International Journal of Engineering and Advanced Technology (IJEAT)
ISSN: 2249-8958 (Online), Volume-12 Issue-5, June 2023

Marathi Text Summarization using Extractive Technique

Kirti Pankaj Kakde, H. M. Padalikar

***Abstract**: Multilingualism has played a key role in India, where people speak and understand more than one language. Marathi, as one of the official languages in Maharashtra state, is often used in sources such as newspapers or blogs. However, manually summarizing bulky Marathi paragraphs or texts for easy comprehension can be challenging. To address this, text summarization becomes essential to make large documents easily readable and understandable. This research article focuses on single document text summarization using the Natural Language Processing (NLP) approach, a subfield of Artificial Intelligence. Automatic text summarization is employed to extract relevant information in a concise manner. Information Extraction is particularly useful when summarizing documents consisting of multiple sentences into three or four sentences. While extensive research has been conducted on English Text Summarization, the field of Marathi document summarization remains largely unexplored. This research paper explores extractive text summarization techniques specifically for Marathi documents, utilizing the LexRank algorithm along with Genism, a graph-based technique, to generate informative summaries within word limit constraints. The experiment was conducted on the IndicNLP Marathi news article dataset, resulting in 78% precision, 72% recall, and 75% F-measure using the frequency-based method, and 78% precision, 78% recall, and 78% F-measure using the Lex Rank algorithm.*

***Keywords**: Artificial Intelligence, Automatic text summarization, Extractive text summarization, Natural Language Processing, Indic NLP.*

I. INTRODUCTION

Text summarization is necessary when a large amount of text data or a paragraph needs to be condensed into a shorter form. Natural Language Processing (NLP) is a subfield of artificial intelligence that deals with this task. It involves extracting valuable information and summarizing a significant volume of unstructured data into a few sentences. The purpose of automatic text summarization is to efficiently manage extensive information. It is crucial to preserve the original meaning of the text during the text summarization process. The reader can save time by reading a concise summary instead of the entire paragraph.

Manuscript received on 29 May 2023 | Revised Manuscript received on 06 June 2023 | Manuscript Accepted on 15 June 2023 | Manuscript published on 30 June 2023.
*Correspondence Author(s)
Mrs. Kirti Pankaj Kakde*, Research Scholar, Department of Computer Application, IMED Bharati Vidyapeeth Deemed to be University, Pune (M.H), India. E-mail: kirtikakde2007@gmail.com, ORCID ID: 0000-0002-1428-6850
Dr. H. M. Padalikar, Department of Computer Application, IMED Bharati Vidyapeeth Deemed to be University, Pune (M.H), India. E-mail: hmpadalikar1965@gmail.com, ORCID ID: 0000-0002-3143-4069

This research article focuses on the exploration of extractive text summarization, which is a technique within Natural Language Processing (NLP) [10]. Extractive text summarization is accomplished using the Genism Python package. In this technique, the model extracts key sentences from the original text. Contrarily, abstractive text summarization models may introduce new words or rephrase sentences. It's important to note that the words appearing in the summary may not necessarily be present in the original text. [1]

II. PREVIOUS WORK

Several text summarization techniques have been explored by researchers. Most text summarization methods work for English language datasets. In this section, the work of researchers specifically focusing on the Marathi language has been explored.

Jagadish S Kalimani et al. [1] suggest text summarization by extractive method. The method consists of data extraction from the source document which is then further processed to extract key concepts from the original text. The author presents techniques such as pre-processing, summarizing, and post-processing.

Yogeshwari V. Rathod et al. [2] presented a technique for the summarization of Marathi news articles using extractive text summarization. The method involves selecting important sentences and concatenating them into a shorter form. The authors implemented a PageRank algorithm specifically for Marathi text summarization.

Desai Nikita et al. [3] presented an approach to design an automatic text summarizer for Hindi news articles. Supervised machine learning tool SVM ^{rank} is used to extract important sentences and sentence ranking is assigned. In the sentence ranking process, sentences are assigned ranks ranging from 4 to 1, with 4 indicating the most important sentence and 1 indicating a less important sentence. Pre-processing, processing, and extraction steps are conducted on the text. During the pre-processing phase, the features of each sentence are calculated.

Nikhil S. Shirwandkar et al. [4] extractive text summarization approach is designed and implemented for single document summarization. Implements the logic of Boltzmann machine and fuzzy logic to select important sentences.

