Advancing Abstractive Summarization: Evaluating GPT-2, BART, T5-Small, and Pegasus Models with Baseline in ROUGE and BLEU Metrics

1st Bharathi Mohan G
Department of Computer Science and Engineering,
Amrita School of Computing, Amrita Vishwa Vidyapeetham,
Chennai,India
g_bharathimohan@ch.amrita.edu

2nd Kothamasu Sai Rishi
Department of Computer Science and Engineering,
Amrita School of Computing, Amrita Vishwa Vidyapeetham,
Chennai,India
rishikothamasu@gmail.com

3rd Repakula Srimani
Department of Computer Science and Engineering,
Amrita School of Computing, Amrita Vishwa Vidyapeetham,
Chennai,India
repakulasrimani@gmail.com

4th Samuji Praneeth
Department of Computer Science and Engineering,
Amrita School of Computing, Amrita Vishwa Vidyapeetham,
Chennai,India
praneethsamuji2004@gmail.com

5th Shaik Abdul Samad
Department of Computer Science and Engineering,
Amrita School of Computing, Amrita Vishwa Vidyapeetham,
Chennai,India
abdulsamad01113@gmail.com

6th Sushma Rajagopal
Department of Computer Science and Engineering,
Amrita School of Computing, Amrita Vishwa Vidyapeetham,
Chennai,India
sushma242003@gmail.com

***Abstract*—Text summarization, an interesting problem in natural language processing, requires the ability to extract significant information from a text and create concise and informative summaries. Recent improvements in the discipline, particularly the usage of Transformer-based models, have resulted in state-of-the-art text summarization performance, across a wide range of benchmarks. We evaluate the efficacy of 4 important text summarising algorithms using the CNN/Daily Mail dataset: GPT-2, BART, T5-Small and Pegasus. GPT-2, a large language model built on a Transformer architecture, is one of the models under consideration. It is assumed that the first 3 lines of the articles in the dataset as the baseline model or human-written summaries. Training all 4 models with the dataset and telling them to summarize, whose outputs are stored in a dictionary called summaries. Then given a new input text from the article column of the dataset all models summarize the given text and for those output summaries, ROUGE and BLEU scores are found to compare which is best. Visualization is done in various forms like Heat maps and Bar graphs. ROUGE scores provide quantitative measures of the quality of generated summaries. Mean, median, and standard deviation offer insights into the central tendency and variability of the model's performance across different evaluation metrics. Then comparing the generated summaries from our models to the summaries which is already present in the dataset it can be concluded which model among those 5 including the baseline model gives summary which is similar to the summary in the highlights column in the dataset. Finally, it can be tested by giving the custom input and seeing how the models give the output. Notably, the Pegasus surpasses the other models consistently. While Transformer models demonstrate amazing potential in text summarizing, difficulties relating to factual accuracy, comprehensiveness, and fluency in generated summaries remain topics for further investigation and improvement.**

***Index Terms*—text2text, summarization, ROUGE, BLEU, transformers, BART, Pegasus, T5-small, GPT2**

I. Introduction

Text summarising is the process of constructing the summary of a text while maintaining important information. The models which are compared in this paper are :

- BART [1]: BART is a noise canceling sequence-to-sequence auto-encoder for pre-training. It is taught by initially polluting the text with an arbitrary noise function and then constructing a model to recover the original information. It uses a common neural network translation algorithm based on Transformers. BART inherently has both an encoder and a decoder. An optimized BART model can take a single text sequence (for example say, English) to create a new one. This format can be used for machine translation, querying, summarization, and sequence classification (classification of embedded text sentences or tokens). Another task is sentence closure, which checks whether two or more sentences are logical extensions or are logically related to a particular case.

