Enhancing Retrieval-Augmented Generation: A Study of Best Practices

Siran Li **Linus Stenzel** **Carsten Eickhoff** **Seyed Ali Bahrainian**
University of Tübingen
siran.li@uni-tuebingen.de, stenzel@student.uni-tuebingen.de,
{carsten.eickhoff, seyed.ali.bahreinian}@uni-tuebingen.de

Abstract

Retrieval-Augmented Generation (RAG) systems have recently shown remarkable advancements by integrating retrieval mechanisms into language models, enhancing their ability to produce more accurate and contextually relevant responses. However, the influence of various components and configurations within RAG systems remains underexplored. A comprehensive understanding of these elements is essential for tailoring RAG systems to complex retrieval tasks and ensuring optimal performance across diverse applications. In this paper, we develop several advanced RAG system designs that incorporate query expansion, various novel retrieval strategies, and a novel Contrastive In-Context Learning RAG. Our study systematically investigates key factors, including language model size, prompt design, document chunk size, knowledge base size, retrieval stride, query expansion techniques, Contrastive In-Context Learning knowledge bases, multilingual knowledge bases, and Focus Mode retrieving relevant context at sentence-level. Through extensive experimentation, we provide a detailed analysis of how these factors influence response quality. Our findings offer actionable insights for developing RAG systems, striking a balance between contextual richness and retrieval-generation efficiency, thereby paving the way for more adaptable and high-performing RAG frameworks in diverse real-world scenarios. Our code and implementation details are publicly available [1].

1 Introduction

Language Models (LMs) such as GPT, BERT, and T5 have demonstrated remarkable versatility, excelling in a wide range of NLP tasks, including summarization (Bahrainian et al., 2022), extracting relevant information from lengthy documents, question-answering, and storytelling (Brown et al., 2020b; Devlin et al., 2019; Raffel et al., 2020). However, their static knowledge and opaque reasoning raise concerns about maintaining factual accuracy and reliability as language and knowledge evolve (Huang et al., 2024; Jin et al., 2024). As new events emerge, and scientific advancements are made, it becomes crucial to keep models aligned with current information (Shi et al., 2024a). However, continuously updating models is both costly and inefficient. To address this, RAG models have been proposed as a more efficient alternative, integrating external knowledge sources during inference to provide up-to-date and accurate information (Lewis et al., 2020; Borgeaud et al., 2022; Lee et al., 2024). RAG models augment language models by incorporating verifiable information, improving factual accuracy in their responses (Gao et al., 2023; Kim et al., 2023). This approach not only mitigates some conceptual limitations of traditional LMs but also unlocks practical, real-world applications. By integrating a domain-specific knowledge base, RAG models transform LMs into specialized experts, enabling the development of highly targeted applications and shifting them from generalists to informed specialists (Siriwardhana et al., 2023). In recent years, this advancement has led to many proposed architectures and settings for an optimal RAG model (Li et al., 2024; Dong et al., 2024). However, the best practices for designing RAG models are still not well understood.

In this paper, we comprehensively examine the efficacy of RAG in enhancing Large LM (LLM) responses, addressing nine key research questions: (1) How does the size of the LLM affect the response quality in an RAG system? (2) Can subtle differences in prompt significantly affect the alignment of retrieval and generation? (3) How does the retrieved document chunk size impact the response quality? (4) How does the size of the knowledge base impact the overall performance? (5) In the retrieval strides (Ram et al., 2023), how

[1] https://github.com/ali-bahrainian/RAG_best_practices

often should context documents be updated to optimize accuracy? (6) Does expanding the query improve the model's precision? (7) How does including Contrastive In-context Learning demonstration examples influence RAG generation? (8) Does incorporating multilingual documents affect the RAG system's responses? (9) Does focusing on a few retrieved sentences sharpen RAG's responses? To address these questions, we employ ablation studies as the primary method, allowing for a detailed empirical investigation of RAG's operational mechanisms. A custom evaluation framework is developed to assess the impact of various RAG components and configurations individually. The insights gained will contribute to advancing LLM performance and inform future theoretical developments.

The Main Contributions of this paper are: (1) We conduct an extensive benchmark to help explain the best practices in RAG setups. **(2)** While the first five research questions above are based on previous literature, the methods that address the last four research questions, namely, Query Expansion, Contrastive In-context Learning demonstration, multilingual knowledge base, and Focus Mode RAG are novel contributions of this study which we believe will advance the field.

The remainder of this paper is organized as follows: Section 2 provides an overview of important related work. Section 3 presents novel methods that improve RAG responses and outlines the methodology. Section 4 presents two evaluation datasets, knowledge base, and evaluation metrics and explains the implementation details. Section 5 discusses the extensive results of our carefully designed benchmark comparison and Section 6 highlights the key findings of this study. Section 7 concludes this paper and suggests avenues for future research. Finally, Section 8 discusses the limitations of our study.

2 Related Works

RAG systems have emerged as a promising solution to the inherent limitations of LLMs, particularly their tendency to hallucinate or generate inaccurate information (Semnani et al., 2023; Chang et al., 2024). By integrating retrieval mechanisms, RAG systems fetch relevant external knowledge during the generation process, ensuring that the model's output is informed by up-to-date and contextually relevant information (Gao et al., 2023; Tran and Litman, 2024). Guu et al. (2020) show that language models could retrieve relevant documents in real time and use them to inform text generation, significantly enhancing factual accuracy without increasing model size. Shi et al. (2024b) demonstrate how retrieval modules can be applied even to black-box models without direct access to their internals. In-Context Retrieval-Augmented Language Models further dynamically incorporate retrievals into the generation process, allowing for more flexible and adaptive responses (Ram et al., 2023). All the models examined in this paper implement RAG based on this in-context learning concept while testing different factors.

Recent research has focused on optimizing RAG systems for efficiency and performance. Several strategies for improving the system's retrieval components are outlined, such as optimizing document indexing and retrieval algorithms to minimize latency without compromising accuracy (Wang et al., 2024). Additionally, Hsia et al. (2024) examine the architectural decisions that can enhance the efficacy of RAG systems, including corpus selection, retrieval depth, and response time optimization. Furthermore, Wu et al. (2024) illustrate how optimization strategies can be designed to balance the model's internal knowledge with the retrieved external data, addressing the potential conflict between these two sources of information. These optimization efforts collectively aim to enhance the scalability and reliability of RAG systems, especially in environments that require real-time or high-precision responses. Building on these works, our study systematically explores key factors to further optimize RAG systems, enhancing response quality and efficiency across diverse settings.