R. C. Balabantaray et al. [5] calculated score for individual words and assigned sentences score by adding words score. The summarizer extracts top ranked sentences and included in the summary.

*Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)*

Manjula Subramaniam et al. [6] have implemented an abstractive method using a rich semantic graph technique to summarize Hindi text. A set of features are extracted from each sentence which helps in the identification of its importance in the document. The authors have used Hindi WordNet for checking SOV(Subject-object-verb). Similarity among the sentences has been calculated and sentences are merged. Arti Jain et al. [7] proposed Hindi text summarization by using a real-coded genetic algorithm. The highest-scored sentence is extracted into the corpus summary. Sentences are selected based on the text's statistical parameters and linguistic features. Extractive summarization is achieved by pre-processing, feature extraction, and preprocessing phase. Feature extraction is achieved with sentence paragraph position, numerical data, sentence length, keywords within a sentence, sentence similarity, Named Entities, English-Hindi words within a sentence, and Term Frequency (TF)- Inverse Sentence Frequency. Vaishali V. Sarwadnya et al. [8] implemented extractive text summarization on a multi-document dataset in the Marathi language. The text summary generated by their approach was evaluated using the ROGUE metric, a commonly used evaluation metric for summarization tasks. Their work employed the TextRank algorithm, which utilizes the weighted positional distribution of sentence scores to determine the importance of sentences in the summary generation process. Virender Dehru et al. [9] explores different techniques of text summarization and evaluated them on different parameters. The author concludes that statistical-based algorithms generate fast and decent summaries. In the extractive method, word frequency is calculated then the greedy approach is used to determine top k sentences. Other methods are word Probability, and TextRank algorithm are implemented in the research.

III. METHODOLOGY

Text summarization means the large text is broken into limited number of sentences by extracting vital information and preserving the actual meaning of the original text using AI and NLP to a large extent [12]. In Text summarization, the shorter text is created without changing the semantic structure of the text [13]. In the text summarization of Marathi news articles, an extractive technique is implemented using the IndicNLP Marathi news article dataset. The dataset consists of 4779 records and is divided into predefined sets of train, test, and validation datasets.

A. Extractive text summarization

In the extractive text summarization process, both the frequency method and LexRank algorithm are employed in the frequency method, the first step involves the pre-processing stage (stop words removal, special characters removal), followed by the calculation of word frequencies. These frequencies are then stored in a dictionary format where the data is tokenized [15]. Tokenization is dividing a complete sentence in the word level. During the summarization process, sentences containing high-frequency words are selected and retained in the final summary. The evaluation of the final summary of each document is performed using ROUGE metrics (Recall-Oriented Understudy for Gisting Evaluation). The complete explanation is summarized to Figure 1.

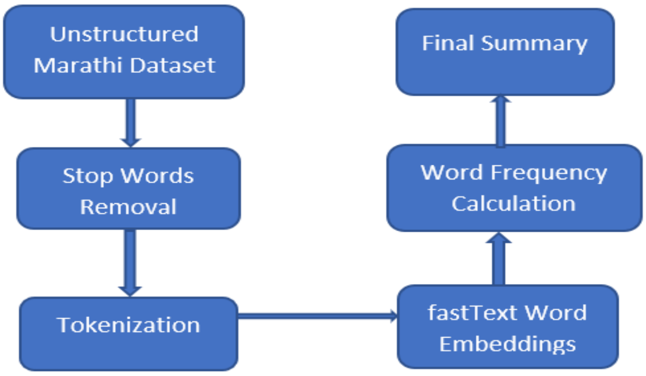

Fig. 1. Text Summarization using Frequency-Based Method

B. Lex Rank Algorithm

In the implementation of the LexRank algorithm for automated text summarization, cosine similarity, and vector-based algorithms are utilized. A Bag of Words model is created to determine the minimum cosine distance. The cosine similarity is computed between two non-zero vectors, which helps identify similar words that are then stored [11]. The LexRank algorithm capitalizes on the similarity between words and phrases to calculate the centrality of sentences. This approach helps identify the most important sentences for inclusion in the summary based on their centrality within the text as shown in Figure 2.