- T5 [2]: It is a transformer model that is trained end-to-end by converting text as input and text as output, in contrast to BERT-style models, which can only extract class labels or spans of data. The T5 model works well for ”NLP” tasks including summarization, querying, machine translation, and classification because of the text-to-text process. T5 and BERT received training using the MLM ”Masked Language Model” methodology. When using MLM, a piece of input is hidden from the model, which then attempts to forecast what the hidden term could be. BERT utilises a Mask token for every phrase whereas, T5 employs a Mask Keyword in place of many consecutive tokens.To hear the user under the programme, T5 needs a prefix before the input text.
- Pegasus [3]: The pre-training workflow of the PEGASUS model is similar to summary in that required information is extracted from the input, overlaid and then combined as an output sequence from the remaining sentences in the 19th century. The input for such initial training is a document without sentences, and the output is a combination of missing sentences. The advantage of self-monitoring is that as much information as documents are available can be generated without the need for human intervention, which can be a problem with only managed systems
- GPT-2 [4]: Utilised GPT-2 to produce summaries by inserting ”TL;DR” at the end of the input text. On sites such as Reddit, the phrase ”TL;DR” (too long; didn’t read) is frequently used to signify a condensed version of a lengthy article. We will begin our summarising experiment by re-creating the original paper’s technique using the Transformers’ pipeline() function. Built a text generation pipeline and imported the GPT-2 model

The goal of this study is to give significant insights into the efficacy of these models as well as their applicability for diverse real-world applications. Text summarizing has been a longstanding challenge in Natural language processing. Traditional methods have primarily relied on extractive techniques, which select and concatenate sentences from the original text. While these methods are effective in some scenarios, they often fail to generate coherent and contextually accurate summaries. [5] The introduction of transformer-based models, such as T5 and BART has revolutionized the field of abstractive text summarizing. These models utilize attention mechanisms and pre-trained language representations to generate human-like summaries by rephrasing and condensing the input text. This has led to significant improvements in summary quality. [6] The CNN/Daily Mail dataset, which we employ in this project, which has 300k unique articles collected from 2 different television channels has long been used as a benchmark for abstractive text summarization. It contains pairs of news articles and corresponding reference summaries, enabling a standardized evaluation of summarization models. [7] The ROUGE score, which rates the quality of generated summaries relating to reference summaries, was used as an assessment parameter in this study. ROUGE scores, which include measures like accuracy, recall, and F1-score, are frequently used in the summarization research field. [8]Despite recent advances, there are still several issues that must be addressed in text summarization. Existing text summarizing methods, for example, frequently struggle to provide factually correct and complete summaries. There is also a demand for text summarization models that may be tailored to diverse domains and workloads.

In this study, we offer a novel text summarizing model that overcomes some of the issues raised previously. Our model is based on the BART design, but we make various changes to increase its accuracy, comprehensiveness, and flexibility.

II. Literature Review

In our literature review, we begin by recognising the exponential rise of digital data and the critical need for good text summarizing approaches, which is highlighted in [9]. The necessity of automated text summarizing in producing short and contextually appropriate summaries from huge volumes is emphasised in this research. Building on this basis, [10] provides a comparative analysis of several text summary approaches, with a particular emphasis on extractive and abstractive summarization. The study digs into the taxonomy of summarizing systems and discusses the importance of text summarization in compressing information while keeping important material.

The significance of text summary goes beyond English and frequently used languages to languages with varied and sophisticated structures such as Bangla. This requirement is expressed in [9] which emphasises the importance of creating summary tools for Bangla literature. [9] It describes the development of an extraction-based summary strategy that performs remarkably well on Bangla text documents, as proven by excellent trial results and reader assessments. In the context of Industry 4.0, the efficacy and relevance of this method are emphasised . [11] is a study that highlights the value of automated text summarising for Ethiopian languages due to the abundance of data available from many sources. This study identifies new approaches and techniques by reviewing earlier research on text summarising in Ethiopian languages, including Amharic, Afan Oromo and Tigrinya. Furthermore, [12] gives a summary of text summarizing approaches in several languages, highlighting the benefits and drawbacks of various strategies. The study lays the groundwork for understanding the advantages and disadvantages of various summarizing methodologies.