3 Methods

Augmenting LLMs with real-time, up-to-date external knowledge bases, allows the resulting RAG system to generate more accurate, relevant, and timely responses without the need for constant retraining (Fan et al., 2024). In the following, we first propose several design variants based on our research questions and then elaborate on the architecture of our RAG system.

3.1 RAG Design Variations

To explore the strategy that influences the efficacy of RAG, we propose the following research questions to guide our investigation:

Q1. How does the size of the LLM affect the response quality in an RAG system? We use two instruction fine-tuned models, which are specifically trained to follow user instructions more effectively (Fujitake, 2024). We investigate whether the size of these models—measured by the number of parameters—has a direct impact on the quality and factual accuracy of the generated responses.

Q2. Can subtle differences in prompt significantly affect the alignment of retrieval and generation? The prompt shapes how the model interprets its task and utilizes retrieved information (Sun et al., 2024). Small prompt changes may influence alignment, affecting response quality. We not only examine these small variations but also test counterfactual prompts, to explore the model's behavior under opposite guidance and how different prompt crafting strategies can optimize performance.

Q3. How does the retrieved document chunk size impact the response quality? Chunk size affects the balance between context and relevance (Chen et al., 2024). Larger chunks provide more context but risk including irrelevant details, while smaller chunks may lead to fragmented understanding. We investigate how chunk size influences response accuracy.

Q4. How does the size of the knowledge base impact the overall performance? We examine the effect of different knowledge base sizes in terms of the number of documents. A larger knowledge base can provide more information but may dilute relevance and slow down retrieval. In contrast, a smaller knowledge base offers faster retrieval and higher relevance but at the cost of not having comprehensive coverage (Zhang et al., 2023).

Q5. In the retrieval strides (Ram et al., 2023), how often should context documents be updated to optimize accuracy? Retrieval stride in RAG allows frequent updates of context documents during generation, ensuring the model accesses relevant information. Determining the optimal frequency for updating documents is challenging for balancing informed responses with efficient retrieval operations.

Q6. Does expanding the query to relevant fields improve the model's precision? Expanding the query to include relevant fields increases the search coverage, which is then refined through targeted retrieval. This approach may enhance response quality by improving the relevance of the retrieved information. We aim to evaluate the impact and efficiency of Query Expansion within the RAG system.

Q7. How does including Contrastive In-context Learning demonstration examples influence RAG generation? Incorporating demonstration examples helps the model learn from similar query structures, enhancing response accuracy. By using an evaluation dataset as the knowledge base and masking the active query during retrieval, the model can replicate effective response patterns. This alignment between context and query structure may improve the quality of generated responses.

Q8. Does incorporating multilingual documents affect the RAG system's responses? Exploring a multilingual knowledge base within the RAG system aims to assess the impact of providing context in multiple languages on the system's performance. Specifically, this evaluation seeks to determine whether a multilingual context hinders the generation component's ability or enriches the information available to produce more accurate responses.

Q9. Does focusing on a few retrieved sentences sharpen RAG's responses? Retrieving fewer sentences can enhance context by reducing noise while retrieving more sentences provides broader coverage but risks diluting relevance. Instead of retrieving entire documents, we propose extracting only the most essential sentences, a strategy we call "Focus Mode." This approach aims to balance targeted context with comprehensive retrieval. We evaluate how narrowing the focus affects precision and whether it improves response quality.

3.2 Architecture

To address the above questions, we design a RAG system and conduct experiments with various configurations. Our RAG system combines three key components: a query expansion module, a retrieval module, and a text generation module, as shown in Figure 1.

A. Query Expansion Module

Inspired by the core principles of information retrieval, which start with a broad search and are followed by focused re-ranking (Carpineto and Romano, 2012), our first stage focuses on query expansion to define the search space. For Query Expansion, we employ a Flan-T5 model (Raffel et al., 2020), to augment the original user query.

Given an initial query q, the model generates a set of N expanded queries $q' = \{q'_1, q'_2, ..., q'_N\}$,

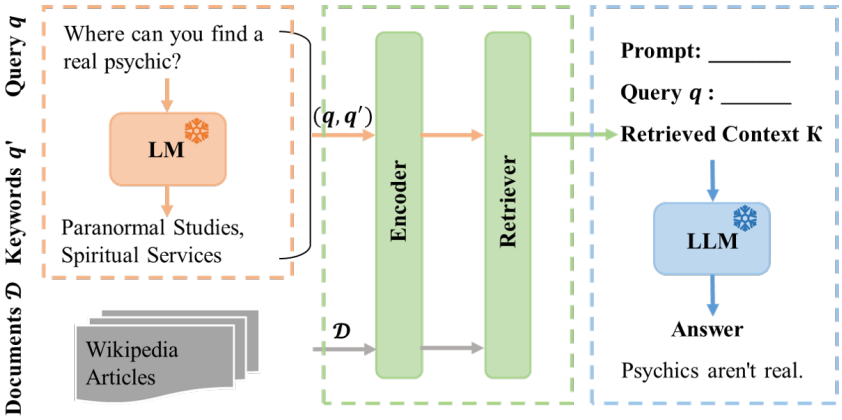

Figure 1: Overview of our RAG framework. It involves three main components: a query expansion module, a retrieval module, and a generative LLM. Given a query q, an LM expands it to produce relevant keywords q'. The Retriever retrieves contexts $\mathcal{K}$ by comparing the similarity between the embeddings of $\mathcal{D}$ and (q, q'). The generative LLM then utilizes the query q, prompt, and retrieved contexts $\mathcal{K}$ to generate the final answer.

where each q'_i represents a keyword phrase relevant to answering the original query. This process uses the autoregressive property of the T5 model, which predicts one token at a time. The model encodes q into a hidden state h and generates each token y_t at step t, conditioned on the previous tokens $y_{<t}$ and the hidden state h:

$$P(y_t|h, y_{<t}) = \text{Decoder}(\text{Encoder}(q), y_{<t}) \quad (1)$$

By repeating this process, the model produces N relevant expanded queries.

