R^{2}AG: Incorporating Retrieval Information into Retrieval Augmented Generation

Anonymous ACL submission

Abstract

Retrieval augmented generation (RAG) has been applied in many scenarios to augment large language models (LLMs) with external documents provided by retrievers. However, a semantic gap exists between LLMs and retrievers due to differences in their training objectives and architectures. This misalignment forces LLMs to passively accept the documents provided by the retrievers, leading to incomprehension in the generation process, where the LLMs are burdened with the task of distinguishing these documents using their inherent knowledge. This paper proposes R^{2}AG, a novel enhanced RAG framework to fill this gap by incorporating **R**etrieval information into **R**etrieval **A**ugmented **G**eneration. Specifically, R^{2}AG utilizes the nuanced features from the retrievers and employs a R^2-Former to capture retrieval information. Then, a retrieval-aware prompting strategy is designed to integrate retrieval information into LLMs' generation. Notably, R^{2}AG suits low-source scenarios where LLMs and retrievers are frozen. Extensive experiments across five datasets validate the effectiveness, robustness, and efficiency of R^{2}AG. Our analysis reveals that retrieval information serves as an anchor to aid LLMs in the generation process, thereby filling the semantic gap.[1]

1 Introduction

Retrieval augmented generation (RAG) (Lewis et al., 2020) significantly enhances the capabilities of large language models (LLMs) by integrating external, non-parametric knowledge provided by retrievers. In RAG framework, the retriever locates and looks up useful documents based on a given query, and then the LLM interacts with these retrieved results to generate a response. The coordination of retrieval and generation achieves impressive performance without additional training. Especially in domain-specific and knowledge-intensive tasks, RAG offers real-time knowledge with high interpretability to LLMs, effectively mitigating the hallucination problem (Mallen et al., 2023).

[1]The source code is available at https://anonymous.4open.science/r/RRAG.

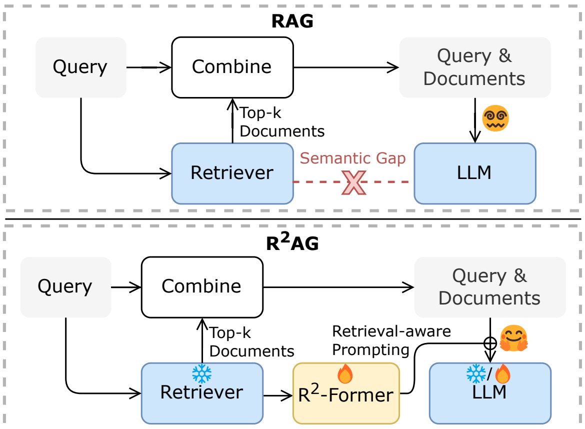

Figure 1: A comparison between RAG and R^{2}AG. R^{2}AG employs a trainable R^2-Former to bridge the semantic gap between retrievers and LLMs. Optionally, LLMs can be fine-tuned to understand the retrieval information further.

However, there exists a semantic gap between retrievers and LLMs due to their vastly different training objectives and architectures (BehnamGhader et al., 2022). Specifically, retrievers, typically encoder architecture, are designed to retrieve the most relevant documents for a query (Zhu et al., 2023b). Conversely, LLMs, generally decoder architecture, are expected to answer questions based on their inherent knowledge or given documents. However, the interaction between retrievers and LLMs in RAG primarily relies on simple text concatenation (BehnamGhader et al., 2022). This poor communication strategy will lead to several challenges for LLMs. **Externally**, it is hard for LLMs to utilize more information from retrievers in separate processes. In RAG, the retrieved documents that only preserve sequential relationships are unidirectionally delivered to LLMs, and LLMs do not fully understand why retrievers provide the documents.

Particularly, low-quality documents inevitably appear in retrieved results (Barnett et al., 2024), but LLMs have to accept this noise passively. **Internally**, it is hard for LLMs to handle all of the retrieved documents with their inherent knowledge. LLMs must process all the results and assess which documents are important, impacting their ability to generate accurate answers (Wu et al., 2024). Moreover, LLMs face the lost-in-middle problem in overly long documents (Liu et al., 2023), leading to further misunderstanding.

