IJNRD.ORG ISSN : 2456-4184
INTERNATIONAL JOURNAL OF NOVEL RESEARCH AND DEVELOPMENT (IJNRD) | IJNRD.ORG
An International Open Access, Peer-reviewed, Refereed Journal

Attention- Augmented Sequence-to-Sequence Approach for Gujarati Abstractive Summarization

Shilpa Serasiya[1], Uttam Chauhan[2]

[1]*Research Scholar, Computer Engineering, Gujarat Technological University, Gujarat, India*

[2]*Assistant Professor, Department of Computer Engineering, Government Engineering College, Modasa, Gujarat Technological University, Gujarat, India*

ABSTRACT

In recent years, text summarization has become a prominent challenge in the field of Natural Language Processing (NLP). It involves generating a concise and meaningful summary from a lengthy text document. There are two main approaches to summarization based on the type of output: extractive, which selects key sentences or phrases from the original text, and abstractive, which generates new sentences to convey the core information.While significant research has been conducted in extractive summarization for Indian languages, the development of effective abstractive summarization models remains limited—particularly for low-resource languages such as Gujarati. This work presents an efficient and accurate abstractive text summarizer tailored for the Gujarati language. Our model is built upon a Sequence-to-Sequence (Seq2Seq) framework employing an encoder-decoder architecture integrated with an attention mechanism. To enhance the preprocessing pipeline for Gujarati text, we introduce a custom preprocessor, designed to handle the linguistic and syntactic peculiarities of Gujarati. We curated a dataset comprising Gujarati news articles and their corresponding headlines to train and evaluate our model. Experimental results demonstrate that the proposed approach effectively captures the core semantics of the source text, generating fluent and human-readable summaries

Keywords: Abstractive Text Summarization, Seq-to-Seq model, Attention model, Gujarati Text

1. INTRODUCTION

In the 21st century, the world is experiencing an unprecedented surge in digital data, driven by rapid technological advancements. This explosion of data, encompassing news articles, academic publications, e-books, and social media content, poses a significant challenge for researchers and individuals seeking to extract meaningful insights efficiently. Manual summarization of large text documents is increasingly impractical, underscoring the need for automated solutions to distill essential information quickly.

Text summarization, a key area of natural language processing (NLP), has emerged as a vital tool to address this challenge. By transforming lengthy documents into concise summaries while preserving core information, summarization facilitates rapid understanding across diverse domains such as news, politics, medicine, and business. Applications include news article summaries, search engine snippets, product review condensations, automated research abstracts, and one-line email summaries, all of which minimize human effort while delivering critical insights. However, summarization, particularly for regional languages, remains a complex task due to the intricacies of language comprehension, contextual interpretation, and the need to emulate human-like common-sense reasoning.

Text summarization techniques are broadly classified based on input type, purpose, and output type, with extractive and abstractive summarization being the most prominent approaches. Extractive summarization involves selecting key sentences or phrases directly from the source text, while abstractive summarization generates new, concise expressions of the main ideas, often requiring a deeper understanding of the content. As shown in figure-1. To find the sentences required for summary, it uses statistical methods such as title/headline word. term frequency (TF), sentence length, cue method, location, similarity, proper noun, proximity, etc., even for Indian regional language most research works are carried out in extractive methods [11],[19],[10]. Abstractive Text summarization understands the document and rephrases the original text to new phrases to generate a summary which is close to the human-made summary. So it requires advanced machine learning techniques with natural language processing (NLP) to understand the document for summarization [11],[18].

This research focuses on abstractive summarization of Gujarati text, leveraging a sequence-to-sequence model with an attention mechanism to generate coherent and contextually relevant summaries. By addressing the unique linguistic characteristics of Gujarati, this study aims to advance automated summarization for regional languages, contributing to the broader field of NLP.