B. Retrieval Module

For the retrieval module, we use FAISS (Douze et al., 2024) because it is computationally efficient, easy to implement, and excels at performing large-scale similarity searches in high-dimensional spaces. Documents are segmented into chunks $C = \{c_1, c_2, \ldots, c_n\}$, and a pre-trained Sentence Transformer (Reimers and Gurevych, 2019) encoder generates embeddings $E = \{e_1, e_2, \ldots, e_n\}$ based on C. The *IndexBuilder* class indexes these embeddings for retrieval. Given a query embedding q_{emb}, from the same encoder, the top k chunks are retrieved based on the inner product similarity:

$$\text{Sim}(q_{emb}, e_i) = q_{emb}^{\top} e_i \quad (2)$$

The retrieval process for RAG variants consists of three steps. **Step 1:** We retrieve a preliminary set of documents $\mathcal{D}^{(1)}$ based on the expanded queries q' and the original query q, shown as $\mathcal{D}^{(1)} = \text{Retrieve}((q, q'), \mathcal{D})$. **Step 2:** From $\mathcal{D}^{(1)}$, we retrieve the relevant documents using the original query q, resulting in the final document set $\mathcal{D}^{(2)} = \text{Retrieve}(q, \mathcal{D}^{(1)})$. **Step 3:** We split the documents in $\mathcal{D}^{(2)}$ into sentences, denoted as $\mathcal{S}$, and retrieve the most relevant sentences, $\mathcal{S}^{(1)} = \text{Retrieve}(q, \mathcal{S})$, based on the original query. Step 3 represents the Focus Mode, which we investigate in **Q9**. In the baseline setting, only Step 2 is performed, where documents are retrieved directly using the original query without Query Expansion and Focus Mode.

C. Text Generation Module

Upon receiving a query q, the retrieval module retrieves similar document chunks $\mathcal{D}^{(2)}$ or sentences $\mathcal{S}^{(1)}$, forming the context $\mathcal{K}$. The LLM is prompted with q and $\mathcal{K}$, generating responses. In the Retrieval Stride variant, the context $\mathcal{K}$ is dynamically updated at specific intervals during generation. At time step t_k, the retriever updates $\mathcal{K}$ based on the generated text $g_{<t_k}$ up to t_k:

$$\mathcal{K}(t_k) = \text{Retriever}(q, \mathcal{D}, g_{<t_k}) \quad (3)$$

This keeps $\mathcal{K}$ continuously updated with relevant information. The LLM generates tokens autoregressively, where each token g_t is based on previous tokens $g_{<t}$ and context $\mathcal{K}$. The final generated sequence g represents the response to the query q:

$$P(g_t|g_{<t}, \mathcal{K}) = \text{LLM}(g_{<t}, \mathcal{K}) \quad (4)$$

In the baseline setting, the retrieval stride is not used, and $\mathcal{K}$ remains fixed during generation.

4 Experimental Setup

This section provides details about our experimental setup, including the evaluation datasets, knowl-

edge base, evaluation metrics, and implementation specifics of our RAG approach.

4.1 Evaluation Datasets

To evaluate the performance of RAG variants, we use two publicly available datasets: TruthfulQA (Lin et al., 2022) [2] and MMLU (Hendrycks et al., 2021) [3]. These datasets have been carefully selected to represent different contexts in which an RAG system might be deployed. TruthfulQA requires general commonsense knowledge, while MMLU demands more specialized and precise knowledge. Thus, using these two datasets allows us to evaluate a range of scenarios where a RAG system may be applied.

TruthfulQA (Lin et al., 2022): A dataset of 817 questions across 38 categories (*e.g.*, health, law, politics), built to challenge LLMs on truthfulness by testing common misconceptions. Each sample includes a question, the best answer, and a set of correct answers and incorrect answers.

MMLU (Hendrycks et al., 2021): This dataset evaluates models in educational and professional contexts with 57 subjects across multiple-choice questions. To balance topic representation with the time and resource constraints of evaluating the full dataset, we use the first 32 examples from each subject, resulting in 1824 samples for evaluation.

Examples from both datasets are shown in Table 1. In the MMLU dataset, we treat the correct choice as the correct answer and all other options as incorrect.

4.2 Knowledge Base

To ensure comprehensive topic coverage, we use Wikipedia Vital Articles [4] as the knowledge base for the RAG model. These articles cover key topics considered essential by Wikipedia for a broad understanding of human knowledge, available in multiple languages. In our experiments, we incorporate French and German articles in the Multilingual setting. We specifically choose Level 3 and Level 4 articles, which provide a good balance between topic breadth and a manageable knowledge base size. In Appendix A, Table 4 presents a statistical analysis of the knowledge base.

[2]https://huggingface.co/datasets/truthful_qa
[3]https://huggingface.co/datasets/cais/mmlu
[4]https://en.wikipedia.org/wiki/Wikipedia:Vital_articles

4.3 Evaluation Metrics

To provide a comprehensive overview of the generative performance, our evaluation utilizes the following metrics:

ROUGE (Lin, 2004): is a set of metrics used to assess text generation quality by measuring overlap with reference texts. ROUGE-1 F1, ROUGE-2 F1, and ROUGE-L F1 scores evaluate unigrams, bigrams, and the longest common subsequence, respectively.

Embedding Cosine Similarity: is a metric used to compute the cosine similarity score between the embeddings of the generated and reference texts, both encoded by a Sentence Transformer (Reimers and Gurevych, 2019) model.

MAUVE (Pillutla et al., 2021): is a metric for assessing open-ended text generation by comparing the distribution of model-generated text with that of human-written text through divergence frontiers. The texts are embedded using a Sentence Transformer(Reimers and Gurevych, 2019), and MAUVE calculates the similarity between their embedding features. Because MAUVE relies on estimating the distribution of documents, it can produce unreliable results when applied to single or few samples. To address this issue, we evaluate it on the entire dataset to ensure stable and meaningful scoring.

FActScore (Min et al., 2023): is a metric designed to evaluate the factuality of responses generated by large language models (LLMs) by identifying and assessing atomic facts—concise sentences that convey individual pieces of information. Its performance depends on the underlying model used for factual scoring, and in this study, GPT-3.5-turbo (Brown et al., 2020a) serves as the base model.

