Empowering Low-Resource Languages: TraSe Architecture for Enhanced Retrieval-Augmented Generation in Bangla

Atia Shahnaz Ipa[a], **Mohammad Abu Tareq Rony**[b*], **Mohammad Shariful Islam**[c]
[a]Department of Mechatronics Engineering, Khulna University
of Engineering & Technology, Khulna, Bangladesh.
[b*]Department of Statistics, Noakhali Science & Technology University, Bangladesh
[c]Department of Computer Science & Telecommunication Engineering,
Noakhali Science & Technology University, Bangladesh
atia.s.ipa@gmail.com, abutareqrony@gmail.com, shariful.cse43@gmail.com

Abstract

Research on Retrieval-Augmented Generation for low-resource languages has been sparse because of limited resources. To address this, we focus on Bangla, a low-resource language, and have created a dataset of 200 question-answer pairs as a basis for our study from Bangla Wikipedia dump data. This paper introduces the TraSe architecture, which enhances RAG for Bangla using Translative prompting. Our experiments demonstrate that TraSe improves answer selection accuracy, achieving 34% with automatic retrieval and 63% with Human-in-the-Loop retrieval, outperforming baseline methods. The TraSe architecture marks a significant advancement in RAG for low-resource languages and has the potential to enhance question-answering systems for Bangla and similar languages. Future research could explore additional low-resource languages. The code is available at the following GitHub repository: `https://github.com/Atia6/TraSe-Bangla-RAG`.

1 Introduction

The rapid advancements in natural language processing (NLP) have led to the development of sophisticated models that can perform a wide range of tasks with high accuracy(Bird, 2024). Among these, Retrieval-Augmented Generation (RAG) has emerged as a powerful approach that combines the strengths of information retrieval and generative models to produce more informed and contextually accurate responses. While RAG has been extensively explored in languages like English, its application in low-resource languages, such as Bangla, remains significantly underdeveloped(Cuconasu et al., 2024).

The scarcity of research and resources in Bangla RAG presents a critical gap in the NLP field, particularly given the language's extensive use by over 230 million speakers worldwide(Bhattacharjee et al., 2022a). Existing systems struggle to meet the nuanced demands of Bangla language processing, often unable to retrieve (Rony et al., 2024) and generate contextually relevant information effectively (Ipa et al., 2024). This gap not only limits the practical applications of NLP in Bangla but also highlights the need for tailored architectures to address the unique challenges posed by this language.

In response to this need, we propose the TraSe architecture, a novel approach specifically designed for the RAG in Bangla. TraSe integrates advanced retrieval mechanisms with generative capabilities, optimizing performance across various tasks by leveraging both pre-existing knowledge and contextual information. This paper presents a detailed examination of TraSe's architecture, its comparative performance against existing systems, and its potential to enhance Bangla language processing. Through this research, we aim to contribute a significant step forward in the development of effective NLP tools for Bangla, bridging the gap in RAG research for this important language.

1.1 Main Contributions

We achieved significant advancements in RAG for the low-resource Bangla language through the Translative method and further enhanced performance using the TraSe method. Our main contributions are as follows:

1. Created a Bangla question-answering dataset consisting of 200 question-answer pairs.

2. Introduced the Translative prompting method specifically designed for Bangla

question answering.

3. Developed the TraSe architecture and demonstrated its superior performance compared to baseline prompting methods.

2 Related Work

Retrieval-Augmented Generation (RAG) has emerged as a powerful framework for addressing key limitations of large language models (LLMs), such as hallucination, outdated knowledge, and lack of transparency (Gao et al., 2023; Huang and Huang, 2024). By integrating external knowledge into the generation process, RAG enhances accuracy, reliability, and contextual relevance (Zhao et al., 2024). Over time, the paradigm has evolved from simple retrieval-based augmentation to more sophisticated modular architectures that optimize retrieval, generation, and augmentation processes (Gao et al., 2023). A notable advancement in this direction is FLARE, an active retrieval mechanism that continuously gathers relevant information throughout the generation process to improve response quality (Jiang et al., 2023). Beyond traditional text-based applications, RAG has demonstrated versatility across multimodal tasks and knowledge-intensive scenarios, reinforcing its potential in various domains (Zhao et al., 2024).