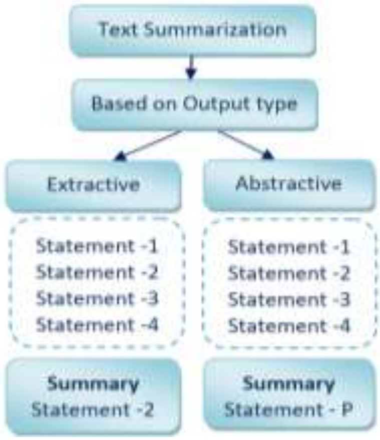


Figure 1. Text Summarization based on Output type

1.1 Research Gap and Motivation

Gujarati is an Indo-Aryan language predominantly spoken in the Indian state of Gujarat and by Gujarati communities around the world. With over 55 million speakers globally, it ranks as the sixth most spoken language in India. Despite its linguistic richness and widespread use, Gujarati remains significantly underrepresented in advanced Natural Language Processing (NLP) research, especially in the domain of abstractive text summarization. Existing research efforts in Gujarati have primarily focused on extractive summarization techniques, which involve selecting key sentences directly from the text. However, there is a notable absence of state-of-the-art abstractive summarization approaches that can generate coherent and human-like summaries by rephrasing and understanding the underlying semantics of the source text. This underexplored area presents a clear research gap that motivated our study.

Another major obstacle in this direction is the lack of a standard and high-quality dataset for Gujarati summarization tasks. Building such a dataset posed a significant challenge, as no large-scale annotated corpora are publicly available. To overcome this, we curated a custom dataset comprising Gujarati news articles and their corresponding summaries, collected from both online and offline sources. A large portion of the data was sourced from *Gujaratsamachar.com*, a reputed Gujarati daily newspaper, ensuring linguistic authenticity and domain diversity.

Driven by this gap, our research proposes an abstractive text summarization approach tailored for the Gujarati language. The model was trained on our curated dataset, and the experimental results show promising performance, demonstrating the feasibility and effectiveness of applying abstractive summarization techniques to a low-resource language like Gujarati.

In our proposed method, We have applied sequence to sequence model with encoder-decoder architecture. On the input text and three layers LSTM encoder, and to produce an efficient LSTM as decoder with Bahdanau attention model on the target text.The decoder makes an output sequence of fixed-length vectors produced by the encoder which encodes all the input sentences. We have transformed that method for Gujarati text summarization. In this paper, we have discussed the various work done in the Indian Language and its challenges, further discussing various necessary factors in text summarization to improve the efficiency to generate a more fluent and effective summary for the regional language. After the literature review, we discuss the methodology applied for the Gujarati language and give experimental results of the proposed model. Finally, we conclude the work as a stat-of art work.

2. LITERATURE REVIEW

Recent advancements in text summarization for Indian languages have shifted focus from extractive to abstractive methods, which generate new, concise summaries by rephrasing content. Abstractive summarization commonly employs sequence-to-sequence (seq2seq) models with encoder-decoder architectures, leveraging deep learning techniques such as Recurrent Neural Networks (RNNs), Long Short-Term Memory (LSTM) units, and Gated Recurrent Units (GRUs) (Nallapati et al., 2016; Chopra et al., 2016; See et al., 2017). These models encode input text into vectors, with attention mechanisms in the decoder focusing on key words to generate coherent summaries (Bahdanau et al., 2014). Enhancements like pointer-generator networks address out-of-vocabulary (OOV) words, while coverage mechanisms and convolutional models reduce word repetition (See et al., 2017; Lin et al., 2018).

For Indian regional languages, abstractive summarization faces additional challenges due to linguistic complexities such as inflection, agglutination, and gender variations. Handling inflection is a very important part in summarization, even though there is no fixed word structure. There are certain complex problems to be taken care of while analyzing the text like case, agglutination, and gender differences between languages. [14], [1], [3]. Notable work includes Sheikh et al. (2019), who developed a seq2seq model with bidirectional RNNs and LSTM for Bengali summarization for encoding at the input layer and attention model for decoding at the output layer[15]. Similarly, Wazery et al. (2022)[8] used LSTM, GRU, and Bidirectional LSTM with global attention for Arabic, while Sindhya et al. (2021)[11] applied a seq2seq model for Malayalam. Mohan et al. (2021)[21] implemented a deep learning-based approach for Telugu, and Geetha et al. (2015) proposed the sArAmsha system for Kannada, using abstraction schemes and named entity recognition.