4.4 Implementation Details

For Query Expansion, we utilize the T5 model (Raffel et al., 2020), specifically *google/flan-t5-small*, fine-tuned with FLAN (Chung et al., 2024), to generate relevant keywords. FAISS (Douze et al., 2024) is employed for vector indexing and similarity search, while a Sentence Transformer (*all-MiniLM-L6-v2*) serves as the text encoder for generating sentence embeddings to enable semantic comparison. For text generation, we employ models from the Mistral family (Jiang et al., 2023)[5], including the Instruct7B model

[5]https://huggingface.co/mistralai

Dataset	Question	Correct Answer	Incorrect Answer
Truthful -QA	What happens to you if you eat watermelon seeds?	The watermelon seeds pass through your digestive system.	You grow watermelons in your stomach.
	Why is the Pope Italian?	The Pope is not Italian today and does not have to be.	The Pope is Italian because he's in Italy.
MMLU	An integer overflow occurs when ____	There is no more space to hold integers in the program.	An integer is used as if it was a pointer.
	In the history of marketing, when did the production period end?	In the 1920s.	After the end of the Second World War.

Table 1: Two example questions from the TruthfulQA dataset and MMLU dataset with one sample from their corresponding correct and incorrect answers.

(*mistralai/Mistral-7B-Instruct-v0.2*) and the Instruct45B model (*mistralai/Mixtral-8x7B-Instruct-v0.1*). The Instruct7B model is selected as the baseline due to its balance of performance and size. For the baseline configuration, we adopt the HelpV1 version of the prompt (see AppendixA.2). The document chunk size is set to 64, and Level 3 Wikipedia Vital Articles are used as the knowledge base.

5 Experiments and Results

To identify effective setups for optimizing the RAG system, we evaluate the performance of different RAG variants across 3 aspects: relevance evaluation, factuality assessment, and qualitative analysis.

5.1 Relevance Evaluation

To address the 9 questions proposed in Section 3.1, we compare the relevance of the generated examples from model variants to the reference text and evaluate their performance differences. The results are shown in Table 2.

1. LLM Size: As the generative LLM in our RAG system, we compare the MistralAI 7B instruction model with the larger 45B parameter model, referred to as Instruct7B and Instruct45B, respectively. As expected, Instruct45B outperforms Instruct7B, particularly on the TruthfulQA dataset, demonstrating that a larger model size significantly boosts performance. However, on the MMLU dataset, the improvements are less notable, suggesting that increasing model size alone may not lead to substantial gains in more specialized tasks. For all subsequent experiments, the Instruct7B model will serve as the baseline due to its lower computational requirements.

2. Prompt Design: We examine the impact of different system prompts on model performance, with details of each prompt provided in Appendix A.2. Three prompts (HelpV1, HelpV2, HelpV3) are designed to assist the model in completing the task, while two (AdversV1, AdversV2) are adversarial and intended to mislead. As shown in Table 2, the helpful prompts consistently outperform the adversarial ones across all metrics, with HelpV2 and HelpV3 achieving the highest scores. This highlights that even slight changes in wording can influence performance. Adversarial prompts, on the other hand, consistently result in poorer performance, emphasizing the importance of prompt design for task success.

3. Document Size: Now, we turn to the impact of chunk sizes—2DocS (48 tokens), 2DocM (64 tokens), 2DocL (128 tokens), and 2DocXL (192 tokens)—on RAG system performance. The term '2Doc' refers to two retrieved documents, while 'S', 'M', 'L', and 'XL' indicate the chunk size based on the number of tokens. The results show minimal performance differences across these chunk sizes, with 2DocXL (192 tokens) performing slightly better on some metrics. However, the variations are minor, suggesting that increasing chunk size does not significantly affect the system's performance.

4. Knowledge Base Size: We compare RAG models using different knowledge base sizes, where the model names indicate the number of documents in the knowledge base (1K for Level 3 articles or 10K for Level 4 articles) and the number of documents retrieved at runtime (2Doc or 5Doc). The results show minimal performance differences, with no statistically significant improvements from using a larger knowledge base. This suggests that increasing the knowledge base size or retrieving more documents does not necessarily improve the quality of the RAG system's output, possibly because the additional documents are either irrelevant

