# Bangla Text Summarization with Few-shot Learning

Anusree Roy*
*Department of Computer Science and Engineering
North South University
anusree.roy@northsouth.edu

Afra Anika Rahman†
†Department of Computer Science and Engineering
North South University
afra.rahman2@northsouth.edu

Mushfiq Uzzaman Suharto‡
‡Department of Computer Science and Engineering
North South University
mushfiq.suharto@northsouth.edu

***Abstract*—This report presents how Few-Shot learning can be effective in text summarization using an LLM model for a low-resource language like Bangla. We leverage the advantage of the mT5multilingualXLSum model for generating summaries which is trained for multilingual tasks over 45 languages including Bangla by employing Few-Shot learning techniques. The method involves constructing prompt with example pairs of long text and their corresponding summaries along with the new text to guide model in generating a coherent and concise summary for the new text. The method was evaluated on metrics like BLEU and ROGUE and a competitive score was achieved which indicates the effectiveness of our approach. The achieved scores are The ROUGE-1, ROUGE-2 and ROUGE-L F1 scores are- 0.50, 0.27, and 0.44 respectively. While, the BLEU score is 0.18. A significant overlap and similarity between the generated and reference summary is noticed through the obtained scores. This approach represents the capability of producing high-quality summaries with little data and training providing a solution for tasks like Bangla text summarization**

***Index Terms*—Bangla text summarization, prompt engineering, natural language processing, language models, text summarization.**

## I. Introduction

There has been considerable progress made in the discipline of natural language processing (NLP) over the past few years primarily because of some powerful language models like BERT, GPT, T5 among many others. They have shown outstanding potential in different situations- text generations, translations, summaries, and so on. One specific but most useful way these tools have been used is through text summarization which aims at producing brief coherent summaries. In the past, creating good summary models needed significant volumes of annotated data and a lot of training. But these days, we can make a great performing model with even just a small number of instances and all this is possible with the help of few-shot instructing. Few-shot teaching uses models that were already trained and a small number of samples to easily fit in new jobs, enabling it to be the easiest way that works when there's no information available about that. Given that Bengali is spoken by over 250 million people worldwide [1], the need for efficient text summarization tools is essential. The ability to summarize Bengali news articles can greatly enhance the accessibility and distribution of information, enabling readers to stay informed without needing to look through extensive content [2]. Even though there have been a significant amount of recent successes in the field of Bengali text summarization, there still remain many limitations that are yet to be solved. The most important limitation is the unavailability of openly accessible resources – mainly a Bengali language domain. Another notable shortcoming is that these text summarization models repeat inaccurate factual details and tend to repeat themselves [3]. In addition to this, a limitation that is constantly faced by others is that the abstractive summarization model sometimes generates incoherent or grammatically incorrect summaries, reflecting the challenge of effectively capturing and rephrasing the original text [4]. Moreover, the limitation was faced by others in the summary generation process too as the model could only make summaries with limited words [5]. This paper examines how to achieve text summarization by using a fine-tuned model of MT5 on the XLSUM dataset, the mT5multilingualXLSum model. The MT5 is multilingual T5, a variation of the T5 model pretrained in 101 languages. The main purpose of the paper is to show how few-shot learning can be effective in condensing Bengali written content. We present a few samples together with their corresponding condensed versions of the model, then we provide a new body of text to be summarized. Afterward, we had our model generate a summary and compare its BLEU and ROGUE scores with those from the summaries given by the GPT 3.5 model on a specific prompt. While text summarization methods have been extensively studied for English text, this project addresses the need for summarization techniques tailored to Bengali text. The novelty of the paper is that it uses few-shot learning for summarizing the text in Bangla language with the help of an LLM. Also, the reference summary was generated by the GPT 3.5 model for comparing it with the model-generated summary for evaluating its quality. This process helps us show how one can use few-shot learning on text summarization tasks: it provides a way of producing good-quality summaries when there is little data available in terms of low-resource languages like Bangla. Various areas such as news briefing, content creation, and information extraction can greatly gain from this method because it has many prospects for application in these spheres.

## II. Related Work

**Text summarization with Prompt Engineering** Few-shot learning has made remarkable strides, particularly in computer vision, and has expanded to numerous other fields [6]. The accuracy for tasks such as the 5-way 1-shot classification on miniImageNet has improved significantly, demonstrating the potential and versatility of FSL methodologies. The continued exploration of diverse approaches suggests a dynamic and evolving field with substantial room for future innovation and application.