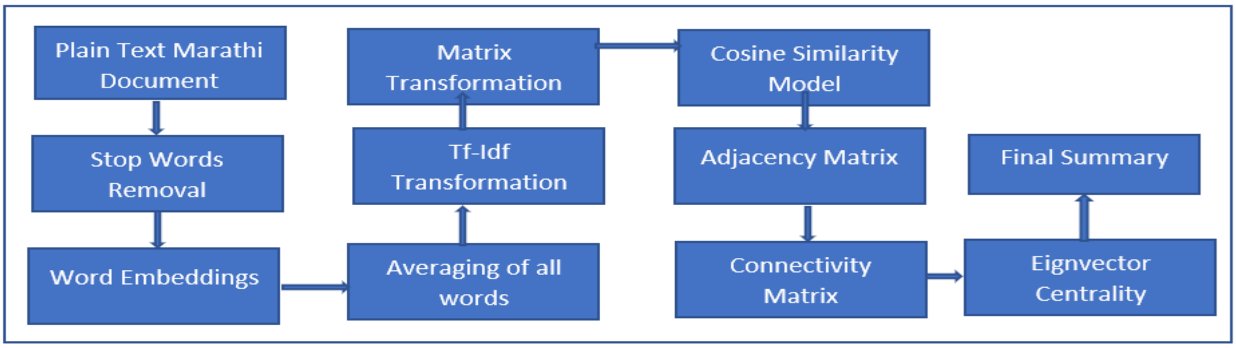

Fig. 2. Text Summarization using Lex Rank Algorithm

Retrieval Number:100.1/ijeat.E42000612523
DOI: 10.35940/ijeat.E4200.0612523
Journal Website: www.ijeat.org

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

All the above phases of Text Summarization using LexRank algorithm as shown in Figure 2 are explained below:

- **Plain text Marathi document**

  In phase One, as shown in Figure 2, Plain Text is given as input to the algorithm. The plain text contains unstructured data which has a 'n' number of sentences. The final output is extracted into a few sentences from the original summary.

- **Stop word removal.**

  In phase Two, as shown in Figure 2, in Marathi documents, stop words such as अधिक, अनेक, अशी, असा, असून, असे, which occur frequently in the document, are removed [2]. By removing these stop words, the sentence score can be increased. To enhance the sentence scores, important and relevant words from the actual summary are used. This approach ensures that the summary includes key information by giving higher weightage to significant words.

- **Word Embeddings**

  In phase Three as shown in Figure 2, Word embeddings are used to represent every single word in vector form during the process of sentiment analysis[15]. For English, word embedding models such as word2vec, fastText word embeddings, and Glove word embeddings are readily available. In this research, the Natural Language Toolkit for Indic Languages (iNLTK) word embeddings are utilized, which have been trained on Marathi Wikipedia text.

  Word embeddings for Marathi text=" अंधेरी पूर्व आणि पश्चिमेला जोडणारा गोखले उड्डाणपूल खुला करण्याची अंतिम मुदत पुन्हा हुकणार आहे. यापूर्वी मे महिन्यापर्यंत गोखले पूल खुला केला जाणार होता. मात्र मुंबई महापालिकेकडून नोव्हेंबरच्या मध्यापर्यंत उड्डाणपूल खुला केला जाणार आहे."

```
[array([-0.732861, 0.930898, 0.06159 , -0.305389, ..., 0.668552, -0.473136, -0.444182, -
0.317486], dtype=float32), array([ 0.79433 , 0.432035, 0.261251, 0.089973, ..., -0.381715,
0.070284, -0.177039, 0.256807], dtype=float32), array([-0.460383, -1.068219, 0.244518, -
0.229393, ..., 0.491053, 0.054434, -0.38986 , -0.024719], dtype=float32), array([ 0.939586, -
0.523766, -0.038508, -0.801881, ..., 0.908812, -0.053073, 0.760309, -0.3133 ],
dtype=float32), array([ 0.651397, 0.705357, -0.496279, -0.157455, ..., -0.306383, 0.208805,
0.143976, -0.596679], dtype=float32), array([ 1.3813 , 0.770594, -0.694167, 0.094508, ...,
0.359712, -0.138906, 0.138542, 0.58531 ], dtype=float32), array([ 0.804245, 0.744594,
0.07197 , -0.453278, ..., -0.315779, 0.07168 , 0.179312, 0.036809], dtype=float32), array([
0.66131 , 0.113566, -0.189015, 0.237278, ..., -0.166633, -0.257219, 0.489197, -0.208763],
dtype=float32), array([ 0.159414, 0.275834, -0.246158, -0.350682, ..., -0.409915, -0.29167 ,
-0.04238 , -0.364499], dtype=float32), array([ 0.521436, -0.313861, -0.042937, 0.152386, ...,
-1.315673, -0.156603, -0.266248, 0.336599], dtype=float32), array([-0.169422, -0.039562, -
0.404804, -0.159102, ..., -0.387959, -0.044609, -0.028966, 0.157989], dtype=float32), array([
0.886413, 0.362037, -0.144303, -0.295601, ..., -0.441747, 0.031105, -1.259664, 0.220495],
dtype=float32), array([-0.444713, 1.204554, -0.428673, -0.734121, ..., 0.986748, 0.151035,
-1.096143, -0.123966], dtype=float32), array([-0.375358, -0.455526, 0.359641, -0.349043, ...,
-0.771446, -0.039537, -1.1662 , 0.092839], dtype=float32)]
shape: (400,)
14
```