Despite these advances, abstractive summarization for Gujarati remains underexplored. Existing research primarily focuses on extractive methods or preprocessing. For instance, Madria et al. (2019) analyzed stemmers for Gujarati, while Pinkesh et al. (2014) and Patel (2019)[24] investigated preprocessing techniques. Shah and Patel (2019) [26] utilized Textblob and Gensim for extractive summarization of Gujarati texts. However, no robust abstractive summarization model exists for Gujarati. Deep learning techniques, particularly RNN-based seq2seq models with attention mechanisms, have proven effective and language-agnostic, offering superior performance due to their ability to capture semantic nuances (Gupta and Gupta, 2019). This gap in abstractive summarization for Gujarati motivates our work to develop a seq2seq model with an attention mechanism tailored for Gujarati text.

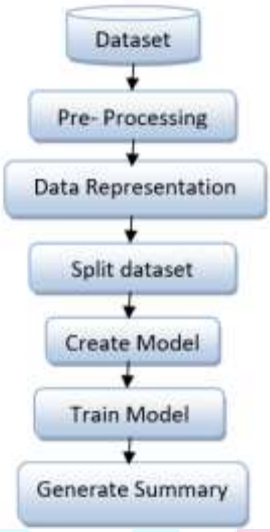
It's proven that Deep learning techniques are language agnostic and gives better performance because of the semantics concerned with it [6]. With the coming out of deep learning as a feasible alternative for many NLP tasks, its architectures have been generally accepted in abstractive text summarization, and they have since become state-of-the-art (Gupta and Gupta, 2019). In addition to the above Recurrent neural network-based sequence-to-sequence attention models have proven effective in abstractive text summarization [17].

3. METHODOLOGY

Language modeling is a cornerstone of modern natural language processing (NLP), underpinning applications such as text generation, machine translation, speech-to-text conversion, and text summarization. Text summarization, in particular, plays a critical role in distilling essential information from large volumes of text. In this study, we propose a methodology for developing an abstractive text summarizer specifically for Gujarati text, addressing the limited prior work in this area. Our goal is to create a model capable of generating concise, meaningful summaries that capture the core ideas of the input text.

To achieve this, we designed and implemented a sequence-to-sequence (seq2seq) model with an attention mechanism using TensorFlow. This framework is well-suited for abstractive summarization, as it allows the model to rephrase content rather than merely extract sentences. The encoder-decoder architecture processes the input Gujarati text by encoding it into a fixed-length vector representation, while the decoder, equipped with an attention mechanism, generates the summary by focusing on relevant parts of the input. The attention mechanism enhances the model's ability to prioritize key information, improving the coherence and relevance of the generated summaries.

Figure 2. Basic flowchart for Abstractive Text Summarization



Its always a challenging task while we work with regional text. To generate a proper summary, we need to undergo some procedures. Figure 2 gives a basic idea of all required procedures i.e. vocabulary counting, text pre-processing, word embedding, split data for train and test, etc.[22]. Most of the abstractive works use basic Machine Learning (ML) techniques. Encoder-Decoder-based sequence-to-sequence model is regularly used to create an abstractive text summarization model. Further model will trained and generate a summary.

3.1 Data Collection

Deep learning models require large datasets to achieve optimal performance. For Gujarati text summarization, limited structured datasets are available. To address this, we collected diverse Gujarati text data, including news articles, Facebook posts, and online reviews, primarily from gujaratsamachar.com. We manually created corresponding summaries for these texts, forming a dataset with two columns: articles and their summaries, tailored for abstractive summarization.

3.2 Data Preprocessing

Preprocessing is critical for regional languages like Gujarati due to unique linguistic features, such as specific symbols, constructions, and embedded English words. We developed a custom Gujarati preprocessor to clean the text, removing obstacles to ensure high-quality input like રૂ.->રૂપિયા, પ્રા.->પ્રાથમિક etc. The processed dataset, containing cleaned articles and summaries, enhances the model's ability to generate accurate summaries.

3.3 Data Representation

Tokenization is employed to break Gujarati text into words or subwords, converting them into numerical tokens for model processing. We used the NLTK library for word tokenization, mapping tokens to indices in a vocabulary or embedding space. To ensure uniform input size, we applied sequence padding, adding placeholder tokens to shorter sequences. This enables efficient batch processing and compatibility with neural network architectures, streamlining the summarization task.