Unfortunately, existing enhanced RAG methods, including pre-processing approaches (Izacard et al., 2022; Yan et al., 2024; Asai et al., 2023; Ke et al., 2024) and compression-based approaches (Yan et al., 2024; Xu et al., 2023; Jiang et al., 2023), do not recognize this semantic gap between retrievers and LLMs. They remain to treat retrieval and generation as separate processes and directly add processed or compressed documents into the inputs for LLMs. These strategies ignore the semantic connections necessary for deeper comprehension, which may lead to potentially misleading LLMs even with perfect retrievers.

To address these challenges, it is essential to bridge the semantic gap between retrievers and LLMs. As previously mentioned, retrievers can provide high-quality semantic representations that can be beneficial for catching nuanced differences among documents (Zhao et al., 2022). Thus, our intuition is to exploit these semantic representations as additional knowledge, empower LLMs to gain a deeper comprehension of the retrieved documents, and thereby generate more accurate responses.

This paper proposes a cost-effective enhanced RAG framework to incorporate **R**etrieval information into **R**etrieval **A**rgumented **G**eneration (named R^{2}AG), enhancing LLMs' perception of the key information among retrieved documents. Specifically, R^{2}AG adopts an input processing pipeline that transforms semantic representations from a retriever into unified retrieval features. Then, a trainable R^2-Former is employed to capture essential retrieval information. As shown in Figure 1, R^2-Former is a pluggable and lightweight model placed between the retriever and the LLM. Finally, through a retrieval-aware prompting strategy, the LLM receives additional embeddings that contain retrieval information. This strategy aligns the knowledge from retrievers with LLMs without changing the content and order of retrieved documents, thereby relieving information loss. R^{2}AG offers the flexibility to fine-tune R^2-Former alone or both with LLMs. Thus, in R^{2}AG framework, both retrievers and LLMs can be frozen to save computational costs, making R^{2}AG suitable for scenarios with limited resources. Overall, our contributions are summarized as follows:

- We propose R^{2}AG, an enhanced RAG framework, to incorporate retrieval information into retrieval augmented generation. Notably, R^{2}AG is compatible with low-source scenarios where retrievers and LLMs are frozen.
- We design a lightweight model, R^2-Former, to bridge the semantic gap between retrievers and LLMs. R^2-Former can be seamlessly integrated into existing RAG frameworks using open-source LLMs.
- We introduce a retrieval-aware prompting strategy to inject retrieval information into the input embeddings, enhancing LLMs' ability to understand relationships among documents without much increase in complexity.

Experimental results demonstrate the superior performance and robustness of R^{2}AG in various scenarios. Our analysis shows that R^{2}AG increases latency by only 0.8% during inference. Furthermore, it demonstrates that retrieval information anchors LLMs to understand retrieved documents and enhances their generation capabilities.

2 Related Works

2.1 Retrieval Augmented Generation

Despite being trained on vast corpora, LLMs still struggle with hallucinations and updated knowledge in knowledge-sensitive tasks (Zhao et al., 2023). RAG (Lewis et al., 2020) is regarded as an efficient solution to these issues by combining a retrieval component with LLMs. In detail, documents gathered by retrievers are bound with the original query and placed into the inputs of LLMs to produce final responses. RAG allows LLMs to access vast, up-to-date data in a flexible way, leading to better performance. Benefiting from the progress of multi-modal alignment techniques (Li et al., 2023b; Zhu et al., 2023a), the idea of RAG has been extended to various domains with modality-specific retrievers, including audios (Koizumi et al., 2020), images (Yasunaga et al., 2023), knowledge graphs (He et al., 2024), and so on. Despite its rapid growth, RAG suffers several limitations, such as sensitivity to retrieval results, increased complexity, and a semantic gap between retrievers and LLMs (Kandpal et al., 2022; Zhao et al., 2024).