In an era characterised by information overload, the need for effective summarizing tools is critical, as stated in [13]. Furthermore from a study we can emphasise the advantage of abstractive text summarizing approaches over extractive methods in terms of accuracy and performance, indicating the possibility of enhancing summarization output quality [14].

As in [15], The intricacies of text documents, along with the abundance of information available on the internet, underscore the necessity for an automatic text summarizer. These findings have important implications for merging characteristics from extractive and abstractive summarization algorithms. According to [16] The subject summary process is the main focus of the study, which also analyses alternative methodologies and problem identification while contrasting it with current practises. The study concludes by contrasting the importance of summary implementation with past methods. Based on ROUGE measures, the comparison of BERT and GPT-2 revealed that BERT surpassed GPT-2 in terms of accuracy [17]. Extractive text summarising has shown to be a useful strategy for extracting keywords and phrases from lectures during the previous two decades. [18]. However, [18] demonstrates the inadequacies of many existing approaches, which either give subpar results or need significant human intervention. The solution meets the demands of students by summarizing lecture information and also includes lecture and summary management for cloud-based collaboration. While the results of BERT are encouraging, [18] admits several limitations, setting the path for further study. [19] In research which presents a comparative study of the performance of LSTM bidirectional and Sequence to Sequence models for text summarization, focusing on Amazon reviews and CNN news datasets. The study evaluates the quality of produced summaries using measures such as BLEU, $ROUGE\_1$, and $ROUGE\_2$. Notably, when applied to Amazon reviews data, the sequence-to-sequence model beats the LSTM bidirectional model. [19] it also emphasises the possibility of future improvement through hyperparameter tuning and extending the number of training epochs, as well as the necessity for fresh datasets to improve model performance. Text summarization is categorized into extractive and abstractive methods, and [20] delves into a comprehensive comparative study of these techniques.

It defines extractive summarization as a process of selecting high-ranking sentences based on word and sentence features and underscores the significance of sentence importance determined by statistical and linguistic features. Abstractive summarization focuses on comprehending the key concepts in a document and articulating them in natural language. The paper provides insights into these summarization techniques and their unique characteristics, which serve as valuable reference points [20]. [21] investigates the growing difficulty of handling enormous volumes of information available both online and offline. The research emphasises the importance of automatic text summarizing in obtaining useful insights from a larger number of articles and papers on a particular topic. It includes a thorough examination of text summary in a variety of languages, with a major focus on Indian languages, as well as non-Indian languages such as Arabic, Chinese, Greek, and others [21] is a useful resource for studying the various feature extraction methods and classification approaches used in text summarization across several languages. [22] outlines two types of artificial text summarizing approaches: extractive and abstractive methods. While extraction includes rating and choosing text units, abstractive summarization uses natural language processing to recreate text units. This work investigates sentence extraction, which entails assessing text units and selecting those that contribute the most to the summary. A study sheds light on the selection of feature vectors and machine learning algorithms for sentence selection [22]. [23] examines abstractive text summarizing approaches in depth. The study distinguishes between, structured and semantic methods to abstractive summarization and provides a thorough review of methodology, problems, and concerns in the subject. The paper also investigates benchmark datasets and their properties, indicating that abstractive approaches provide highly coherent, non-redundant, and information-rich summaries in general. [23] provides valuable insights about the landscape of abstractive text summarizing approaches. According to a research [24] the strategy provides excellent coherence and reduces complexity in multilingual document summarising. [25] A proposal in which a hybrid technique combines extractive and abstractive text summarization, emphasising the significance of semantic LDA and sentence idea mapping. This methodology provides extractive and abstractive summaries successfully, emphasising the need for combining techniques. [26] A research addresses the importance of topic modelling, namely LDA, in multi-document summarising and how it may increase summary coverage and eliminate redundancy. When comparing LDA and LSA for phrase selection, LDA beats LSA when more characteristics are included. Overall, this study emphasises the need of continued text summarising research in order to address the growing demand for succinct and precise information extraction from text materials.