In the paper by [7], the studies highlight the potential of using prompt engineering for pre-trained language models to advance clinical NLP and a range of medical AI applications, especially when annotated data is scarce. Prompt-based learning tunes models for new tasks by defining task templates instead of fine-tuning, helping models achieve strong performance in zero-shot learning scenarios where they generalize to new classes without examples.

In another paper by [8], the main objective was to identify the cause-effect relationship between the formulation detail of the prompt and the performance of the automated medical report. The study demonstrated medical dialogues using summarization techniques and used two types of prompting strategies to enhance the performance of automated medical reporting. Then, the automated medical reports were evaluated using the ROUGE score and human evaluators.

In another paper by [9] the experiments conducted showed that in-context learning-based evaluators performed well in comparison to learned evaluation frameworks for the task of text summarization, establishing state-of-the-art on dimensions such as relevance and factual consistency. The effects of factors such as the selection and number of in-context examples on performance were analyzed and the efficacy of in-context learning-based evaluators in evaluating zero-shot summaries written by large language models such as GPT-3 were studied.

The paper by [10] investigates few-shot learning paradigms and observes that its performances can be matched or exceeded by simple 0-shot prompts. By exploring the nature of successful 0-shot prompts, the paper proposes general methods of prompt engineering through the lens of natural language semiotics.

Work by [11] concentrates on improving prompting performance over a broad set of models and tasks. It does so by combining the outputs of multiple effective but imperfect prompts, aiming to improve results. For each task input, individual prompts generate votes for the correct label, which are then aggregated to determine the final prediction. The paper introduces ASK ME ANYTHING PROMPTING (AMA), a straightforward approach that remarkably allows open-source large language models with 30x fewer parameters to exceed the few-shot performance of GPT3-175B.

In this paper [12], a collection of straightforward and complementary techniques for fine-tuning language models called LM-BFF—Better Few-shot Fine-tuning of language models was introduced. Key methodologies comprised: (1) prompt-based fine-tuning combined with an innovative pipeline for automating prompt generation, and (2) a refined strategy for dynamically and selectively incorporating demonstrations into each context. Additionally, a systematic evaluation to assess few-shot performance across various NLP tasks was conducted.

This survey by [13], presents a comprehensive framework for categorizing the diverse range of prompting techniques that have emerged in recent academic literature for conversational Pre-trained Large Language Models. The techniques and approaches have been classified into seven distinct categories, each representing high-level conceptualizations of their methodologies and intended uses. By organizing these methods into cohesive groups can help practitioners with a clear roadmap for selecting the most suitable prompting strategies for their specific applications. Whether practitioners aim to generate creative content, answer questions, or engage in natural language conversations, the proposed framework enables them to easily identify relevant categories and explore techniques aligned with their particular goals.

In the paper by [14], CEDAR (Code Example Demonstration Automated Retrieval), a technique for automatically creating prompts by retrieving code demonstrations similar to a developer's task using embedding or frequency analysis, was introduced. This approach was applied to both statically and dynamically typed programming languages and to tasks such as test assertion generation and program repair. CEDAR was evaluated against state-of-the-art task-specific and fine-tuned models, demonstrating its effectiveness in few-shot learning settings without requiring extensive task or language-specific training. The findings suggest that CEDAR could be beneficial in multilingual and multitasking environments with minimal examples and effort. The evaluation involves using CODEX to test various few-shot learning scenarios, including zero-shot, one-shot, and n-shot prompts, both randomly and systematically selected, with or without natural language descriptions. This paper inspects the ability of ChatGPT to evaluate the factual inconsistency under zero-shot settings for tasks like binary summary ranking, entailment inference, and consistency rating [15]. The experiment found that Chatgpt outperformed other evaluation metrics in three tasks. This indicates that it has great potential in terms of factual inconsistency evaluation.

This paper proposed an enhanced approach to constructing prompts by introducing tagged facts generated automatically through semantic analysis [16]. This was found that this approach to constructing prompts improves the performance on the task of code summarization.

**LLMs in the Domain of Text Summarization** The introduction of mT5 and mC4 marks a significant step forward in multilingual language understanding [17]. By extending the T5 model and C4 dataset to support multiple languages, the researchers demonstrated strong performance across various benchmarks. The identification and mitigation of illegal predictions enhance the robustness of multilingual models in zero-shot scenarios. The open release of code and datasets aims to drive further research and development in this field.