Fig. 3. Word Embeddings for Marathi Text

- **Averaging of all words:**

  In phase Four, averaging all word embeddings from a sentence is used to compare it with another sentence that is the implementation of intra sentence of a sentence is used.

- **TF-IDF transformation:**

  In phase Five, Term Frequency — Inverse Document Frequency [4] is used for information retrieval or feature extraction. This process shows the importance of the word in a sentence.

- **Matrix transformation**

  In phase Six, important words are fetched in the TF-IDF process and are transformed into numeric data in terms of the matrix for further processing [14].

- **Cosine similarity**

  This is phase Seven as shown in Figure 2, Text vectors are generated using the Bag of Words model for the representation of N-Dimensional vectors. For every single word in a document that is present in a sentence, the value of the respective dimension from the vector representation of the sentence is the number of occurrences of the word in the sentence times Idf of the word. The cosine score is formulated by the formula:

Retrieval Number:100.1/ijeat.E42000612523
DOI: 10.35940/ijeat.E4200.0612523
Journal Website: www.ijeat.org

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

Idf modified cosine (a.b) =

$$\frac{\sum_{w \epsilon a,b} tf_{w,a} tf_{w,b} (idf_w)^2}{\sqrt{\sum_{a_i \epsilon a} \left(tf_{a_i,a_i} df_{a_i}\right)^2} \sqrt{\sum_{b_i \epsilon b} \left(tf_{b_i,b} idf_{b_i}\right)^2}} \quad (1)$$

- **Adjacency matrix**

In step number 8 of Figure 2, The adjacency matrix is utilized to determine similarities between all sentences. The primary reason for implementing the LexRank algorithm is that it considers both the weightage of the sentences and their importance.

- **Connectivity matrix**

In phase 9 of Figure 2, The connectivity matrix is used to keep track of the links from other sentences, mitigating the issue of local traps where some sentences in a document may be more relevant while others are less relevant [10]. To address this, a threshold is applied in the LexRank algorithm to count the number of connections based on the cosine similarity value.

- **Eigenvector centrality**

In phase 10 of Figure 2, Using Eigenvector centrality, introductory sentences are determined. Here, is the step-by-step process:

Step 1: Initially, every element in the matrix is set to 1.

Step 2: In the second step, the rows of the matrix are squared, and the square root of the sum is calculated. This step is repeated iteratively until the normalized values no longer change significantly.

Step 3: The formulation of the sentence importance is expressed using a specific formula.

$$P(u) = \sum \frac{p(v)}{deg(v)} \quad (2)$$

P(u) is the centrality of the node u, deg(v) is the degree of the nodel v.

The final summary after applying Eigen Vector is:

Sentence 1: अंधेरी पूर्व आणि पश्चिमेला जोडणारा गोखले उड्डाणपूल खुला करण्याची अंतिम मुदत पुन्हा हुकणार आहे. यापूर्वी मे महिन्यापर्यंत गोखले पूल खुला केला जाणार होता.

Sentence 2: मात्र मुंबई महापालिकेकडून नोव्हेंबरच्या मध्यापर्यंत उड्डाणपूल खुला केला जाणार आहे.

Sentence 3: मुंबई महापालिका अतिरिक्त आयुक्त (पूल विभाग) वेलरासू आणि अन्य पालिका अधिकाऱ्यांनी पुलाची पाहणी केली.

Sentence 4: गोखले उड्डाणपुलाचे पाडकाम पश्चिम रेल्वे, तर पुलाची पुनर्बांधणी मुंबई महापालिका करत आहे.

Sentence 5: पूल पुनर्बांधणीचे काम मे २०२३पर्यंत पूर्ण करून किमान एक मार्गिका सुरू करण्याचा प्रयत्न होता. या पुलाच्या आरेखनाला पश्चिम रेल्वेने २ फेब्रुवारीला मंजुरी दिली.