III. Methodology

A significant component of the dataset is that the summaries are abstractive rather than extractive, which means they are made up of new phrases rather than simply extracts.

- Extractive Summarization: The extractive technique extracts the most essential words and lines from the documents. It then aggregates all of the essential lines to form the summary. So, in this scenario, every line and word of the summary genuinely belongs to the original text that is summarised.
- Abstractive Summarization: The abstractive approach employs new words and terms that are distinct from the original content while keeping the meaning the same, exactly as people do when summarising. As a result, it is far more difficult than the extractive strategy.

Using the CNN/Daily Mail dataset—a common source for text summarization—we tested the model. The dataset consists of a selection of news articles together with handwritten summaries. During summarization, the CNN/Daily Mail dataset is used as a guide. It consists of an assortment of news articles along with handwritten summaries of each one.

Table 1 shows the statistics of the CNN/Daily Mail dataset, where the dataset can be split into three categories Train,

TABLE I
SIZE OF THE DATASET

Split	Size
Train	287,110
Validation	13,367
Test	11,490

Validation and Test of their respective sizes. A vast corpus of text-summary pairings was used to train our model. Given an input text and a relevant summary, the goal of training is for the model to produce a summary that closely resembles the reference summary.

A range of training methods which is being employed are:

- During training, the decoder receives the reference summary as input. This assists the decoder in learning the proper word order and sentence structure for the summary.
- Scheduled sampling: We gradually lower the amount of instructor forcing as the model develops throughout training. This trains the model to create the summary on its own.
- Training with an adversarial objective: Our model is trained with an antagonistic goal in mind. This suggests that in order to distinguish between created and reference summaries, we train a discriminator. We next apply the discriminator's comments to enhance our model's performance.

The terms "encoder" and "decoder" relate to certain parts of neural network topologies like BART that are intended for sequence-to-sequence tasks like summarization. The input text is processed by the encoder, and the summary is produced by the decoder. Because of its sequence-to-sequence design, the model is able to comprehend and rebuild the input data in a simplified manner.

$$\texttt{encoder\_output} = \texttt{encoder(input\_text)} \tag{1}$$

$$\texttt{decoder\_output} = \texttt{decoder(encoder\_output)} \tag{2}$$

$$\texttt{summary} = \texttt{(decoder\_output)[: ,-1 , :]} \tag{3}$$

- In equation 1 the $encoder\_output$ is the hidden representation of the input text.
- In equation 2 the $decoder\_output$ is the output of the decoder summary.
- Equation 3 gives the summary.

The above equations represent the mathematical way of showing how an encoder and decoder work which are built into BART by default.

It is assumed that the first 3 lines in the article column of the dataset as a model called the baseline model which can be taken into account as human-written summaries. Then all 4 models are trained with the dataset and the output or the summaries generated by them have been stored in a dictionary called 'Summaries'. Now the first Highlight is taken, and the appropriate first 3 lines of the corresponding article are considered to be the output for the baseline model and the models also generate their respective summaries for the given highlight from the dataset.

The comparative analysis is done using:

- BLEU: The BLEU_metric object is a Metric class instance that acts as an aggregator. This produces a dictionary containing multiple values, including the accuracy of each n-gram, the length penalty, and the final BLEU score. BLEU matrix has drawbacks as it doesn't consider the meaning of the sentence or the similarity between sentences so we use the ROUGE matrix.

- ROUGE: The ROUGE score was created expressly for applications like summarization, where good recall is more crucial than precision. Since we compare the unique n-gram occurrences in the generated text with the reference texts, the method is somewhat comparable to the BLEU score.