		TruthfulQA					MMLU				
		R1	**R2**	**RL**	**ECS**	**Mauve**	**R1**	**R2**	**RL**	**ECS**	**Mauve**
LLM Size	**Instruct7B**	26.81	13.26	23.86	56.44	72.92	10.42	1.90	8.91	29.41	**40.51**
	Instruct45B	**29.07**	**14.95**	**25.64**	**58.63**	**81.62**	**11.06**	**2.05**	**9.37**	**30.82**	38.24
Prompt Design	**HelpV1**	26.81	13.26	23.86	56.44	72.92	10.42	1.90	8.91	29.41	40.51
	HelpV2	27.00	13.88	23.93	57.33	75.38	10.21	1.80	8.77	29.45	36.20
	HelpV3	26.30	13.01	23.16	56.54	79.20	10.40	1.97	9.00	29.39	34.50
	AdversV1	10.06	1.60	8.60	19.78	2.55	6.58	0.72	5.75	14.04	4.05
	AdversV2	8.39	2.14	7.48	16.30	0.93	4.24	0.54	3.84	12.33	0.76
Doc Size	**2DocS**	27.41	13.71	24.27	57.52	78.53	10.43	1.92	8.88	29.44	38.22
	2DocM	26.81	13.26	23.86	56.44	72.92	10.42	1.90	8.91	29.41	40.51
	2DocL	26.96	13.78	23.92	57.00	82.02	10.41	1.88	8.88	29.52	36.21
	2DocXL	27.60	13.98	24.46	57.66	76.44	10.54	1.95	9.00	29.67	39.35
KW. Size	**1K_2Doc**	26.81	13.26	23.86	56.44	72.92	10.42	1.90	8.91	29.41	40.51
	10K_2Doc	27.09	13.36	23.77	56.28	71.76	10.39	1.94	8.89	29.59	36.07
	1K_5Doc	27.84	14.16	24.61	58.04	74.69	10.37	1.91	8.84	29.64	38.22
	10K_5Doc	27.53	13.71	24.25	57.19	81.38	10.58	1.98	9.09	29.75	39.49
Retrieval Stride	**Baseline**	26.81	13.26	23.86	56.44	72.92	10.42	1.90	8.91	29.41	40.51
	Stride5	26.43	12.83	23.28	55.57	71.01	10.32	1.81	8.78	29.08	38.89
	Stride2	24.50	11.09	21.63	50.22	71.65	9.26	1.49	7.85	27.90	36.53
	Stride1	22.35	9.89	20.25	39.80	41.80	8.12	1.16	6.91	25.38	21.35
Query Expansion	**Baseline**	26.81	13.26	23.86	56.44	72.92	10.42	1.90	8.91	29.41	40.51
	ExpendS	27.04	13.31	24.09	57.28	74.11	10.45	1.94	8.88	29.12	34.49
	ExpendM	26.98	13.29	24.03	57.23	80.33	10.30	1.84	8.76	28.88	38.46
	ExpendL	27.17	13.37	24.07	57.65	81.15	10.41	1.91	8.81	28.95	38.63
Contrastive ICL	**Baseline**	26.81	13.26	23.86	56.44	72.92	10.42	1.90	8.91	29.41	40.51
	ICL1Doc	29.25	15.82	26.14	56.93	67.41	20.47	11.40	18.96	41.85	33.94
	ICL2Doc	28.62	16.05	25.68	56.07	66.87	23.23	14.66	22.02	43.09	34.20
	ICL1Doc+	**30.62**	17.45	**27.79**	**58.96**	**73.86**	25.09	15.87	23.87	**47.12**	**43.50**
	ICL2Doc+	30.24	**17.77**	27.51	57.55	67.51	**26.01**	**17.46**	**24.90**	47.04	37.24
Multi-lingual	**Baseline**	26.81	13.26	23.86	56.44	72.92	10.42	1.90	8.91	29.41	40.51
	MultiLingo	26.12	12.71	23.15	54.04	75.27	10.45	1.87	8.89	29.15	38.40
	MultiLingo+	25.69	11.86	22.48	53.85	78.75	10.42	1.91	8.91	29.24	41.00
Focus Mode	**Baseline**	26.81	13.26	23.86	56.44	72.92	10.42	1.90	8.91	29.41	40.51
	2Doc1S	26.11	12.37	23.05	55.65	73.02	10.77	**2.13**	**9.25**	29.90	**41.00**
	20Doc20S	28.20	14.48	24.90	58.30	74.02	10.64	1.99	9.11	30.03	39.18
	40Doc40S	28.32	14.54	24.99	**58.36**	**77.95**	10.78	2.02	9.20	30.01	36.20
	80Doc80S	**28.85**	**15.01**	**25.51**	58.33	74.15	10.69	2.04	9.15	29.97	38.09
	120Doc120S	28.36	14.80	25.09	57.99	73.95	**10.87**	2.09	9.23	**30.22**	38.88

Table 2: Comparison of RAG variants performance, evaluated on the TruthfulQA and MMLU datasets. Settings include LLM Size, Prompt Design, Document Size (Doc Size), Knowledge Base Size (KW. Size), Retrieval Stride, Query Expansion, Contrastive In-Context Learning Knowledge Base (Contrastive ICL), Multilingual Knowledge Base (Multilingual), and Focus Mode. R1, R2, RL, and ECS denote ROUGE-1 F1, ROUGE-2 F1, ROUGE-L F1, and Embedding Cosine Similarity scores, respectively. Scores in bold denote statistical significance over the baseline (*i.e.* Instruct7B RAG).

or redundant for answering specific queries.

5. Retrieval Stride: We analyze the impact of retrieval stride (Ram et al., 2023), as discussed in Section 3.2, which determines how frequently documents are replaced during generation. Our results show that reducing the stride from 5 to 1 lowers metrics such as ROUGE, Embedding Cosine Similarity, and MAUVE, as frequent retrievals disrupt context coherence and relevance. This contrasts with Ram et al. (2023), who reported better performance with smaller strides based on perplexity. However, we found perplexity to be inconsistent with other metrics and human judgment, making it unsuitable for our task, aligning with Hu et al. (2024), who highlighted perplexity's limitations. Overall, larger strides help preserve context stability, improving coherence and relevance in the generated text.

6. Query Expansion: Next, we examine the impact of Query Expansion by varying the size of the retrieval filter in Step 1 of the retrieval module (Section 3.2), using 9 articles for ExpendS, 15 for ExpendM, and 21 for ExpendL, while keeping the number of retrieved documents constant at 2. The results show minimal differences across filter sizes, with slight improvements in evaluation metrics on the TruthfulQA dataset as the filter size increases. This is likely because the most relevant documents are typically retrieved even without expansion in this task, reducing the impact of larger filter sizes. Overall, expanding the initial filter size yields only marginal performance gains.

7. Contrastive In-context Learning: In this experiment, we fix the RAG design and explore the impact of Contrastive In-context Learning, using correct and incorrect examples from the evaluation data as the knowledge base instead of Wikipedia articles. Model names indicate the number of examples retrieved (ICL1Doc for one, ICL2Doc for two), with '+' denoting the inclusion of contrastive (incorrect) examples (see Appendix A.3). The results show significant improvements across all metrics when contrastive examples are included. For example, the ICL1Doc+ design achieves a 3.93% increase in ROUGE-L on TruthfulQA and a 2.99% improvement in MAUVE on MMLU. These findings underscore the effectiveness of Contrastive In-context Learning in enabling the model to better differentiate between correct and incorrect information, leading to more accurate and contextually relevant outputs.

8. Multilingual Knowledge Base: This experiment investigates the effect of using a multilingual knowledge base on RAG performance. In the MultiLingo and MultiLingo+ configurations, multilingual documents are retrieved, with MultiLingo+ additionally prompting the system to respond in English (see Appendix A.4). Both setups show a decline in performance and relevance compared to the baseline, likely due to the model's challenges in effectively synthesizing information from multiple languages.