Despite these advancements, RAG still faces challenges in evaluation, retrieval quality, and real-world implementation. Researchers are actively working to develop comprehensive benchmarks and refine methodologies to improve retrieval accuracy, optimize integration with LLMs, and enhance system adaptability (Zhao et al., 2024; Huang and Huang, 2024). Several recent innovations have focused on addressing these limitations. Corrective RAG, introduced by (Yan et al., 2024), incorporates a retrieval evaluator to assess document quality and dynamically trigger different retrieval actions, such as web searches, thereby improving the reliability of retrieved content. SelfMem (Cheng et al., 2023) takes a different approach by iteratively using a retrieval-augmented generator to build an unbounded memory pool, leveraging past model outputs as a self-referential knowledge base. Meanwhile, Iter-RetGen (Shao et al., 2023) adopts an iterative retrieval-generation cycle where model-generated content informs subsequent retrieval steps, refining relevance and coherence. These methods specifically address issues related to retrieval precision, fixed corpus constraints, and complex information needs, demonstrating improved performance across various NLP tasks, including question answering, summarization, and dialogue generation.

Further developments continue to push the boundaries of RAG optimization. Stochastic RAG (Zamani and Bendersky, 2024) introduces an end-to-end optimization framework that utilizes straight-through Gumbel-top-k selection, enhancing retrieval and generation efficiency while achieving state-of-the-art results across multiple tasks. Blended RAG (Sawarkar et al., 2024) improves retrieval effectiveness by leveraging hybrid query strategies and semantic search, surpassing conventional fine-tuning approaches on datasets like SQuAD. Additionally, Graph Retrieval-Augmented Generation (GRAG) (Hu et al., 2024) presents a divide-and-conquer strategy for retrieving structured textual subgraphs, facilitating multi-hop reasoning and significantly outperforming standard RAG models in handling networked document structures.

Beyond these techniques, other research efforts have sought to refine RAG's adaptability and evaluation. R^2AG (Ye et al., 2024) aims to bridge the semantic gap between retrievers and LLMs by embedding retrieval information directly into the generation process. RAGAs (Shahul et al., 2023) introduces a reference-free evaluation framework to assess retrieval relevance, LLM faithfulness, and overall generation quality, providing a more holistic assessment of RAG pipelines. The RAGGED framework (Hsia et al., 2024) analyzes different RAG configurations, revealing that optimal performance depends on varying model architectures and context utilization strategies. Additionally, MemoRAG (Qian et al., 2024) pioneers a memory-augmented approach that employs a dual-system architecture—where a lightweight LLM manages global memory, while a more expressive LLM handles final answer generation—enabling better handling of ambiguous queries and long-term knowledge retention.

Together, these advancements illustrate the increasing sophistication of RAG techniques and their transformative potential for LLMs. By improving retrieval strategies, optimizing generative integration, and expanding to new application areas, RAG continues to evolve as a fundamental enabler of more accurate, contextually aware, and reliable AI-generated content.

3 Methodology

In this study, we developed TraSe architecture, a selection-based process to improve the performance of RAG for Bangla question answering with the help of the translative method. We further compared the performance of our model with existing techniques.

3.1 Dataset

We created 200 questions from the Bangla Wikipedia dump for our experiment. The raw Bangla dataset that we utilized, consisted of 27 topics in 27 articles. The dataset is preprocessed to convert to chunks of 5 sentences. Along with 200 questions, 3 related contexts are accompanied by each question for Human In the Loop (HIL) context insertion in the LLM. Dataset details are given in Table 1. In Table 2, several question-answer pairs along with their corresponding answer types are presented.

3.2 Baselines

The baseline methods for comparison are described below.