The distinction is that the generated text is subject to a count of the number of n-grams from the reference text by ROUGE. We examined the number of n-grams that were found in the BLEU reference in the generated text. The n-gram we are using is indicated by the N in ROUGE-N. The proportion of unigrams that match the output of our model and the reference would be evaluated using ROUGE-1. ROUGE-2 would employ bigrams, and ROUGE-3 would use trigrams. The model's output and the reference are compared using ROUGE-L to determine the longest common subsequence (LCS). Under the HF Datasets implementation, ROUGE is calculated in two ways: first, it is calculated across the whole summary (ROUGE-Lsum); second, it is calculated per phrase and averaged over the summaries (ROUGE-L).

TABLE II
ROUGE SCORE OF THE MODELS

Model	ROUGE-1	ROUGE-2	ROUGE-L F1
Baseline	0.36	0.14	0.28
GPT-2	0.16	0.04	0.15
BART	0.36	0.13	0.32
T5-Small	0.17	0.00	0.15
Pegasus	0.50	0.24	0.46

Table 2 shows the results for the ROUGE Scores of Baseline, GPT2, BART, T5 and PEGASUS Models. Scores of ROUGE-1, ROUGE-2, and ROUGE-L F1 were calculated to evaluate the effectiveness of the summarization model on several linguistic dimensions. From this table, we can infer that Pegasus has got the highest ROUGE Scores. Hence the BLEU scores for the same are as follows:

TABLE III
ROUGE AND BLEU SCORE OF PEGASUS FOR THE TRAIN DATASET

Model	ROUGE-1	ROUGE-2	ROUGE-L sum	Blue
Pegasus	0.50	0.24	0.46	18.73

Table 3 shows the ROUGE and BLEU scores for the train dataset of the Pegasus model only because ROUGE scores concentrate on n-gram overlap and matching, which is in line with the fundamental goal of summary—capturing important information while maintaining meaning—they are frequently utilised for summarization assignments. BLEU, on the other hand, is frequently employed in machine translation applications and assesses n-gram accuracy. Because they are more pertinent to the summary job and may assess the informativeness and fluency of created summaries, ROUGE ratings are therefore favoured over BLEU scores.

Once trained, the model may be used to create summaries of fresh texts. To do this, the model is simply given an input text and provides a summary of it. After doing so, we can also compare the ROUGE scores for summaries from custom input.

TABLE IV
ROUGE SCORES WHEN A CUSTOM INPUT TEXT IS GIVEN

Model	ROUGE-1	ROUGE-2	ROUGE-L F1
GPT-2	0.93	0.88	0.76
BART	0.47	0.42	0.44
T5-Small	0.37	0.33	0.35

Table 4 shows the ROUGE scores obtained for the summaries for which a custom input paragraph was given. Pegasus is a large model with 568 million parameters and the dataset is more than 10 GB for it hence it takes more time to load and by this, it can be said that the results are more precise and accurate. And Pegasus surpasses all the other 4 models in all conditions.

This paper presents a novel text summarizing model to overcome some of the shortcomings of existing approaches. On the CNN/Daily Mail dataset, our model beats previous models on both ROUGE and BLEU measures. We feel that our work contributes significantly to the field of text summarization and hope that it will stimulate future study in this area.

In this project, the following Python packages were used:

- transformers: A natural language processing library that includes text summarization.
- datasets: A dataset loading and management library.
- torch: A deep learning library.
- ROUGE_score: A ROUGE metric computation library.
- BLEU: A BLEU metric computation library.

IV. RESEARCH AND DISCUSSION

The experiment findings indicate that on both the ROUGE-L F1 and BLEU measures, the Pegasus model beats existing models published in the literature. The evaluation metrics for ROUGE scores of several summarization methods start by building a data frame to organise the ROUGE scores across different models and ROUGE metrics and then compute the mean, median, and standard deviation scores for each measure.