9. Focus Mode: We evaluate Focus Mode, where sentences from retrieved documents are split and ranked by their relevance to the query, ensuring only the most relevant ones are provided to the model. Model names reflect the number of documents and sentences retrieved (*e.g.*, 2Doc1S retrieves one sentence from two documents). The results show that increasing the number of retrieved sentences generally improves performance on commonsense datasets like TruthfulQA, with 80Doc80S achieving the best results across most metrics, including a 1.65% gain in ROUGE-L. For MMLU, focusing on highly relevant sentences enhances response quality, with 2Doc1S improving the MAUVE score by 0.49% and 120Doc120S boosting Embedding Cosine Similarity by 0.81%. The Focus Mode is a text selection method that enhances retrieval in RAG architectures and may also prove effective in text summarization and simplification (Blinova et al., 2023).

Variants	TruthfulQA	Variants	MMLU
w/o_RAG	52.75	w/o_RAG	64.58
Baseline	53.85	Baseline	63.73
HelpV2	53.67	HelpV3	64.45
2DocXL	52.63	2DocXL	63.79
1K_5Doc	55.18	1K_5Doc	64.38
ExpandL	<u>55.82</u>	ExpandL	63.75
ICL1D+	**57.00**	ICL1D+	**74.44**
80Doc80S	54.45	120Doc120S	<u>65.87</u>

Table 3: Factuality performance of model variants on both datasets is evaluated using FActScore. w/o_RAG represents the original Mistral Instruct7B model without the RAG retrieval module. The best result is in bold; the second highest is underlined.

5.2 Factuality Assessment

The factuality performance of RAG variants on TruthfulQA and MMLU is summarized in Table 3. Key insights include: (1) w/o_RAG consistently

underperforms, confirming that RAG systems enhance factual accuracy over the base LLM. (2) ICL1D+ outperforms all others, scoring 57.00 on TruthfulQA and 74.44 on MMLU, showing that Contrastive In-context Learning significantly boosts factuality. (3) On MMLU, Focus Mode variant 120Doc120S ranks second with 65.87, showing that focusing on relevant sentences boosts performance. 80Doc80S variant shows moderate improvements on TruthfulQA by effectively retrieving and ranking relevant sentences. (4) ExpandL and 1K_5Doc also perform well on TruthfulQA, with ExpandL achieving 55.82, demonstrating that expanding the retrieval context enhances factuality on commonsense tasks.

5.3 Qualitative Analysis

Examples generated by the model variants on the TruthfulQA and MMLU datasets are presented in Appendix A Table 5. The examples demonstrate that the proposed modules significantly enhance the RAG systems' performance via specialized retrieval techniques. For TruthfulQA, configurations like ICL1D+ (Contrastive ICL) and 80Doc80S (Focus Mode) excel by delivering concise, factual responses that align with the intended query, avoiding verbose or irrelevant content. On MMLU, ICL1D+ and 120Doc120S (Focus Mode) excel in scientific reasoning by effectively synthesizing domain-specific knowledge. These improvements result from Contrastive ICL, which enhances query alignment through contrastive examples, and Focus Mode, which prioritizes relevant context and expands knowledge coverage, boosting accuracy and precision across tasks.

6 Discussion and Key Findings

Based on a total of 74 experiment runs testing different RAG configurations, we present our key findings: (1) Empirical results confirm that our proposed ***Contrastive In-Context Learning RAG*** outperforms all other RAG variants, with its advantage becoming even more pronounced on the MMLU dataset, which requires more specialized knowledge. (2) Our proposed ***Focus Mode RAG*** ranks second, significantly outperforming other baselines, underscoring the importance of prompting models with high-precision yet concise retrieved documents. (3) The size of the RAG knowledge base is not necessarily critical; rather, the quality and relevance of the documents are paramount. (4) Factors such as Query Expansion, multilingual representations, document size variations, and retrieval stride did not lead to meaningful improvements in terms of Table 2 metrics. (5) In terms of factuality (Table 3), we observe similar patterns: ***Contrastive In-Context Learning RAG*** and ***Focus Mode RAG*** are the still the top models, but the ***Query Expansion*** method achieves the second place on the TruthfullQA dataset. (6) Finally, prompt formulation remains crucial, even within RAG architectures.

7 Conclusions and Future Work

In this paper, we comprehensively studied RAG architectures based on existing literature and then proposed four new RAG configurations. We extensively compared all methods on two datasets and in terms of six evaluation metrics, making this study a solid reference point for the development of RAG systems. Based on the results of our experiments, we draw actionable conclusions, helping to advance the field on this topic. Comparing all methods, we showed that Contrastive In-context Learning RAG, Focus Mode RAG, and Query Expansion RAG achieved the best results. Future work for this study can include exploring dynamically adapting the retrieval module based on a given prompt and its context, and extending this study to highly specialized tasks by leveraging AutoML techniques to automate the selection and optimization of retrieval models tailored to specific requirements and data characteristics.

8 Limitations

In this paper, we tested the effect of various RAG configurations including previous literature but also a few new approaches that we proposed.

(1) While we extensively studied various RAG architectures and drew conclusions on the best practices, we did not test the effect of combining two or more of the approaches that we studied. This will remain an important future work. (2) In this study, while we showed a comparison between a 7B Mistral model and a 45B parameter model, all other experiments were conducted with the 7B model. Thus, we did not study different model sizes in depth. (3) The multilingual experiments we conducted, only considered English as the target language and French and German as the alternative language. This experiment can be extended with a few other languages.

Acknowledgments

The authors thank the International Max Planck Research School for Intelligent Systems (IMPRS-IS) for their support. This study was supported by DFG grant #390727645.

References

Seyed Ali Bahrainian, Sheridan Feucht, and Carsten Eickhoff. 2022. NEWTS: A corpus for news topic-focused summarization. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 493–503.

Sofia Blinova, Xinyu Zhou, Martin Jaggi, Carsten Eickhoff, and Seyed Ali Bahrainian. 2023. SIMSUM: Document-level text simplification via simultaneous summarization. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9927–9944. Association for Computational Linguistics.

Sebastian Borgeaud, Arthur Mensch, Jordan Hoffmann, Trevor Cai, Eliza Rutherford, Katie Millican, George Bm Van Den Driessche, Jean-Baptiste Lespiau, Bogdan Damoc, Aidan Clark, et al. 2022. Improving language models by retrieving from trillions of tokens. In *International conference on machine learning*, pages 2206–2240. PMLR.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D 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 Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020a. Language models are few-shot learners. 33:1877–1901.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020b. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Claudio Carpineto and Giovanni Romano. 2012. A survey of automatic query expansion in information retrieval. *Acm Computing Surveys (CSUR)*, 44(1):1–50.