Zero Shot: The zero-shot method involves assigning a task to a model without prior examples or specific training, relying solely on the model's pre-existing knowledge. This approach is useful for generalization in low-data scenarios. (Arora et al., 2023) explored the use of zero-shot retrieval in their work.

2 Shot: The two-shot method provides the model with two examples before a new task, helping it better understand the task structure and improve performance. (Brown et al., 2020) explored the few-shot technique in their work on GPT-3.

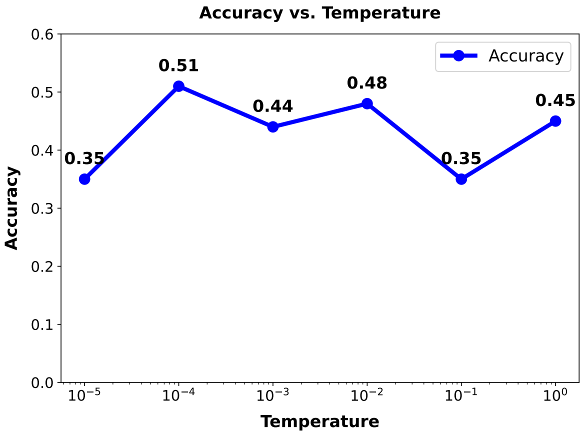

Figure 1: Temperature vs accuracy for the zero-shot method with HIL context.

Self-Ask: Self-Ask encourages the model to ask clarifying questions before answering, breaking down complex queries for more accurate responses. (Press et al., 2023) discussed this method in their study.

ReAct: ReAct (Reasoning and Acting) alternates between reasoning and action steps, allowing the model to iteratively refine its understanding and outputs, which is particularly useful in complex tasks. This method was introduced by (Yao et al., 2023).

3.3 LLM Parameter

For this experiment, we used the Llama 2 7B model, which supports over 260 languages, in a text generation pipeline via the transformers[1] library. The model, optimized with bfloat16 data and automatic device mapping, generates sequences of up to 3000 tokens. Sampling with a 'top_k' of 10 promotes diverse yet coherent outputs. Zero-shot direct prompting and HIL context were applied as shown in Figure 1, and after testing temperatures from 0.00001 to 1, the most accurate results were achieved at a temperature of 0.0001, which was selected for the final setup. In this research, we used LangChain[2] to integrate the Hugging Face pipeline, allowing us to efficiently apply prompting techniques with pretrained models.

[1] https://pypi.org/project/transformers/

[2] https://www.langchain.com/

Table 1: Dataset description

Dataset	No of Articles	No. of Words	No. of Chunks	Question Answer Pair	Text Based Answer	Number Based Answer
Bangla Wikipedia Dump	27	53,575	710	200	70	130

Table 2: Question-answer pairs with answer type

Question	Answer	Answer Type
ঢাকা শহর কতটি সংসদীয় এলাকায় বিভক্ত? *(How many parliamentary constituencies is Dhaka city divided into?)*	২৫ টি *(25)*	Number-based
সচিবালয় কোথায় অবস্থিত? *(Where is the Secretariat located?)*	রমনায় *(In Ramna)*	Text-based
জাতীয় সংসদ ভবনের স্থপতি কে ছিলেন? *(Who was the architect of the National Parliament Building?)*	লুইস কান *(Louis Kahn)*	Text-based
বাংলাদেশের জাতীয় সংসদ ভবন কয় কক্ষবিশিষ্ট? *(How many chambers does the National Parliament Building of Bangladesh have?)*	এক কক্ষ *(Single chamber)*	Text-based
বাংলাদেশের জাতীয় মসজিদ কোনটি? *(What is the national mosque of Bangladesh?)*	বায়তুল মুকাররম *(Baitul Mukarram)*	Text-based
ঢাকায় প্রতিবছর কত টন কঠিন বর্জ্য উৎপন্ন হয়? *(How many tons of solid waste are generated in Dhaka each year?)*	৯৭ লক্ষ টন *(9.7 million tons)*	Number-based
বাংলাদেশের প্রধান বাণিজ্যিক কেন্দ কোনটি? *(What is the main commercial hub of Bangladesh?)*	ঢাকা *(Dhaka)*	Text-based

3.4 Translative Prompting

Llama 2 has not been trained on a large amount of Bangla data. Therefore, its performance is not that great in the case of Bangla. The translative method instructs the model to translate the query and context to English, then find the answer, and then translate the answer to Bangla as depicted in Figure 2. This method has been seen to be useful for text-based answers in this study.