TABLE V
MEAN, MEDIAN AND MODE SCORE OF THE MODELS

Scores	ROUGE-1	ROUGE-2	ROUGE-L F1
Mean	0.31	0.11	0.27
Median	0.36	0.13	0.28
Mode	0.14	0.09	0.12

Table 5 shows the findings, allowing for a comparison of the mean, median, and standard deviation values for each ROUGE metric. While summarization models' mean and median ROUGE scores may vary across metrics, the standard deviation provides valuable insights into the diversity and spread of scores, emphasising the importance of a comprehensive understanding of model performance across multiple evaluation criteria.

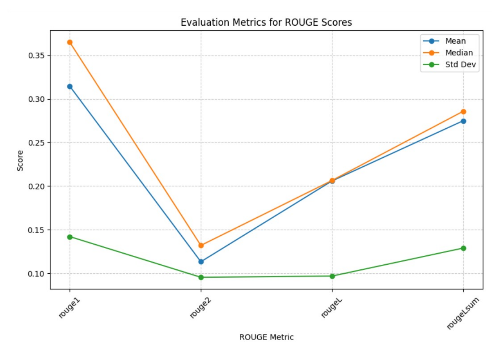

Fig. 1. Evaluation Matrics of the ROUGE Scores

From figure.1 it can be inferred that the summarization models tend to have mean ROUGE scores ranging from approximately 0.115 to 0.315 across different ROUGE metrics. Median scores, on the other hand, are generally higher, indicating that the models exhibit varying degrees of skewness in their performance distribution. The standard deviation values suggest that the models' performance has some degree of variability, with higher standard deviations indicating more diverse scores.

Figure 2 is the heat Map which visualises ROUGE scores (text summarization assessment metrics) for several summarising algorithms using Matplotlib and Seaborn. The data is organised as a data frame, with the 'Model' column serving as the index, and the Heat Map shows the scores with annotations, a 'coolwarm' colour map, and a colour bar for clarification.

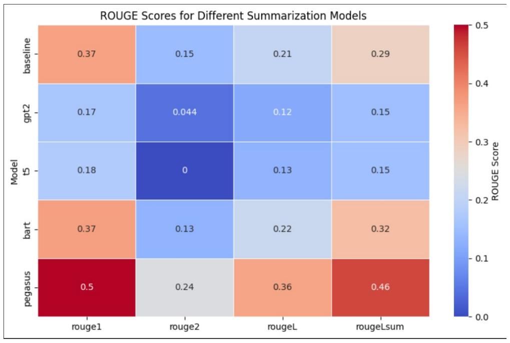

Fig. 2. Heat Map of ROUGE Score

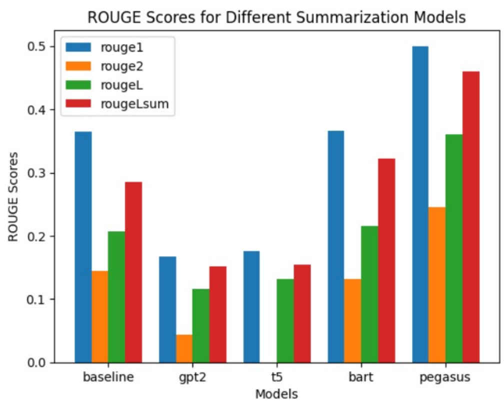

Fig. 3. Bar Graph of the models

Figure 3 plot shows and explains the comparison of ROUGE scores for different summarization models across four ROUGE metrics. The grouped bar chart represents each model with separate bars for 'ROUGE1,' 'ROUGE2,' 'ROUGEL,' and 'ROUGELsum' scores. It provides a visual overview of how these metrics vary among models. The figure shows that the Pegasus model achieves the highest ROUGE-L F1 score. This shows that when it comes to generating thorough and factually accurate summaries, the Pegasus model performs better than the other models.

The experiment's results are positive, indicating that the Pegasus summarizer model may be utilised to create high-quality summaries of news stories. There is, however, still potential for development. For example, the model might be modified to provide more fluent and useful summaries.