This paper [18] introduces a monolingual BERT model for the Bengali language called Bangla-BERT. They constructed Bengali language model dataset, BanglaLM. This paper resolves the mBERT's limitation for Bengali trained on limited and more structured data only and mixed weights issues among 104 languages. The model outperformed other models and surpassed all prior state-of-the-art results by 3.52%, 2.2%, and 5.3%.

This paper [19] embarks on an exploration of text summarization with a diverse set of LLMs, including MPT-7b-instruct, falcon-7b-instruct, and OpenAI ChatGPT textdavinci-003 models. The primary objective is to provide a comprehensive understanding of the performance of Large Language Models (LLMs) like MPT-7b-instruct, falcon-7b-instruct, and OpenAI ChatGPT textdavinci-003 when applied to different datasets like xsum and CNN-DailyMail News Text Summarization datasets. From their experiment, text-davinci-003 outperformed the others on both datasets.

The study [20] introduces an approach where a single text is first summarized using four pre-trained Bengali text summarization models, namely- mt5 XLSum, mT5 CrossSum, Scibert uncased, and mT5 by Shahidul. Then its respective human-written reference summary is taken. The proposed approach for choosing the best summary involves evaluating the similarity of all candidate summaries with the reference summary (human-written). Borah et al. [21] obtained the T5 model's effectiveness in abstractive text summarization on datasets like CNNDM, MSMO, and XSUM. Their ROUGE and BLEU scores demonstrated the model's superior capability in generating concise summaries, particularly excelling with the MSMO dataset.

## III. Methodology

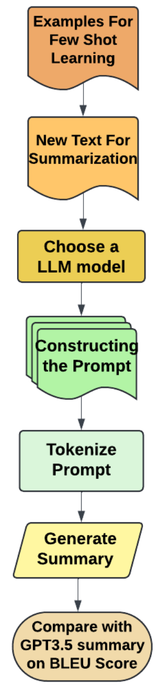


Fig. 1: Workflow of Bengali Text Summarization with Few-Shot learning

Fig. 1 shows the primary steps taken in Bangla text summarization with Few-Shot learning. Starting from creating examples to constructing the prompt with a new text and then passing it to the model for generating the summary of it and lastly evaluating it on BLEU and ROGUE scores by comparing the generated one with the reference summary given by GPT 3.5 model.

### A. Creating Examples

For Bangla text summarization with few-shot learning, some examples are prepared. Each of these examples contains an input text—a longer one and an output text—a summary of the input text. These examples are essential when teaching a language model how to produce summaries.

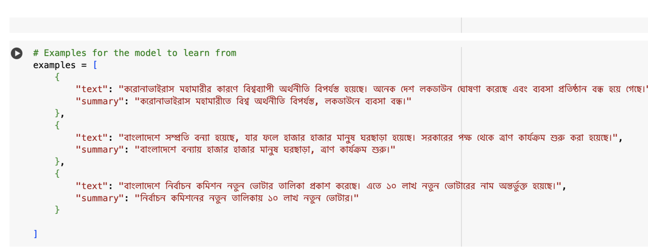


Fig. 2: Some examples for the model to learn from

Fig. 2 shows some examples that contains longer text and its summary pairs which model uses to learn the pattern from and generate summary for the new text.

### B. New Text for Summarization

A new text is selected and the model will generate a summary for it. The prompt is selected in a way that the previous examples are merged with the new text and tells the model to generate a summary for the new text by learning from the pattern of the examples of longer text and its summary. The new text is the longer one that is given to the model but its summary is not provided.

### C. Model Selection

We have selected the csebuetnlp/mT5multilingualXLSum model as it is a different form of the T5 (Text-To-Text Transfer Transformer) model. The MT5 is the multilingual version of T5 model, pre-trained on 101 languages for solving multilingual tasks. The mT5multilingualXLSum model used in the project for Bangla text summarization with few-shot learning was fine-tuned on the XLSum dataset consisting of article-summary pairs from BBC News over 45 languages, thus making it qualify for multilingual text summarization works.

### D. Constructing the Prompt

Prompt construction is one of the most crucial steps for this few-shot learning approach. The new text is merged with the previous examples containing long text and its summary pairs along with a prompt that tells the model to generate the summary for the new text by learning from the example pairs. Some delimiters were used to separate the examples from the new text to clear out any sort of vagueness.