Sentence 6: १५ जुलैनंतर रेल्वे हद्दीतील भाग पूर्ण करणे आणि अन्य कामांसाठी तीन महिन्यांची आवश्यकता आहे.

Sentence 7: त्यामुळे हा पूल दिवाळीच्या मध्यापर्यंत पालिकेकडून खुला करण्याचा प्रयत्न असल्याची माहिती पालिकेतील सूत्रांनी दिली.

Sentence 8: मे महिन्यात सुरू होणाऱ्या पुलाला पाच महिने विलंब होणार असल्याचे स्पष्ट झाले आहे. गोखले उड्डाणपूल बंद असल्याने अन्य पर्यायी मार्गांचा वापर करताना वाहतूककोंडीचा सामना करावा लागतो. सन १९७५मध्ये बांधण्यात आलेला गोखले पुलाचा भाग ३ जुलै २०१८ रोजी कोसळून दोन जणांचा मृत्यू झाला होता. रेल्वे हद्दीतील भाग धोकादायक असल्याच्या तक्रारीमुळे त्याचे काम हाती घेण्यात आले आणि ७ नोव्हेंबर २०२२ पासून पूल वाहतुकीसाठी बंद करण्यात आला होता.

Table-I: Lex Rank Eigenvector Centrality performce evaluation

1st Iteration	Sentence 1	Sentence 2	Sentence 3	Sentence 4	Sentence 5	Sentence 6	Sentence 7	Sentence 8	Standardized
Sentence1	0	1	0	0	1	0	1	1	0.532
Sentence 2	1	1	0	0	0	1	0	1	0.511
Sentence 3	0	1	0	1	0	1	1	1	0.224
Sentence 4	0	0	0	1	1	0	0	0	0.698
Sentence 5	1	0	0	0	0	0	1	0	0.524
Sentence 6	0	1	0	0	0	1	0	1	0.697
Sentence 7	1	0	0	1	0	1	1	1	0.698
Sentence 8	0	0	0	0	0	0	1	1	0.345

Based on the provided information shown in Table-I, Sentence 4, Sentence 6, and Sentence 7 have the highest scores and will be extracted in the final summary.

IV. EXECUTION OF LexRank ALGORITHM ON MARATHI LANGUAGE DOCUMENT:

```
In [1]: original_text = 'अंधेरी पूर्व आणि पश्चिमेला जोडणारा गोखले उड्डाणपूल खुला करण्याची अंतिम मुदत पुन्हा हुकणार आहे. यापूर्वी मे महिन्यापर्यंत गोखले पूल खुला केला
        print(original_text)

अंधेरी पूर्व आणि पश्चिमेला जोडणारा गोखले उड्डाणपूल खुला करण्याची अंतिम मुदत पुन्हा हुकणार आहे. यापूर्वी मे महिन्यापर्यंत गोखले पूल खुला केला जाणार होता. मात्र
मुंबई महापालिकेकडून नोव्हेंबरच्या मध्यापर्यंत उड्डाणपूल खुला केला जाणार आहे. मुंबई महापालिका अतिरिक्त आयुक्त (पूल विभाग) वेलरासू आणि अन्य पालिका अ
धिकाऱ्यांनी पुलाची पाहणी केली. गोखले उड्डाणपुलाचे पाडकाम पश्चिम रेल्वे, तर पुलाची पुनर्बांधणी मुंबई महापालिका करत आहे. पूल पुनर्बांधणीचे काम मे २०२३पर्यंत पूर्ण
करून किमान एक मार्गिका सुरू करण्याचा प्रयत्न होता. या पुलाच्या आरेखनाला पश्चिम रेल्वेने २ फेब्रुवारीला मंजुरी दिली. या स्टील उपकरणांसाठी दोन उत्पादक आहेत.
यातील एका उत्पादकाच्या प्लाण्टमध्ये अचानक अनिश्चित काळासाठी संप झाल्याने पुलासाठी लागणारे गर्डर आणि अन्य साहित्य पुरवठ्यावर परिणाम झाला. त्यामुळे एप्रिल अ
खेरीस पुरवठ्याला सुरुवात होणार आहे. याचा परिणाम पुलाच्या पुनर्बांधणीवर झाला आहे. १५ जुलैनंतर रेल्वे हद्दीतील भाग पूर्ण करणे आणि अन्य कामांसाठी तीन महिन्यांची
आवश्यकता आहे. त्यामुळे हा पूल दिवाळीच्या मध्यापर्यंत पालिकेकडून खुला करण्याचा प्रयत्न असल्याची माहिती पालिकेतील सूत्रांनी दिली. मे महिन्यात सुरू होणाऱ्या पुलाला
पाच महिने विलंब होणार असल्याचे स्पष्ट झाले आहे. गोखले उड्डाणपूल बंद असल्याने अन्य पर्यायी मार्गांचा वापर करताना वाहतूककोंडीचा सामना करावा लागतो. सन १९७५
मध्ये बांधण्यात आलेला गोखले पुलाचा भाग ३ जुलै २०१८ रोजी कोसळून दोन जणांचा मृत्यू झाला होता. रेल्वे हद्दीतील भाग धोकादायक असल्याच्या तक्रारीमुळे त्याचे काम
हाती घेण्यात आले आणि ७ नोव्हेंबर २०२२ पासून पूल वाहतुकीसाठी बंद करण्यात आला होता.
```