3.5 TraSe Architecture

The TraSe architecture can be seen in Figure 3. BanglaBERT (Bhattacharjee et al., 2022b) and bert base multilingual case (Devlin et al., 2018) embedding models have been used to embed query and document. Cosine similarity is used to retrieve the top 3 contexts. We have also used accurate 3 contexts along with the query for HIL context to evaluate the performance of the model when the retrieval process is accurate.

As Translative prompting is more useful for text-based answers than the others, a selective model has been proposed. In the model, query, contexts, answers generated from Translative prompting, and answers generated from one of the other methods (zero-shot, 2-shot, Self Ask, and ReAct) are inserted into the LLM pipeline and asked to select one of the answers based on the query and context.

3.6 Evaluation Metrics

Accuracy: Accuracy is the percentage of correct answers. The generated answers were manually evaluated and assigned as right or wrong answers. Based on manual evaluation the accuracy has been determined. We have taken an answer to be accurate if the information is correct, whether it is answered

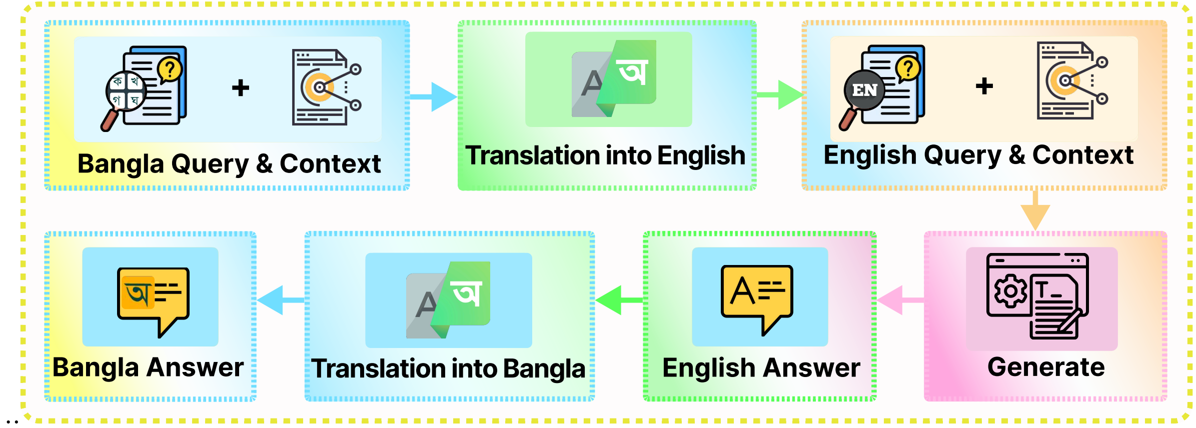

Figure 2: Flowchart of Translative method.

in Bangla or English. In the equation, TP means true positives (correct positives), TN means true negatives (correct negatives), FP means false positives (incorrect positives), and FN means false negatives (incorrect negatives). The formula for accuracy is given by:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

F1 Score: The F1 score is the harmonic mean of precision and recall, making it a more reliable metric than accuracy when dealing with imbalanced datasets. The formula for F1 Score is:

$$F1 \text{ Score} = 2 \cdot \frac{\text{Recall} \cdot \text{Precision}}{\text{Precision} + \text{Recall}} \tag{2}$$

Exact Match is an important evaluation metric for question answering. However, in our case, it is not useful as the generated answer is not always in Bangla. One example is given below.