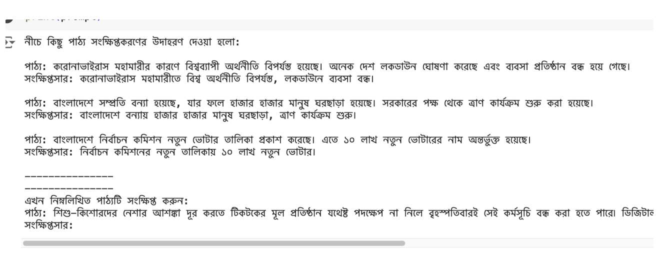


Fig. 3: Construction of prompt

Fig. 3 shows how prompts are constructed with merging new text and example pairs along with the prompt asking to generate summary by learning from the examples.

### E. Model Inference

The constructed prompt including the new text and the previous example pairs is tokenized with the model tokenizer and then passed into the model for generating the summary of the new text. The generated summary of the model is then decoded back to text.

### F. Generating reference summary with GPT 3.5 model

For assessing the quality of the model-generated summary, it has to be compared with a reference summary. So the reference summary was generated with the GPT 3.5 model of openAI. The GPT 3.5 model was prompted to generate a brief summary within ten words for the given long text.

### G. Loss Function

We will use the Cross-Entropy Loss for Bangla text summarization with a few examples. For each token position, the difference between the generated and reference summary is measured by the loss function. This dissimilarity should be minimized in order to produce better text summaries. The Cross-Entropy loss function is given below:

$$\text{Loss}(y_t, \hat{y}_t) = -\sum_{i=1}^{V} y_{t,i} \log(\hat{y}_{t,i}) \tag{1}$$

where:

- $y_t$ is the true one-hot encoded vector for the token at position $t$ in the target sequence.
- $\hat{y}_t$ is the predicted probability distribution over the vocabulary for the token at position $t$.
- $V$ is the size of the vocabulary.

### H. Evaluation

The generated summary is compared with the reference summary and evaluated on the BLUE (Bilingual Evaluation Understudy) and ROGUE (Recall-Oriented Understudy for Gisting Evaluation) scores. To measure the similarity between the generated summary and a reference summary, the BLEU and ROGUE scores are calculated. This requires that the n-gram precision be computed after tokenizing both summaries, with the brevity penalty taken into account in shorter generated summaries. In our case, we generated the reference summary with the help of the GPT 3.5 model and then used it to compare it with the generated one.

## IV. Result Analysis

In this section, we will analyze the results we obtained from the experiments we did for evaluating the performance of our Bangla text summarization model through Few-Shot learning. The analysis is based on BLUE and ROGUE scores we got and for assessing the model's effectiveness, compares the model-generated summary with the reference one which is the summary achieved with the GPT 3.5 model. In addition, we compared our results with other summarization techniques for a comprehensive understanding of our approach.

ROUGE is a set of metrics used to evaluate the quality of summaries or translations by comparing them to reference summaries or translations. It measures the overlap of n-grams (contiguous sequences of n items, typically words) between the generated summary and the reference summaries. For ROUGE-1 a score around 0.5 is good and below 0.4 is low. A score above 0.4 is good, and 0.2 to 0.4 is moderate for ROUGE-2. A good ROUGE-L score is around 0.4 and low at 0.3 to 0.4. In our case, the ROGUE values obtained are ROUGE-1: 0.50, ROUGE-2: 0.27 ROUGE-L: 0.44. The ROUGE scores suggest a reasonable overlap between the reference and generated summary. The ROUGE-1 score of 0.50 is equivalent to a situation where half of the single words contained in the generated summary are the same as those in the reference summary. The bigram metrics show that agreement on word sequences is okay with a ROUGE-2 score of 0.27. The model captures the most important aspects of the sentence if the ROUGE-L score is 0.44.

BLEU is a metric for evaluating the quality of machine-translated text by comparing it to one or more reference translations. It measures the precision of n-grams (contiguous sequences of n items, typically words) in the generated translation compared to the reference translations. BLEU computes a precision score for each n-gram up to a certain length (usually up to 4 grams) and combines them using a weighted geometric mean. Scores of BLEU range from ) to 1 and the higher the score the better the quality. The BLEU score we achieved in our project was 0.18. A BLEU score of 0.18 shows that there is some replication in terms of word choice and arrangement but not in the phrasing or structure of the generated summary.