Retrieval Number:100.1/ijeat.E42000612523
DOI: 10.35940/ijeat.E4200.0612523
Journal Website: www.ijeat.org

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

```
In [25]: # Creating a summary of 3 sentences.
         lex_rank_summarizer = LexRankSummarizer()
         lexrank_summary = lex_rank_summarizer(my_parser.document,sentences_count=3)

         # Printing the summary
         for sentence in lexrank_summary:
           print(sentence)
```

Output after executing LexRank algorithm on Marathi Text is

गोखले उड्डाणपुलाचे पाडकाम पश्चिम रेल्वे, तर पुलाची पुनर्बांधणी मुंबई महापालिका करत आहे.१५ जुलैनंतर रेल्वे हद्दीतील भाग पूर्ण करणे आणि अन्य कामांसाठी तीन महिन्यांची आवश्यकता आहे. त्यामुळे हा पूल दिवाळीच्या मध्यापर्यंत पालिकेकडून खुला करण्याचा प्रयत्न असल्याची माहिती पालिकेतील सूत्रांनी दिली.

The Plain text document contains 20 sentences and after performing LexRank Algorithm the output contains only 3 sentences as shown in the above output.

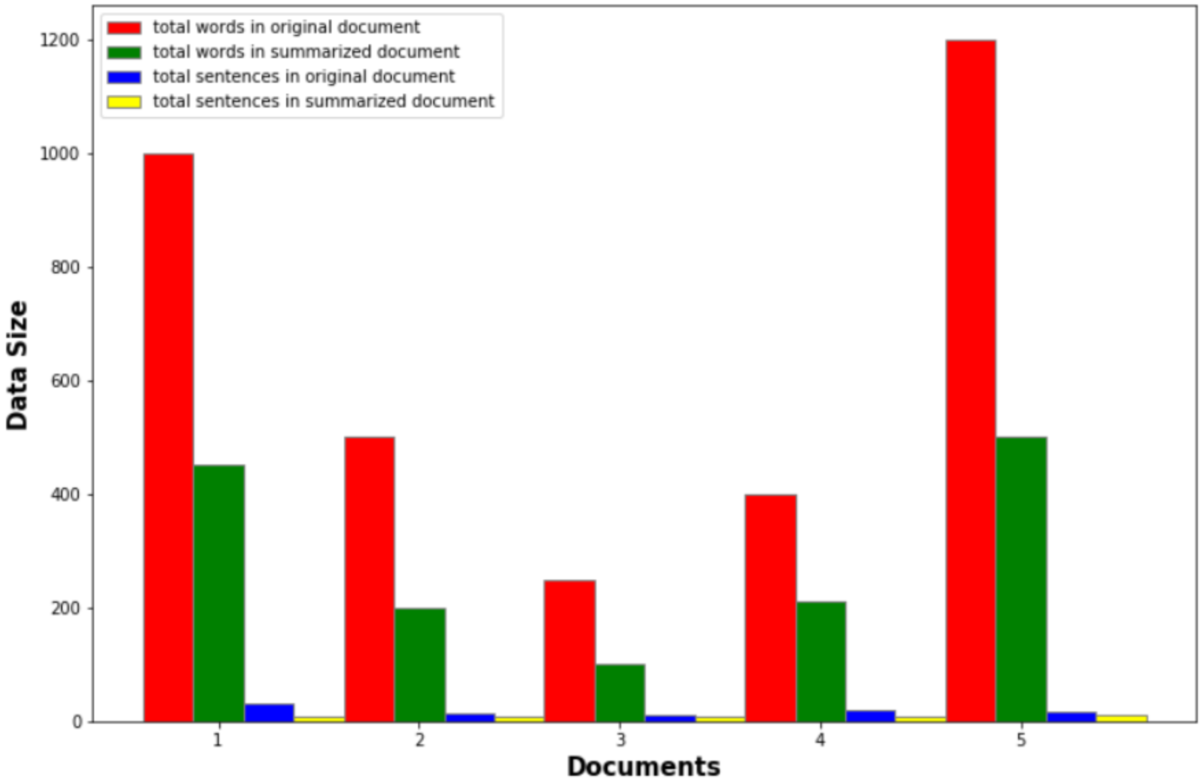

Fig. 4. LexRank Result

In the provided graph as shown in Figure 4, four parameters are taken into consideration for text summarization namely the total number of words in the original document, the total number of words in the summarized document, the total number of sentences in the original document, and the total number of sentences in the summarized document. According to the analysis of Figure 4, for Document 1:

- Out of 1000 words in the original document, 450 words are included in the summarized document.
- Out of 30 sentences in the original document, only 6 sentences are extracted in the final summary.

V. EXPERIMENTAL RESULTS

In the evaluation of the text summarization as given in Table II, the ROUGE metrics (Recall-Oriented Understudy for Gisting Evaluation) are used. Specifically, the ROUGE-N measure is implemented, which counts all shared words. The ROUGE-N measure is utilized because it assesses the adequacy of the summary. It is preferred for evaluation as it aims to minimize the need for further post-processing by humans. In this research, the F-score achieved by the LexRank algorithm is better compared to the Frequency-based method.

Table-II: ROUGE N=2 evaluation measure

Techniques	ROUGE 1	ROUGE 2
Frequency Based	0.804	0.794
Lex Rank	0.865	0.856

Retrieval Number:100.1/ijeat.E42000612523