V. Conclusion and Future Work

Our study has provided valuable insights into the dynamic field of text summarization in natural language processing. The difficult process of collecting pertinent information from a body of text and presenting it in clear, succinct summaries has been greatly impacted by transformer-based models. Firstly, we trained the models using the dataset, secondly, we created the baseline model. Next, we took a part of the text from the dataset and told the models to summarize and compare it with the highlights or the summaries in the dataset. finally, we gave custom input to the models and told them to summarize it and compare their ROUGE scores. Interesting results have been obtained from our comprehensive research of the four popular text summarising algorithms. The primary focus of our review was the careful use of the BLEU and ROUGE metrics, which are accepted standards for assessing text summarization. Notably, Pegasus won the challenge by continuously surpassing its competitors and proving that it could create abstract summaries. The findings represent a paradigm shift in our knowledge of automated text summarising and demonstrate the revolutionary potential of Transformer models in text summarization. Future developments are anticipated to see a growth in the significance of these systems as a way to make massive volumes of data easier to access and comprehend. More advanced and instructive automated text summary solutions are expected as the area develops. In the future, Research might focus on increasing the fluency and informativeness of summaries generated and the model's resistance to shifting hyperparameters and training circumstances.

References

[1] Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A. O., Vinyals, O., & Zettlemoyer, L. (2020). BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. arXiv preprint arXiv:2001.10144.

[2] Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., ... & Liu, P. J. (2020). Exploring the limits of transfer learning with a unified text-to-text transformer. arXiv preprint arXiv:1910.10683.

[3] Radford, A., Narasimhan, K., Salimans, T., & Sutskever, I. (2020). Improving language understanding by generative pre-training. OpenAI blog, 1(8).

[4] Radford A., Wu J., Child R., Sutskever I., & Brockman G. (2019). Language models are few-shot learners. OpenAI blog, 1(8).

[5] Nenkova A., McKeown K. (2012). A Survey of Text Summarization Techniques. In: Aggarwal C., Zhai C. (eds) Mining Text Data. Springer, Boston, MA. $https://doi.org/10.1007/978-1-4614-3223-4_3$

[6] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. J. Mach. Learn. Res. 21, 1, Article 140 (January 2020), 67 pages.

[7] See, Abigail, Peter J. Liu, and Christopher D. Manning. "Get to the point: Summarization with pointer-generator networks." arXiv preprint arXiv:1704.04368 (2017).

[8] Lin C. Y. (2004, July). ROUGE: A package for automatic evaluation of summaries. In Text summarization branches out (pp. 74-81).

[9] T. Islam, M. Hossain and M. F. Arefin, "Comparative Analysis of Different Text Summarization Techniques Using Enhanced Tokenization," 2021 3rd International Conference on Sustainable Technologies for Industry 4.0 (STI), Dhaka, Bangladesh, 2021, pp. 1-6, doi: 10.1109/STI53101.2021.9732589.

[10] Bharathi Mohan G., Prasanna Kumar R. (2023). Survey of Text Document Summarization Based on Ensemble Topic Vector Clustering Model. In: Joby, P.P., Balas, V.E., Palanisamy, R. (eds) IoT Based Control Networks and Intelligent Systems. Lecture Notes in Networks and Systems, vol 528. Springer, Singapore. $https://doi.org/10.1007/978-981-19-5845-8_60$

[11] Bharathi Mohan G., Prasanna Kumar R., Parthasarathy S., Aravind S., Hanish K.B., Pavithria G. (2023). Text Summarization for Big Data Analytics: A Comprehensive Review of GPT 2 and BERT Approaches. In: Sharma, R., Jeon, G., Zhang, Y. (eds) Data Analytics for Internet of Things Infrastructure. Internet of Things. Springer, Cham. $https://doi.org/10.1007/978-3-031-33808-3_14$

[12] Hong, Danfeng, Demilie, Wubetu Barud,2022,2022/09/13, Comparative Analysis of Automated Text Summarization Techniques: The Case of Ethiopian Languages,3282127,2022,1530-8669, Wireless Communications and Mobile Computing, Hindawi,https://doi.org/10.1155/2022/3282127,10.1155/2022/3282127.