Query: 'রাষ্ট্রপতি এরশাদ কত খিস্টাব্দ পর্যন্ত দেশ শাসন করেন?' *Until when President Ershad ruled the country?*

Actual Answer: ১৯৯১ খ্রিস্টাব্দ *1999 AC*

Generated Answer: The answer to the query is 1991.

So, the generated answer is correct but not an exact match with the actual answer.

4 Result and Discussion

The efficiency of the translative method for text-based question answering is evident in Figure 4. With an accuracy of 0.28 for BanglaBERT, 0.24 for Bert-base-multilingual-case, and 0.61 for the HIL context, this method consistently outperforms the other four methods for text-based answering. Additionally, the translative method demonstrates competitive accuracy in number-based answers.

Table 3 presents the f1 scores and accuracy for various models, including baseline methods and the Translative prompting technique, with and without retrieval using BanglaBERT embeddings, Bert-base-multilingual-case embeddings, and Human-in-the-Loop (HIL) retrieval. The results show that the Translative model generally outperforms baseline models across different retrieval methods. Notably, all TraSe models demonstrate significant improvements over the baselines. For instance, the combination of zero-shot and Translative prompting achieves a 33% accuracy with Bert-base-multilingual-case, a substantial improvement over the 22% accuracy of the baseline 0-shot direct method. Similarly, in the HIL retrieval context, the TraSe method with zero-shot and Translative prompting achieves a 63% accuracy, compared to 51% for the baseline, indicating a notable improvement. Additionally, the 2-shot Translative combination is competitive with the zero-shot Translative method for BanglaBERT embeddings, achieving a 34% accuracy compared to 33%. Overall, when retrieval is accurate, the combination of zero-shot and Translative prompting with the TraSe architecture consistently achieves higher accuracy, with up to 63% in the HIL retrieval setting, showcasing the effectiveness of the TraSe approach.