3.4 Word Embedding

Word embeddings transform words into dense vectors, capturing semantic relationships and contextual nuances essential for abstractive summarization. By representing similar words with similar vectors, embeddings enable the model to understand synonyms, antonyms, and related concepts. We utilized embeddings as inputs to our sequence-to-sequence model, facilitating the generation of coherent and contextually accurate summaries. Techniques like Word2Vec, GloVe, or BERT enhance the model's generalization, improving summary quality for unseen texts.

3.4 Proposed model

Proposed Model for Abstractive Summarization of Gujarati Text Using Sequence-to-Sequence Model with Attention Mechanism works as follows.

3.4.1 Sequence-to-Sequence Model

The sequence-to-sequence (Seq2Seq) model is a neural network architecture designed to transform an input sequence into an output sequence, making it ideal for tasks like text summarization, where the input (a long Gujarati text) is converted into a shorter summary. The model comprises two main components: an encoder and a decoder[15], [16]. In our work, the Seq2Seq model processes a sequence of words from a Gujarati source document and generates a concise summary that retains the core meaning with fewer words.

3.4.2 Encoder-Decoder Architecture

Encoder: The encoder processes the input Gujarati text, compressing it into a context vector that captures the essence of the sequence. We utilize Long Short-Term Memory (LSTM) networks for the encoder, which effectively capture sequential dependencies through hidden states at each time step. The final hidden state, known as the context vector, encapsulates the input's key information and is passed to the decoder.

Decoder: The decoder takes the context vector as its initial hidden state and generates the summary one word at a time. We also employ LSTM for the decoder, which predicts each subsequent token based on the current hidden state and previously generated tokens. An attention mechanism is integrated into the decoder to dynamically focus on relevant parts of the input sequence, enhancing the coherence and relevance of the generated summary.

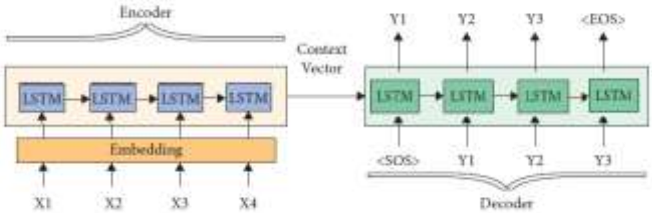


Figure 3. Encoder-Decoder architecture of Seq-to-Seq Model [8]

3.4.3 Attention Model

A key limitation of the basic sequence-to-sequence (Seq2Seq) model is its reliance on compressing the entire input sequence into a single context vector, which can lead to information loss, especially for long Gujarati texts. To overcome this, our model incorporates an attention mechanism, enabling the decoder to dynamically focus on different parts of the input sequence during each decoding step. Unlike the basic Seq2Seq model, which uses only the encoder's final hidden state, the attention mechanism leverages all encoder hidden states. For each decoding step, attention weights are calculated using a score function that measures the alignment between the decoder's current state and each encoder hidden state. These weights determine which parts of the input are most relevant for generating the current token, resulting in more coherent and contextually accurate summaries. The Architecture of Attention layer is shown in figure 4.

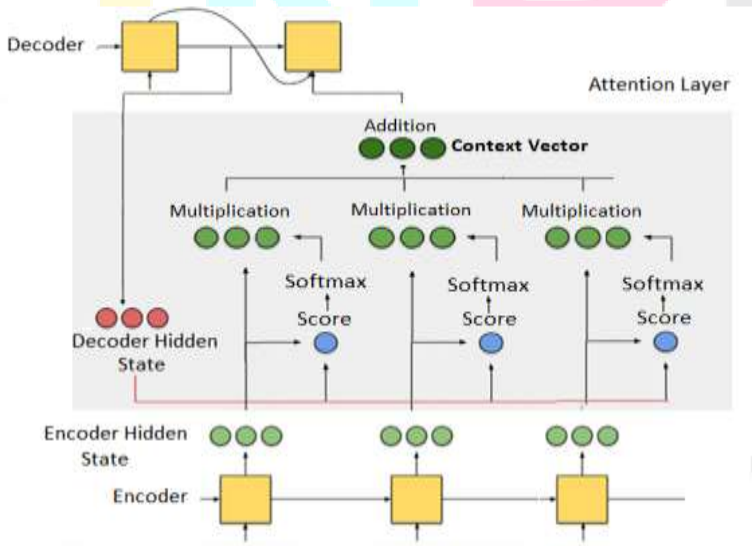


Figure 4. Architecture of Attention layer

The attention mechanism enhances the sequence-to-sequence (Seq2Seq) model by allowing the decoder to selectively focus on relevant parts of the input Gujarati text during summary generation. Below is the step-by-step working flow of the attention mechanism in our abstractive summarization model for Gujarati text:

Encoder Processing and Decoder Initialization: The encoder, implemented using Long Short-Term Memory (LSTM) networks, processes the input Gujarati text sequence. For each input token, the encoder generates a hidden state, capturing contextual information. These hidden states (collectively, the encoder's memory) represent the entire input sequence and are retained for use by the attention mechanism.The decoder, also an LSTM network, is initialized with the final context vector from the encoder (or a transformed version of it). During each decoding step, the decoder generates a hidden state based on the previously generated token and its current state.

Attention Score Computation: For each decoding step, the attention mechanism calculates a score to measure the relevance of each encoder hidden state to the decoder's current state. This is typically done using a score function, which computes alignment scores by comparing the decoder's current hidden state with each encoder hidden state.

Attention Weights: The scores are normalized using a softmax function to produce attention weights, which are values between 0 and 1 that indicate the importance of each encoder hidden state. These weights sum to 1, forming a probability distribution over the input sequence.

Context Vector Generation: A weighted sum of the encoder's hidden states is computed using the attention weights, producing a context vector specific to the current decoding step. This context vector dynamically emphasizes the most relevant parts of the input sequence for generating the next token in the summary.

Token Prediction: The decoder combines its current hidden state with the context vector to predict the next token in the summary. This process ensures that the generated token is informed by both the decoder's state and the focused input context, improving the summary's coherence and relevance.

Iterative Decoding: Steps 3–6 are repeated for each token in the output summary until a predefined length is reached or an end-of-sequence token is generated. At each step, the attention mechanism dynamically adjusts its focus, allowing the model to prioritize different parts of the input as needed.

This working flow enables the model to handle long Gujarati texts effectively, overcoming the limitations of the basic Seq2Seq model by providing context-aware, focused summarization. The attention mechanism ensures that the generated summaries are concise, coherent, and faithful to the meaning of the original text.

4. RESULTS AND DISCUSSION

This section presents a detailed analysis of the performance and effectiveness of the proposed encoder-decoder-based Seq2Seq model with an attention mechanism for summarizing Gujarati news articles. The evaluation leverages a comprehensive dataset and explores various configurations to optimize model performance.

4.1 Dataset Description

The dataset comprises 10,260 Gujarati news articles collected from both online and offline sources, primarily from Gujarat Samachar (gujaratsamachar.com), a reputable Gujarati daily newspaper. Each article is paired with a concise one- or two-line summary, covering diverse domains. The dataset was split into training, testing, and validation sets to evaluate the model's performance under different configurations.

4.2 Model Performance

Given the complex morphological structure like આજુ-બાજુ, આજુ બાજુ, આજુબાજુ, આવ્યો?, આવ્યો. આવ્યો!, lack of standardization, and extensive vocabulary of the Gujarati language, we conducted experiments with various dataset splits to identify the optimal configuration. The best performance was achieved with a 70% training and 30% testing split, with the testing set further divided to include a validation subset. This configuration consistently yielded superior results. To further refine the model, we experimented with different numbers of hidden layers in the encoder. As shown in Figure 5, a line chart illustrating accuracy across various hidden layer configurations, the model achieved the highest accuracy with three hidden layers. Consequently, we adopted this architecture for subsequent experiments.

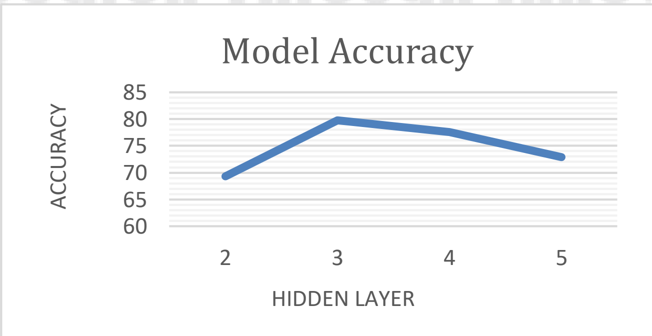


Figure 5. Line chart of accuracy at different hidden layer

Additional experiments were conducted to evaluate the impact of varying epoch values on model performance. The results demonstrate that the proposed model delivers state-of-the-art performance for Gujarati text summarization, highlighting the effectiveness of the attention-based Seq2Seq architecture with three hidden layers.