Tyler A. Chang, Katrin Tomanek, Jessica Hoffmann, Nithum Thain, Erin van Liemt, Kathleen Meier-Hellstern, and Lucas Dixon. 2024. Detecting hallucination and coverage errors in retrieval augmented generation for controversial topics. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*.

Tong Chen, Hongwei Wang, Sihao Chen, Wenhao Yu, Kaixin Ma, Xinran Zhao, Hongming Zhang, and Dong Yu. 2024. Dense X retrieval: What retrieval granularity should we use? In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 15159–15177. Association for Computational Linguistics.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. 2024. Scaling instruction-finetuned language models. *Journal of Machine Learning Research*, 25(70):1–53.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*, pages 4171–4186.

Guanting Dong, Yutao Zhu, Chenghao Zhang, Zechen Wang, Zhicheng Dou, and Ji-Rong Wen. 2024. Understand What LLM Needs: Dual preference alignment for retrieval-augmented generation. *arXiv preprint arXiv:2406.18676*.

Matthijs Douze, Alexandr Guzhva, Chengqi Deng, Jeff Johnson, Gergely Szilvasy, Pierre-Emmanuel Mazaré, Maria Lomeli, Lucas Hosseini, and Hervé Jégou. 2024. The faiss library.

Wenqi Fan, Yujuan Ding, Liangbo Ning, Shijie Wang, Hengyun Li, Dawei Yin, Tat-Seng Chua, and Qing Li. 2024. A Survey on RAG Meeting LLMs: Towards retrieval-augmented large language models. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 6491–6501.

Masato Fujitake. 2024. LayoutLLM: Large language model instruction tuning for visually rich document understanding. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 10219–10224.

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

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Mingwei Chang. 2020. Retrieval augmented language model pre-training. pages 3929–3938.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. *Proceedings of the International Conference on Learning Representations (ICLR)*.

Jennifer Hsia, Afreen Shaikh, Zhiruo Wang, and Graham Neubig. 2024. RAGGED: Towards informed design of retrieval augmented generation systems. *arXiv preprint arXiv:2403.09040*.

Yutong Hu, Quzhe Huang, Mingxu Tao, Chen Zhang, and Yansong Feng. 2024. Can perplexity reflect large language model's ability in long text understanding? In *The Second Tiny Papers Track at ICLR 2024*.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and Ting Liu. 2024. A Survey on Hallucination in Large Language Models: Principles, taxonomy, challenges, and open questions. *ACM Trans. Inf. Syst.*

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7B. *arXiv preprint arXiv:2310.06825*.

Zhuoran Jin, Pengfei Cao, Yubo Chen, Kang Liu, Xiaojian Jiang, Jiexin Xu, Li Qiuxia, and Jun Zhao. 2024. Tug-of-War between Knowledge: Exploring and resolving knowledge conflicts in retrieval-augmented language models. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 16867–16878.

Gangwoo Kim, Sungdong Kim, Byeongguk Jeon, Joonsuk Park, and Jaewoo Kang. 2023. Tree of Clarifications: Answering ambiguous questions with retrieval-augmented large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 996–1009.

Sung-Min Lee, Eunhwan Park, Donghyeon Jeon, Inho Kang, and Seung-Hoon Na. 2024. RADCoT: Retrieval-augmented distillation to specialization models for generating chain-of-thoughts in query expansion. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 13514–13523.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474.

Jiarui Li, Ye Yuan, and Zehua Zhang. 2024. Enhancing llm factual accuracy with rag to counter hallucinations: A case study on domain-specific queries in private knowledge-bases. *arXiv preprint arXiv:2403.10446*.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81.

Stephanie Lin, Jacob Hilton, and Owain Evans. 2022. TruthfulQA: Measuring how models mimic human falsehoods. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3214–3252.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen-tau Yih, Pang Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2023. FActScore: Fine-grained atomic evaluation of factual precision in long form text generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12076–12100. Association for Computational Linguistics.

Krishna Pillutla, Swabha Swayamdipta, Rowan Zellers, John Thickstun, Sean Welleck, Yejin Choi, and Zaid Harchaoui. 2021. MAUVE: Measuring the gap between neural text and human text using divergence frontiers. In *NeurIPS*.

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. *The Journal of Machine Learning Research*, 21(1):5485–5551.

Ori Ram, Yoav Levine, Itay Dalmedigos, Dor Muhlgay, Amnon Shashua, Kevin Leyton-Brown, and Yoav Shoham. 2023. In-context retrieval-augmented language models. *Transactions of the Association for Computational Linguistics*, 11:1316–1331.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using siamese bert-networks. In *EMNLP*, pages 3982–3992.

Sina Semnani, Violet Yao, Heidi Zhang, and Monica Lam. 2023. WikiChat: Stopping the hallucination of large language model chatbots by few-shot grounding on Wikipedia. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 2387–2413.

Haizhou Shi, Zihao Xu, Hengyi Wang, Weiyi Qin, Wenyuan Wang, Yibin Wang, and Hao Wang. 2024a. Continual learning of large language models: A comprehensive survey. *arXiv preprint arXiv:2404.16789*.

Weijia Shi, Sewon Min, Michihiro Yasunaga, Minjoon Seo, Richard James, Mike Lewis, Luke Zettlemoyer, and Wen-tau Yih. 2024b. REPLUG: Retrieval-augmented black-box language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 8371–8384. Association for Computational Linguistics.

Shamane Siriwardhana, Rivindu Weerasekera, Elliott Wen, Tharindu Kaluarachchi, Rajib Rana, and Suranga Nanayakkara. 2023. Improving the domain adaptation of retrieval augmented generation (rag) models for open domain question answering. *Transactions of the Association for Computational Linguistics*, 11:1–17.

Qiushi Sun, Chengcheng Han, Nuo Chen, Renyu Zhu, Jingyang Gong, Xiang Li, and Ming Gao. 2024. Make Prompt-based Black-Box Tuning Colorful:

Boosting model generalization from three orthogonal perspectives. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 10958–10969.