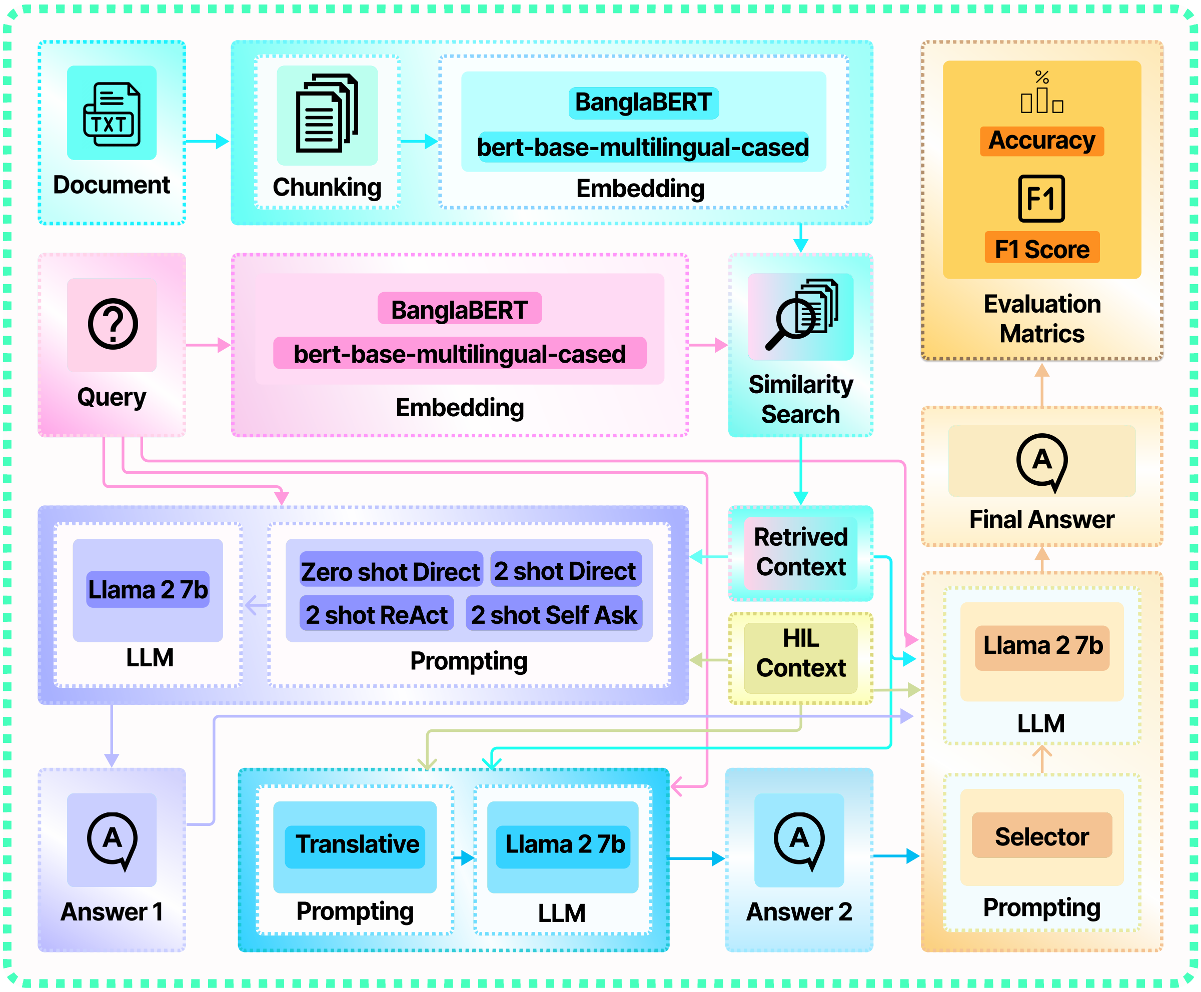

Figure 3: Flowchart of TraSe method.

5 Conclusion

In this study, we introduced the Translative prompting model, which demonstrated strong performance in both number-based and text-based answers for Bangla RAG. Building on this, we developed the TraSe model, leveraging the strengths of Translative prompting to enhance answer selection from previously generated responses. The TraSe model achieved notable accuracy improvements, reaching 34% accuracy with automatic retrieval and 63% accuracy with Human-in-the-Loop (HIL) retrieval, underscoring its effectiveness in both automated and human-assisted retrieval contexts.

Future research should prioritize incorporating a variety of language models, larger and more diverse datasets, and an expanded set of low-resource languages to validate and build upon these findings, ultimately contributing to a deeper and more generalizable understanding of language model performance.

Limitations

A limitation of this study is that it utilizes a single language model, which may not capture the full spectrum of performance across different models. Additionally, the smaller sample size may affect the generalizability of the results. Future research could benefit from incorporating a variety of models and larger datasets to validate and extend these findings. Furthermore, investigating other low-resource languages could provide additional insights and enhance the robustness of the conclusions. Investigating additional languages would not only enhance the robustness of the conclusions but also provide a more comprehensive understanding of how language models perform in diverse linguistic contexts.