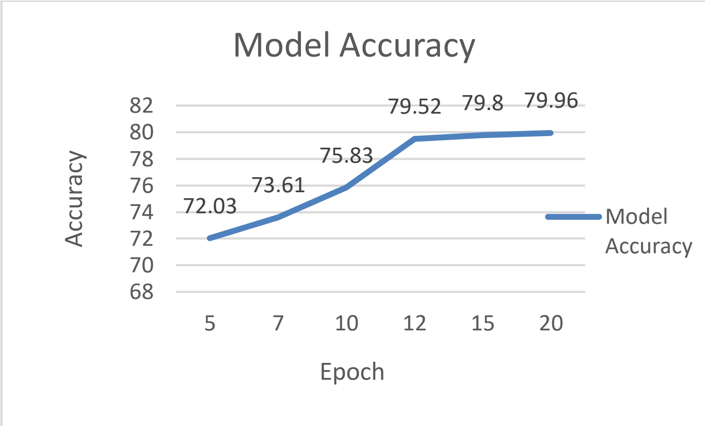


Figure 6. Model Accuracy Graph

Evaluation Metrics

To assess the quality of the summaries generated by the proposed encoder-decoder-based Seq2Seq model with an attention mechanism, we employed three standard evaluation metrics: ROUGE-1 (unigram), ROUGE-2 (bigram), and ROUGE-L (longest common subsequence). These metrics evaluate the overlap between the generated summaries and the reference summaries, measuring precision, recall, and F1-score for unigrams, bigrams, and the longest common subsequence, respectively.

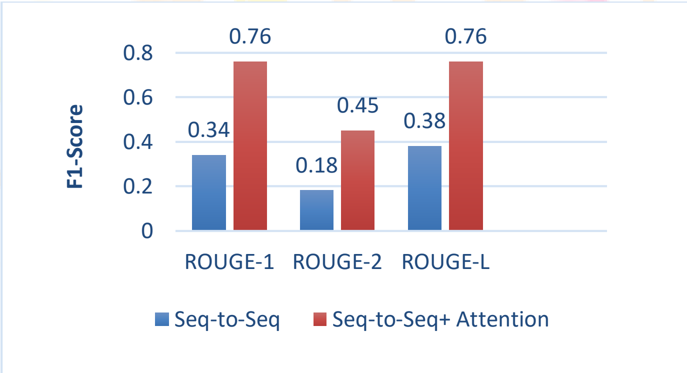


Figure 7. ROUGE Score with and without attention model Graph

Similar to performance observed in English and other regional languages, the Seq2Seq model with an attention mechanism outperforms the standard Seq2Seq model for Gujarati text summarization. The attention mechanism enhances the model's ability to focus on relevant parts of the input, resulting in higher ROUGE scores and improved summary quality.

5. CONCLUSION

This study presents a sequence-to-sequence model with an attention mechanism for abstractive summarization of Gujarati text, achieving promising results in generating coherent and meaningful summaries. The model effectively reduces training loss, demonstrating its capability to capture key information from input texts. A significant contribution of this work is the development of a standardized Gujarati news dataset, addressing the scarcity of structured data for this language. Our analysis reveals that Gujarati news articles exhibit complex morphological variations compared to English, compounded by the presence of English words, symbols, and abbreviations. To tackle these challenges, we developed a custom preprocessor and an LSTM-based encoder-decoder model with attention, which produces human-like summaries that preserve the essence of the original documents.

Large-scale experiments validate the effectiveness of the proposed model for Gujarati abstractive summarization. However, challenges remain, particularly in normalizing non-standard elements such as acronyms and symbols. Future work will focus on enhancing the model to generate multi-line summaries and improving handling of non-standard linguistic features to further advance abstractive summarization for Gujarati.