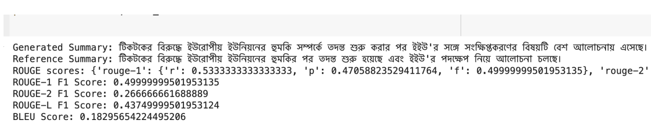
Generated Summary: টিকটকের বিরুদ্ধে ইউরোপীয় ইউনিয়নের হুমকি সম্পর্কে তদন্ত শুরু করার পর ইইউ'র সঙ্গে সংক্ষিপ্তকরণের বিষয়টি বেশ আলোচনায় এসেছে।
Reference Summary: টিকটকের বিরুদ্ধে ইউরোপীয় ইউনিয়নের হুমকির পর তদন্ত শুরু হয়েছে এবং ইইউ'র পদক্ষেপ নিয়ে আলোচনা চলছে।
ROUGE scores: {'rouge-1': {'r': 0.5333333333333333, 'p': 0.47058823529411764, 'f': 0.49999999501953135}, 'rouge-2'
ROUGE-1 F1 Score: 0.49999999501953135
ROUGE-2 F1 Score: 0.266666661688889
ROUGE-L F1 Score: 0.43749999501953124
BLEU Score: 0.18295654224495206

Fig. 4: A Sample of predicted and reference summary and their BLEU and ROGUE scores

TABLE I: Results Achieved

| BLEU Score | ROUGE-1 | ROUGE-2 | ROUGE-L |
|---|---|---|---|
| 0.18 | 0.50 | 0.27 | 0.44 |

## V. Discussion

Our finding's implications and their strengths and limitations will be examined in this section along with any potential issues. It has been observed from our experiment that prompt engineering with the help of a pre-trained transformer model can effectively generate summaries that are high-quality for a resource-constrained language like Bangla. The ROGUE and BLUE scores that we obtained from our experiment indicate that there has been a significant overlap between the generated and reference summary in terms of context and structure. This is an indication that Few-shot learning with prompts that are created carefully could be a promising answer to low-resource language text summarization for a language such as Bangla.

### A. Strengths of Our Approch

The ability to perform well with minimal data is one of the main strengths of our experiment. A large annotated dataset is required for traditional methods that are mostly unavailable for low-resource languages. Our approach eliminates this problem by leveraging the pre-trained LLM model and only a few examples to guide the model for summarizing the new text. This approach also takes less training time making the whole process more effective. This means that by simply changing the prompt structure as well as using different examples, one can adjust their model to make either short or long summary styles appealing. This is essential as it makes the model adaptable to different summarization requirements.

### B. Limitations

The model heavily depends on the relevance and quality of the provided prompt in order to summarize effectively. While the model shows good performance in general text summarization, it might face difficulties in handling intricate or technical texts that need a more profound understanding of their content. This limitation implies that one may need more fine-tuning or include domain-specific knowledge for particular summarization tasks. Despite ROGUE and BLEU being a quantitative measure of the quality of the summary generated, the coherence and readability is not always well captured by these scores. Incorporation of human assessment could help improve this.

## VI. Conclusion

Even though sufficient work has been done in prompt engineering using the English language, very few attempts have been made in prompt engineering using the Bengali language, due to Bangla being a low-resource language. Hence, we have decided to make a contribution to this field. Some future work of our project would be, using a more diverse set of examples as prompts in the Bengali language in order to get more accurate results and better performance, experimenting with prompt engineering using the Bengali language on a variety of other multilingual models, and comparing the results. And also, to find out and experiment with other suitable evaluation metrics to evaluate the output. Furthermore, we hope to conduct state-of-the-art multi-dimensional evaluation across multiple dimensions such as fluency, coherence, and actuality.

## REFERENCES


[1] M. S. Islam, "Research on bangla language processing in bangladesh: progress and challenges," in *8th international language & development conference*, 2009, pp. 23–25.

[2] R. R. Chowdhury, M. T. Nayeem, T. T. Mim, M. S. R. Chowdhury, and T. Jannat, "Unsupervised abstractive summarization of bengali text documents," *arXiv preprint arXiv:2102.04490*, 2021.

[3] N. Dhar, G. Saha, P. Bhattacharjee, A. Mallick, and M. S. Islam, "Pointer over attention: An improved bangla text summarization approach using hybrid pointer generator network," in *2021 24th International Conference on Computer and Information Technology (ICCIT)*. IEEE, 2021, pp. 1–5.