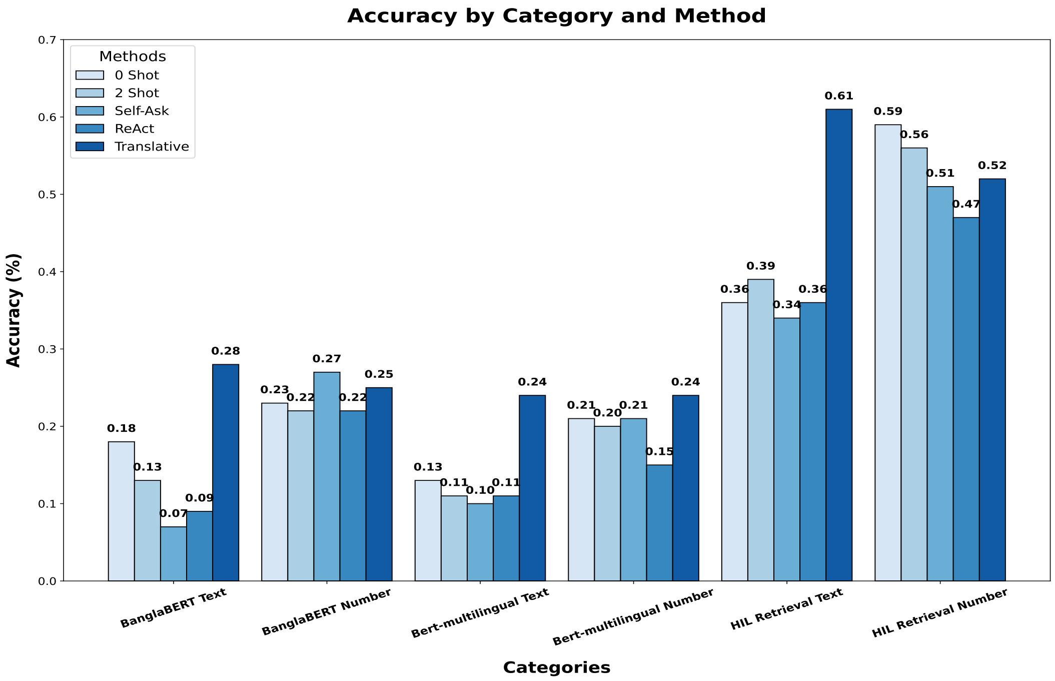

Figure 4: Accuracy of text-based and number-based answers

Table 3: Performance comparison between methods with and without retrieval across different models.

Method	**Without Retrieval**		**With Retrieval**					
			BanglaBERT		Bert-base-multilingual-case		Human In the loop Retrieval	
	F1 Score	Accuracy	F1 Score	Accuracy	F1 Score	Accuracy	F1 Score	Accuracy
0 shot direct	.06	.03	.36	.22	.31	.18	.68	.51
2 shot direct	.13	.07	.32	.19	.28	.16	.67	.50
Self-Ask	-	-	.33	.20	.29	.17	.62	.45
ReAct	-	-	.29	.17	.25	.14	.60	.43
Translative	-	-	.41	.26	.39	.24	.71	.55
	TraSe Method							
0shot+ Translative	-	-	.50	.33	.45	.29	.77	.63
2shot+ Translative	-	-	.51	.34	.41	.26	.75	.60
SelfAsk+ Translative	-	-	.46	.30	.43	.27	.76	.61
ReAct + Translative	-	-	.45	.29	.36	.22	.74	.59

References

Daman Arora, Anush Kini, Sayak Ray Chowdhury, Nagarajan Natarajan, Gaurav Sinha, and Amit Sharma. 2023. Gar-meets-rag paradigm for zero-shot information retrieval. *ArXiv*, abs/2310.20158.

Abhik Bhattacharjee, Tahmid Hasan, Wasi Ahmad, Kazi Samin Mubasshir, Md Saiful Islam, Anindya Iqbal, M. Sohel Rahman, and Rifat Shahriyar. 2022a. BanglaBERT: Language model pretraining and benchmarks for low-resource language understanding evaluation in Bangla. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 1318–1327, Seattle, United States. Association for Computational Linguistics.

Abhik Bhattacharjee, Tahmid Hasan, Wasi Ahmad, Kazi Samin Mubasshir, Md Saiful Islam, Anindya Iqbal, M. Sohel Rahman, and Rifat Shahriyar. 2022b. BanglaBERT: Language model pretraining and benchmarks for low-resource language understanding evaluation in Bangla. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 1318–1327, Seattle, United States. Association for Computational Linguistics.

Steven Bird. 2024. Must nlp be extractive? In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14915–14929.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners.

Xin Cheng, Di Luo, Xiuying Chen, Lemao Liu, Dongyan Zhao, and Rui Yan. 2023. Lift yourself up: Retrieval-augmented text generation with self memory. *ArXiv*, abs/2305.02437.