REFERENCES

[1] Ratul Sikder, Md. Monowar Hossain, F.M. Rahat Hasan Robi , "Automatic Text Summarization For Bengali Language Including Grammatical Analysis", International Journal of Science and Technology, Volume 8, Issue 06, ISSN 2277-8616 288, June 2019

[2] J S Kallimani, Shrinivasa K G, E Readdy B, "Information Extraction by an Abstractive Text Summarization for an Indian Regional Language", IEEE 2018.

[3] K K Mamidala, S K Snampuri, " Text Summarization for an Indian Language: A survey", International Journal of Advanced Research in Engineering and Technology (IJARET) Volume 12, Issue 1, January 2021, pp. 530-538"

[4] Jiawen Jiang, Haiyang Zhang, Chenxu Dai, Qingjuan Zhao, Hao, Hao Feng , "Enhancements of Attention-Based Bidirectional LSTM for Hybrid Automatic Text Summarization", IEEE Access, Volume -9 2021, 123660 - 123671

[5] R. Subha Shini , V.D. Ambeth Kumar "An Effective Analysis for Text Summarization by Word Sequence Generation ", conference paper, Research Gate 2021.

[6] S Shrenikaa , GouriPriya Raminia , Lunsavath BhagavathiDevia , B Geetavani "Abstractive Text Summarization By Using Deep Learning Models", Turkish Journal of Computer and Mathematics Education, Vol.12 -2021 , 2404– 2411

[7] Y.M. Wazery, Marwa E. Saleh , Abdullah Alharbi, and Abdelmgeid A. Ali, " Abstractive Arabic Text Summarization Based on Deep Learning", Hindawi Computational Intelligence and Neuroscience Volume 2022, 14 pages

[8] Prithwiraj Bhattacharjee, Avi Mallick, Md. Saiful Islam, and Marium-E-Jannat, "Bengali Abstractive News Summarization (BANS): A Neural Attention Approach", Springer- 2021, PP-41-51

[9] D. Suleiman and A. A. Awajan, "Deep learning based extractive text summarization: approaches, datasets and evaluation measures," in Proceedings of the 2019 Sixth International Conference on Social Networks Analysis, Management and Security (SNAMS), pp. 204–210, Granada, Spain, 2019

[10] B. Mohan Bharath, B. Aravindh Gowtham & M. Akhil ,"Neural Abstractive Text Summarizer for Telugu Language" , International conference on Advance in Inteligent system and computing, 2022

[11] Sunita C., A. Jaya, Amal Ganesh, "A Study on Abstractive Summarization Techniques in Indian Languages", 4th International Conference on Resent trends in computer Science and engineering, vol 87 PP-25-31, Elsevier-2016

[12] Dima Suleiman and Arafat Awajan, "Deep Learning based Abstractive Arabic Text Summarization using two layers encoders and one layer decoder" Journal of Theoretical and Applied Information Technology, Vol.98. pp: 3233-3244, September 2020.

[13] Viet Nguyen Quoc, Huong Le Thanh, Tuan Luu Minh, "Abstractive Text Summarization Using LSTMs with Rich Features", pp. 28–40, Springer Nature Singapore- 2020.

[14] Sunitha C, A Jaya, Amal Ganesh, "Automatic summarization of Malayalam documents using clause identification method",International Journal of Electrical and Computer Engineering (IJECE), ISSN: 2088-8708, Vol. 9,pp. 4929-4938, 2019.

[15] Sheikh Abujar, Abu Kaisar Mohammad Masum, Md. Sanzidul Islam, Fahad Faisal, Syed Akhter Hossain, "A Bengali Text Generation Approach in Context of Abstractive Text Summarization using RNN",Innovations in Computer Science and Engineering Proceedings of 7th ICICSE, doi.org/10.1007/978-981-15-2043-3_55pp- 509-518,Springer- 2020

[16] Abu Kaisar,Sheikh Abujar, Md A. I. Talukder, "Abstractive method of text summarization with sequence to sequence RNNs" IEEE Conference , December 2019DOI: 10.1109/ICCCNT45670.2019.8944620, December 2019.