[4] S. Abujar, A. K. M. Masum, M. Mohibullah, S. A. Hossain *et al.*, "An approach for bengali text summarization using word2vector," in *2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*. IEEE, 2019, pp. 1–5.

[5] M. A. I. Talukder, S. Abujar, A. K. M. Masum, F. Faisal, and S. A. Hossain, "Bengali abstractive text summarization using sequence to sequence rnns," in *2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*. IEEE, 2019, pp. 1–5.

[6] A. Parnami and M. Lee, "Learning from few examples: A summary of approaches to few-shot learning," *arXiv preprint arXiv:2203.04291*, 2022.

[7] J. Wang, E. Shi, S. Yu, Z. Wu, C. Ma, H. Dai, Q. Yang, Y. Kang, J. Wu, H. Hu *et al.*, "Prompt engineering for healthcare: Methodologies and applications," *arXiv preprint arXiv:2304.14670*, 2023.

[8] D. van Zandvoort, L. Wiersema, T. Huibers, S. van Dulmen, and S. Brinkkemper, "Enhancing summarization performance through transformer-based prompt engineering in automated medical reporting," *arXiv preprint arXiv:2311.13274*, 2023.

[9] S. Jain, V. Keshava, S. M. Sathyendra, P. Fernandes, P. Liu, G. Neubig, and C. Zhou, "Multi-dimensional evaluation of text summarization with in-context learning," *arXiv preprint arXiv:2306.01200*, 2023.

[10] L. Reynolds and K. McDonell, "Prompt programming for large language models: Beyond the few-shot paradigm," in *Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems*, 2021, pp. 1–7.

[11] S. Arora, A. Narayan, M. F. Chen, L. Orr, N. Guha, K. Bhatia, I. Chami, and C. Re, "Ask me anything: A simple strategy for prompting language models," in *The Eleventh International Conference on Learning Representations*, 2022.

[12] T. Gao, A. Fisch, and D. Chen, "Making pre-trained language models better few-shot learners," *arXiv preprint arXiv:2012.15723*, 2020.

[13] O. Fagbohun, R. M. Harrison, and A. Dereventsov, "An empirical categorization of prompting techniques for large language models: A practitioner's guide," *arXiv preprint arXiv:2402.14837*, 2024.

[14] N. Nashid, M. Sintaha, and A. Mesbah, "Retrieval-based prompt selection for code-related few-shot learning," in *2023 IEEE/ACM 45th International Conference on Software Engineering (ICSE)*. IEEE, 2023, pp. 2450–2462.

[15] Z. Luo, Q. Xie, and S. Ananiadou, "Chatgpt as a factual inconsistency evaluator for abstractive text summarization," *arXiv preprint arXiv:2303.15621*, 2023.

[16] T. Ahmed, K. S. Pai, P. Devanbu, and E. Barr, "Automatic semantic augmentation of language model prompts (for code summarization)," in *Proceedings of the IEEE/ACM 46th International Conference on Software Engineering*, 2024, pp. 1–13.

[17] L. Xue, N. Constant, A. Roberts, M. Kale, R. Al-Rfou, A. Siddhant, A. Barua, and C. Raffel, "mt5: A massively multilingual pre-trained text-to-text transformer," *arXiv preprint arXiv:2010.11934*, 2020.

[18] M. Kowsher, A. A. Sami, N. J. Prottasha, M. S. Arefin, P. K. Dhar, and T. Koshiba, "Bangla-bert: transformer-based efficient model for transfer learning and language understanding," *IEEE Access*, vol. 10, pp. 91 855–91 870, 2022.

[19] L. Basyal and M. Sanghvi, "Text summarization using large language models: A comparative study of mpt-7b-instruct, falcon-7b-instruct, and openai chat-gpt models," *arXiv preprint arXiv:2310.10449*, 2023.

[20] G. Shahariar, T. Talukder, R. A. K. Sotez, and M. T. R. Shawon, "Rank your summaries: Enhancing bengali text summarization via ranking-based approach," in *International Conference on Big Data, IoT and Machine Learning*. Springer, 2023, pp. 153–167.

[21] M. Borah, P. Dadure, P. Pakray *et al.*, "Comparative analysis of t5 model for abstractive text summarization on different datasets," 2022.