Florin Cuconasu, Giovanni Trappolini, Federico Siciliano, Simone Filice, Cesare Campagnano, Yoelle Maarek, Nicola Tonellotto, and Fabrizio Silvestri. 2024. The power of noise: Redefining retrieval for rag systems. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 719–729.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, Qianyu Guo, Meng Wang, and Haofen Wang. 2023. Retrieval-augmented generation for large language models: A survey. *ArXiv*, abs/2312.10997.

Jennifer Hsia, Afreen Shaikh, Zhiruo Wang, and Graham Neubig. 2024. Ragged: Towards informed design of retrieval augmented generation systems. *ArXiv*, abs/2403.09040.

Yuntong Hu, Zhihan Lei, Zhengwu Zhang, Bo Pan, Chen Ling, and Liang Zhao. 2024. Grag: Graph retrieval-augmented generation. *ArXiv*, abs/2405.16506.

Yizheng Huang and Xiangji Huang. 2024. A survey on retrieval-augmented text generation for large language models. *ArXiv*, abs/2404.10981.

Atia Shahnaz Ipa, Priyo Nath Roy, Mohammad Abu Tareq Rony, Ali Raza, Norma Latif Fitriyani, Yeonghyeon Gu, and Muhammad Syafrudin. 2024. Bdsentillm: A novel llm approach to sentiment analysis of product reviews. *IEEE Access*.

Zhengbao Jiang, Frank F. Xu, Luyu Gao, Zhiqing Sun, Qian Liu, Jane Dwivedi-Yu, Yiming Yang, Jamie Callan, and Graham Neubig. 2023. Active retrieval augmented generation. *ArXiv*, abs/2305.06983.

Ofir Press, Muru Zhang, Sewon Min, Ludwig Schmidt, Noah A. Smith, and Mike Lewis. 2023. Measuring and narrowing the compositionality gap in language models.

Hongjin Qian, Peitian Zhang, Zheng Liu, Kelong Mao, and Zhicheng Dou. 2024. Memorag: Moving towards next-gen rag via memory-inspired knowledge discovery. *ArXiv*, abs/2409.05591.

Mohammad Abu Tareq Rony, Mohammad Shariful Islam, Tipu Sultan, Samah Alshathri, and Walid El-Shafai. 2024. Medigpt: Exploring potentials of conventional and large language models on medical data. *IEEE Access*.

Kunal Sawarkar, Abhilasha Mangal, and Shivam Raj Solanki. 2024. Blended rag: Improving rag (retriever-augmented generation) accuracy with semantic search and hybrid query-based retrievers. *2024 IEEE 7th International Conference on Multimedia Information Processing and Retrieval (MIPR)*, pages 155–161.

ES Shahul, Jithin James, Luis Espinosa Anke, and Steven Schockaert. 2023. Ragas: Automated evaluation of retrieval augmented generation. In *Conference of the European Chapter of the Association for Computational Linguistics*.

Zhihong Shao, Yeyun Gong, Yelong Shen, Minlie Huang, Nan Duan, and Weizhu Chen. 2023. Enhancing retrieval-augmented large language models with iterative retrieval-generation synergy.

Shi-Qi Yan, Jia-Chen Gu, Yun Zhu, and Zhen-Hua Ling. 2024. Corrective retrieval augmented generation.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. 2023. React: Synergizing reasoning and acting in language models.

Fuda Ye, Shuangyin Li, Yongqi Zhang, and Lei Chen. 2024. R2ag: Incorporating retrieval information into retrieval augmented generation. In *Conference on Empirical Methods in Natural Language Processing*.

Hamed Zamani and Michael Bendersky. 2024. Stochastic rag: End-to-end retrieval-augmented generation through expected utility maximization. *ArXiv*, abs/2405.02816.

Penghao Zhao, Hailin Zhang, Qinhan Yu, Zhengren Wang, Yunteng Geng, Fangcheng Fu, Ling Yang, Wentao Zhang, and Bin Cui. 2024. Retrieval-augmented generation for ai-generated content: A survey. *ArXiv*, abs/2402.19473.