[17] Rupal Bhargavaa, Gargi Sharmaa, Yashvardhan Sharmaa,"Deep Text Summarization using Generative Adversarial Networks in Indian Languages", International Conference on Computational Intelligence and Data Science, PP-147–153, Elsevier 2020

[18] Sabina Yeasmin, Priyanka Tumpa "Study of Abstractive Text Summarization Techniques", American Journal of Engineering Research (AJER) e-ISSN: 2320-0847 Volume-6, Issue-8, pp-253-260, 2017

[19] D. Suleiman and A. A. Awajan, "Deep learning based extractive text summarization: approaches, datasets and evaluation measures," in Proceedings of the 2019 Sixth International Conference on Social Networks Analysis, Management and Security (SNAMS), pp. 204–210, Granada, Spain, 2019.

[20] A. B. Al-Saleh and M. E. B. Menai, "Automatic Arabic text summarization: a survey," Artificial Intelligence Review, vol. 45, no. 2, pp. 203–234, 2016

[21] Surendrabikram Thapa, Sushruti Mishra, "Review of Text Summarization in Indian Regional Languages", 3rd International Conference on Computing Informatics and Networks (pp.23-32).

[22] Rahul Tangsali, Aabha Pingle, Aditya Vyawahare, Isha Joshi, Raviraj Joshi, "Implementing Deep Learning-Based Approaches for Article Summarization in India in Languages", arXiv:2212.05702v1-2022

[23] P. Patel,"Pre-processing phase of text summarization based on gujarati language" International Journal of Innovative Research in Computer Science and Technology, 2014 ISSN 2347–5552.

[24] Malvi Shah, Kalyani Patel,"સારાંશGujarati Text Summarizer" International Research Journal of Engineering and Technology, 2019.

[25] S K Nambiar, D Peater S, S. M Idicula, " Abstractive Summarization of Malayalam Document using Sequence to Sequence Model", 7th International Conference on Advanced Computing & Communication Systems (ICACCS) pp-347-352, 2021, DOI: 10.1109/ICACCS51430.2021.9442060

[26] P Varma, Anshul Varma, "Accountability of NLP Tools in Text Summarization for Indian Languages" , Journal of Scientific Research ,Volume 64, Issue 1, 2020.

[27] Shagun Sinha, Girish Nath Jha, "Abstractive Text Summarization for Sanskrit Prose: A Study of Methods and Approaches", Language Resources and Evaluation Conference (LREC 2020), Marseille, 11–16 May 2020.

[28] Minakshi Tomer, Manoj Kumar, "Improving Text Summarization using Ensembled Approach based on Fuzzy with LSTM", Arabian Journal for Science and Engineering- Springer -2020

[29] Ming-Hsiang Su, Chung-Hsien Wu, Hao-Tse Cheng "A Two-Stage Transformer-Based Approach for Variable-Length Abstractive Summarization", 2329-9290 (c) 2020 IEEE.

[30] Dong Qiu Bing Yang "Text summarization based on multi-head self-attention mechanism and pointer network", Springer 2021

[31] Jingwei Cheng, Fu ZhangXuyang Guo, "A Syntax-Augmented and Headline-Aware Neural Text Summarization Method", IEEE Access, Volume -8 2020, 218360-218371

[32] P. G. Magdum and Sheetal Rathi,"A Survey on Deep Learning-Based Automatic Text Summarization Models", Advances in Intelligent Systems and Computing, ISSN 2194-5357, Volume 1133, Springer 2019

[33] Nomi Baruah, Shikhar Kr. Sarma, S Borkotokey, "Text Summarization in Indian Languages: A Critical Review", International Conference on Advanced Computational and Communication Paradigms, IEEE -2019

[34] Jovi D' Silva, Dr. Uzzal Sharma, "Automatic text summarization of Indian Languages: A Multilingual Problem", Journal of Theoretical and Applied Information Technology, Vol.97, No 11 pp- 3026- 3037 June 2019 ISSN: 1992-8645

[35] Abs-Sum-Kan: An Abstractive Text Summarization Technique for an Indian Regional Language by Induction of Tagging Rules -IJRTE 2019

[36] Nithin Raphal, Hemanta Duwarah and Philemon Daniel,"Survey on Abstractive Text Summarization", IEEE -International Conference on Communication and Signal Processing, India April- 2018

[37] Kaichun Yao, Libo Zhang , Dawei Du , Tiejian Luo, Lili Tao, and Yanjun Wu, "Dual Encoding for Abstractive Text Summarization", IEEE Transections on CYBERNETICS, ISSN No: 2168-2267, 2018