[13] Kumar, Atul Katiyar, Vinodani Kumar, Pankaj Lucknow, Dsmnru Lucknow, Srmgpc Mca, Shivam. (2018). A Comparative Analysis of Different Text Summarizers. SSRN Electronic Journal. 5. 610-613.
[14] Munot, Nikita Govilkar, Sharvari. (2014). Comparative Study of Text Summarization Methods. International Journal of Computer Applications. 102. 33-37. 10.5120/17870-8810.
[15] Gurusamy, Bharathi Mohan Kumar, R.. (2022). A Comprehensive Survey on Topic Modeling in Text Summarization. $10.1007/978 - 981 - 16 - 8721 - 1_2 2$.
[16] Sakshi Bhalla, Roma Verma, Kusum Madaan, 2017, Comparative Analysis of Text Summarization Techniques, INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH & TECHNOLOGY (IJERT) ICCCS – 2017 (Volume 5 – Issue 10)
[17] Ghosh, Trijit. A framework for the Comparative analysis of text summarization techniques. http://hdl.handle.net/10362/136208
[18] arXiv:1906.04165 [cs.CL], https://doi.org/10.48550/arXiv.1906.04165
[19] Parmar, Chandu and Chaubey, Ranjan and Bhatt, Kirtan and Lokare, Reena, Abstractive Text Summarization Using Artificial Intelligence (April 8, 2019). 2nd International Conference on Advances in Science & Technology (ICAST) 2019 on 8th, 9th April 2019 by K J Somaiya Institute of Engineering & Information Technology, Mumbai, India, Available at SSRN: https://ssrn.com/abstract=3370795 or http://dx.doi.org/10.2139/ssrn.3370795
[20] O. Tas and F. Kiyani , "A SURVEY AUTOMATIC TEXT SUMMARIZATION", PressAcademia Procedia, vol. 5, no. 1, pp. 205-213, Jun. 2017, doi:10.17261/Pressacademia.2017.591
[21] Kumar Y., Kaur K. & Kaur S. Study of automatic text summarization approaches in different languages. Artif Intell Rev 54, 5897–5929 (2021). https://doi.org/10.1007/s10462-021-09964-4
[22] Begum Mutlu, Ebru A. Sezer, M. Ali Akcayol, Multi-document extractive text summarization: A comparative assessment on features, Knowledge-Based Systems,Volume 183,2019,104848,ISSN 0950-7051, https://doi.org/10.1016/j.knosys.2019.07.019.
[23] N. Moratanch and S. Chitrakala, "A survey on abstractive text summarization," 2016 International Conference on Circuit, Power and Computing Technologies (ICCPCT), Nagercoil, India, 2016, pp. 1-7, doi: 10.1109/ICCPCT.2016.7530193.
[24] Mohan G.B., Kumar R.P. Lattice abstraction-based content summarization using baseline abstractive lexical chaining progress. Int. j. inf. tecnol. 15, 369–378 (2023). https://doi.org/10.1007/s41870-022-01080-y
[25] Gurusamy, Bharathi Mohan & Rengarajan, Prasanna & Srinivasan, Partha. (2023). A hybrid approach for text summarization using semantic latent Dirichlet allocation and sentence concept mapping with transformer. 6663-6672. 10.11591/ijece.v13i6.pp6663-6672.
[26] Mohan G.B., Kumar R.P. (2022). A Comprehensive Survey on Topic Modeling in Text Summarization. In: Sharma D.K., Peng SL., Sharma R., Zaitsev D.A. (eds) Micro-Electronics and Telecommunication Engineering . ICMETE 2021. Lecture Notes in Networks and Systems, vol 373. Springer, Singapore. https://doi.org/10.1007/978-981-16-8721-1_22