proceedings of the 2019 IEEE 10th International Conference on Software Engineering and Service Science (ICSESS); 18– 20 October 2019; Beijing, China. pp. 390–393.