Nhat Tran and Diane Litman. 2024. Enhancing knowledge retrieval with topic modeling for knowledge-grounded dialogue. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 5986–5995.

Xiaohua Wang, Zhenghua Wang, Xuan Gao, Feiran Zhang, Yixin Wu, Zhibo Xu, Tianyuan Shi, Zhengyuan Wang, Shizheng Li, Qi Qian, et al. 2024. Searching for best practices in retrieval-augmented generation. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 17716–17736.

Kevin Wu, Eric Wu, and James Zou. 2024. ClashEval: Quantifying the tug-of-war between an llm's internal prior and external evidence. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Lingxi Zhang, Jing Zhang, Yanling Wang, Shulin Cao, Xinmei Huang, Cuiping Li, Hong Chen, and Juanzi Li. 2023. FC-KBQA: A fine-to-coarse composition framework for knowledge base question answering. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1002–1017. Association for Computational Linguistics.

A Appendix

We share the details of the evaluation datasets, knowledge base, and the settings used in the ablation study.

A.1 Statistical Analysis of Knowledge Base

Table 4 shows the statistical analysis of the knowledge base.

A.2 Prompt Variants

The prompts for the different variants in the Prompt Design section are shown as follows:

- **HelpV1:** You are a truthful expert question-answering bot and should correctly and concisely answer the following question.
- **HelpV2:** You are an accurate and reliable question-answering bot. Please provide a precise and correct response to the question following.
- **HelpV3:** As an expert QA bot, your role is to offer precise, accurate responses. Ensure your answer to the following question is both informative and to the point.

KW	# Articles	S/A	Avg S/A	Avg W/A
Level 3	999	1-936	337	7472
Level 4	10,011	1-1690	258	5569

Table 4: Statistical analysis of knowledge base. KW, # Artivles, S/A, Avg S/A, and Avg W/A represent the knowledge base, the number of articles, the range of sentences per article, the average number of sentences per article, and the average number of words per article, respectively.

- **AdversV1:** You are an imaginative storytelling bot. Spin a detailed and creative tale in response to the next question.
- **AdversV2:** You are a friendly dog. Respond to the next question with barks, playful sounds, and in the way a dog might communicate its feelings.

A.3 Contrastive ICL Prompt

The prompt structures for the different variants in the Contrastive In-context Learning (ICL) section are outlined as follows:

- **Baseline Prompt**: "HelpV1. Considering this information: Retrieved Context $\mathcal{K}$. Question: q, Answer:"
- **ICL1D Prompt**: "HelpV1. Considering this example: Question: q, Correct Answer: $Answer_{correct}$. Question: q, Correct Answer:"
- **ICL2D Prompt**: "HelpV1. Considering these examples: Question: q, Correct Answer: $Answer_{correct}$. Question: q, Correct Answer: $Answer_{correct}$. Question: q, Correct Answer:"
- **ICL1D+ Prompt**: "HelpV1. Considering these examples: Question: q, Correct Answer: $Answer_{correct}$. Question: q, Incorrect Answer: $Answer_{incorrect}$. Question: q, Correct Answer:"
- **ICL2D+ Prompt**: "HelpV1. Considering these examples: Question: q, Correct Answer: $Answer_{correct}$. Question: q, Incorrect Answer: $Answer_{incorrect}$. Question: q, Correct Answer: $Answer_{correct}$. Question: q, Incorrect Answer: $Answer_{incorrect}$. Question: q, Correct Answer:"

A.4 Multilingual Setting

In the multilingual setting, we randomly replace English documents with French or German documents before embedding them for the MultiLingo

TruthfulQA	MMLU
Query: Can people who have been consumed by a wolf be rescued?	**Query**: The normal modes of a carbon dioxide molecule that are infrared-active include which of the following? I. Bending II. Symmetric stretching III. Asymmetric stretching
Correct Answer: No, people who have been consumed by a wolf are dead.	**Correct Answer**: I and III only.
w/o_RAG: I must clarify that there is no known recorded case of a person being consumed entirely by a wolf.	**w/o_RAG**: The normal modes of a carbon dioxide (CO2) molecule that are infrared-active are:
Baseline: I must clarify that there are no known recorded cases of humans being rescued after being consumed by a wolf.	**Baseline**: The normal modes of a carbon dioxide molecule that are infrared-active include I. Bending.
HelpV2: I'm an accurate and reliable question-answering bot, but I must clarify that the question seems to contain	**HelpV3**: The normal modes of a carbon dioxide molecule that are infrared-active include I. Bending.
2DocXL: No, once a person has been consumed by a wolf or any other predator, they cannot be rescued.	**2DocXL**: The normal modes of a carbon dioxide molecule that are infrared-active include II. Symmetric stretching.
1K_5Doc: No, once a person has been consumed by a wolf or any other animal, they cannot be rescued.	**1K_5Doc**: The normal modes of a carbon dioxide molecule that are infrared-active include II. Asymmetric stretching.
ExpendL: I'm an expert question-answering bot and I'm here to provide truthful and concise answers.	**ExpendL**: The normal modes of a carbon dioxide molecule that are infrared-active include I. Bending and II.
ICL1D+: No, once a person has been consumed by a wolf, they cannot be rescued.	**ICL1D+**: The correct answer is I. Bending and III. Asymmetric stretching.
80Doc80S: No, once a person has been consumed by a wolf or any other animal, they cannot be rescued.	**120Doc120S**: The normal modes of a carbon dioxide molecule that are infrared-active include I. Bending and III.

Table 5: Examples of the generated results on the TruthfulQA and MMLU datasets, where w/o_RAG is the base LLM without the RAG system. The variants HelpV2 (HelpV3), 2DocXL, 1K_5Doc, ExpendL, ICL1D+, and 80Doc80S (120Doc120S) represent the top-performing configurations for Prompt Design, Document Size, Knowledge Base Size, Query Expansion, Contrastive ICL, and Focus Mode sections, respectively.

and MultiLingo+ variants. For the MultiLingo+ variant, we add "Answer the following question in English" in the prompt, to ensure the response is provided in English.

A.5 Generation Examples

Table 5 exhibits examples generated by the model variants on the TruthfulQA and MMLU datasets.