DOI: 10.35940/ijeat.E4200.0612523
Journal Website: www.ijeat.org

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

Table III: Comparison of Evaluation Measure of Extractive Text Summarization Methods

Document number	Frequency Based			LexRank Algorithm		
	Precision	*Recall*	*F-Measure*	*Precision*	*Recall*	*F-Measure*
1	0.63	0.67	0.65	0.64	0.66	0.64
2	0.64	0.65	0.64	0.63	0.65	0.64
3	0.66	0.64	0.65	0.68	0.64	0.66
4	0.67	0.65	0.66	0.69	0.66	0.67
5	0.68	0.62	0.65	0.66	0.65	0.65
6	0.70	0.58	0.64	0.77	0.69	0.73
7	0.72	0.61	0.66	0.72	0.70	0.71
8	0.75	0.56	0.65	0.75	0.76	0.75
9	0.78	0.67	0.72	0.78	0.77	0.77
10	0.78	0.72	0.75	0.78	0.78	0.78

From the observation of the table as given in Table-III, it is noted that the maximum features are incorporated in Document 10. Comparing the results obtained from the frequency-based method and LexRank algorithm, it is observed that the LexRank algorithm produces superior results for many of the documents when compared to the frequency-based method. These values are obtained from the Marathi News article dataset.

V. CONCLUSION

In this research work, Extractive Text Summarization is performed on the Marathi News Article dataset, resulting in state-of-the-art results for generating concise text summaries. The final summary is extracted, containing a few sentences from the original text. Two extractive text summarization techniques, namely the frequency-based method and the LexRank algorithm, are evaluated and compared in this research. The results indicate that the LexRank algorithm outperforms the frequency-based method, as evaluated using the ROUGE metric and F-score. Further research can be conducted to explore the potential of deep learning techniques in text summarization.

DECLARATION

Funding/ Grants/ Financial Support	No, I did not receive.
Conflicts of Interest/ Competing Interests	No conflicts of interest to the best of our knowledge.
Ethical Approval and Consent to Participate	No, the article does not require ethical approval and consent to participate with evidence.
Availability of Data and Material/ Data Access Statement	Not relevant.
Authors Contributions	All authors have equally participated in various phases of the work like identification of the problem, review of literature, design of the problem, and execution of the algorithm.

REFERENCES

1. Kallimani, J. S., & Srinivasa, K. G. (2010, August). Information retrieval by text summarization for an Indian regional language. In Proceedings of the 6th International Conference on Natural Language Processing and Knowledge Engineering (NLPKE-2010) (pp. 1-4). IEEE. [CrossRef]
2. Rathod, Y. V. (2018). Extractive text summarization of Marathi news articles. Int. Res. J. Eng. Technol, 5, 1204-1210.
3. Desai, N., & Shah, P. (2016). Automatic text summarization using supervised machine learning technique for Hindi langauge. Int. J. Res. Eng. Technol, 5(06), 361-367. [CrossRef]
4. Shirwandkar, N. S., & Kulkarni, S. (2018, August). Extractive text summarization using deep learning. In 2018 fourth international conference on computing communication control and automation (ICCUBEA) (pp. 1-5). IEEE. [CrossRef]
5. Balabantaray, R. C., Mohammad, M., & Sharma, N. (2012). Multi-class twitter emotion classification: A new approach. International Journal of Applied Information Systems, 4(1), 48-53. [CrossRef]
6. Subramaniam, M., & Dalal, V. (2015). Test model for rich semantic graph representation for Hindi text using abstractive method. International Research Journal of Engineering and Technology (IRJET), 2(2), 113-116.
7. Jain, A., Arora, A., Morato, J., Yadav, D., & Kumar, K. V. (2022). Automatic text summarization for Hindi using real coded genetic algorithm. Applied Sciences, 12(13), 6584. [CrossRef]
8. Sarwadnya, V. V., & Sonawane, S. S. (2018, August). Marathi extractive text summarizer using graph-based model. In 2018 fourth international conference on computing communication control and automation (ICCUBEA) (pp. 1-6). IEEE. [CrossRef]
9. Dehru, V., Tiwari, P. K., Aggarwal, G., Joshi, B., & Kartik, P. (2021, March). Text summarization techniques and applications. In IOP Conference Series: Materials Science and Engineering (Vol. 1099, No. 1, p. 012042). IOP Publishing. [CrossRef]
10. Kakde, K.., & Padalikar, H. M. (2022). Context-based Sentiment analysis of Indian Marathi Text using Deep Learning. International Journal on Recent and Innovation Trends in Computing and Communication, 10(11), 71–76. https://doi.org/10.17762/ijritcc.v10i11.5782 [CrossRef]
11. Mamidala, K. K., & Sanampudi, S. K. (2021). Text summarization for Indian languages: a survey. Int J Adv Res Eng Technol (IJARET), 12(1), 530-538.
12. Sunitha, C., Jaya, A., & Ganesh, A. (2016). A study on abstractive summarization techniques in Indian languages. Procedia Computer Science, 87, 25-31. [CrossRef]
13. Sri, S. H. B., & Dutta, S. R. (2021, October). A survey on automatic text summarization techniques. In Journal of Physics: Conference Series (Vol. 2040, No. 1, p. 012044). IOP Publishing. [CrossRef]
14. D'silva, J., & Sharma, U. (2019). Automatic text summarization of Indian languages: a multilingual problem. J Theor Appl Inf Technol, 97(1).
15. Baruah, N., Sarma, S. K., & Borkotokey, S. (2019, February). Text summarization in Indian languages: a critical review. In 2019 Second International Conference on Advanced Computational and Communication Paradigms (ICACCP) (pp. 1-6). IEEE. [CrossRef]

AUTHORS PROFILE

Kirti Kakde, Pursuing Ph.D. **i**n Computer Application from IMED, Bharati Vidyapeeth Deemed to be University Pune, Maharashtra, India. Completed MCA from Mumbai University, Maharashtra, India. Her research areas include Machine Learning, Artificial Intelligence, Natural Language Processing, Programming Concepts Cyber Laws, and Data Bases.

Retrieval Number:100.1/ijeat.E42000612523
DOI: 10.35940/ijeat.E4200.0612523
Journal Website: www.ijeat.org

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

Dr. H. M. Padalikar, Associate Professor in IMED, Bharati Vidyapeeth Deemed to be University Pune, Maharashtra, India. He has completed Ph.D. in Computer Application from IMED, Bharati Vidyapeeth Deemed to be University Pune, Maharashtra, India. His research areas include Cloud Computing, Machine Learning, Artificial Intelligence, Natural Language Processing, and Programming Concepts.

Retrieval Number:100.1/ijeat.E42000612523
DOI: 10.35940/ijeat.E4200.0612523
Journal Website: www.ijeat.org

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)