2.2 Enhanced RAG

Recent works develop many enhanced approaches based on the standard RAG framework. To directly improve the effectiveness of RAG, REPLUG (Shi et al., 2023) and Atlas (Izacard et al., 2022) leverage the LLM to provide a supervisory signal for training a better retriever. However, the noise will inevitably appear in retrieval results (Barnett et al., 2024). Recent studies focus on pre-processing the retrieved documents before providing them to LLMs. Techniques such as truncation and selection are effective methods to enhance the quality of ranking lists without modifying the content of documents (Gao et al., 2023; Xu et al., 2024). CRAG (Yan et al., 2024) trains a lightweight retrieval evaluator to exclude irrelevant documents. BGM (Ke et al., 2024) is proposed to meet the preference of LLMs by training a bridge model to re-rank and select the documents. Some studies aim to train small LMs to compress the retrieval documents, thus decreasing complexity or reducing noise. Jiang et al. (2023) propose LongLLMLingua to detect and remove unimportant tokens. RECOMP (Xu et al., 2023) adopts two compressors to select and summarize the retrieved documents. However, the pre-processing methods introduce additional computational costs during inference and may lead to the loss of essential information.

Notably, the above methods target providing higher-quality retrieval results to LLMs and actually treat retrieval and generation as two distinct processes. This separation fails to bridge the semantic gap between retrievers and LLMs fully.

3 R^{2}AG

3.1 Problem Formulation and Overview

RAG involves the task that aims to prompt an LLM to generate answers based on a query and documents returned by a retriever. Formally, given a query q and a list of documents $\mathcal{D}=\{d_1, d_2, \cdots, d_k\}$ in preference order ranked by the retriever $f_{\mathbf{R}}$, the LLM, a generator $f_{\mathbf{G}}$, is expected to generate the output $\hat{y}$. The pipeline can be expressed as:

$$\hat{y} = f_{\mathbf{G}}\left(\mathrm{P}\left(q, \mathcal{D}\right)\right), \tag{1}$$

where P is a predefined prompt template. It shows the retrievers and LLMs are couple in a simplistic prompt-based method, which will lead to miscommunication and the semantic gap.

Figure 2 illustrates the overall framework of R^{2}AG. Initially, given a query and retrieved documents, R^{2}AG processes representations modeled by a retriever into unified-format features. These list-wise features consider nuanced relationships both between the query and documents and among the documents themselves. Then, a R^2-Former is designed to capture retrieval information for LLM usage. It allows unified features to interact with each other via self-attention mechanism, enabling it to understand complex dependencies. To integrate retrieval information into the LLM's generation process, R^{2}AG adopts a retrieval-aware prompting strategy to insert the retrieval information into the LLM's input embedding space without causing information loss or increasing much complexity. Besides, R^{2}AG is flexible to be applied in low-source scenarios where LLMs are frozen.

3.2 Retrieval Feature Extraction

Before generation, it is necessary to obtain high-quality retrieval features. In R^{2}AG, we first get semantic representations from the retriever f_R. Formally, a query q and document d are encoded into representations as $\mathbf{x}^q{=}f_{\mathbf{R}}(q)$ and $\mathbf{x}^d{=}f_{\mathbf{R}}(d)$, respectively. However, these representations can not be directly used because a single representation can not capture interactive features for LLM's generation. Moreover, to suit various retrievers, it is intuitive to transform representations in different spaces into unified format features.

Inspired by works in retrieval downstream tasks (Ma et al., 2022; Ye and Li, 2024), we align these representations into retrieval features by computing relevance, precedent similarity, and neighbor similarity scores. Specifically, these scores are calculated by a similarity function such as dot product or cosine similarity. The relevance score r_i is between the query and the i-th document and is also used to sort the documents. The precedent and neighbor similarity scores are computed between the i-th document representation and its precedent-weighted and adjacent representations, respectively. Detailed formulations are provided in Appendix A.

Finally, three features are concatenated as input: $\text{input}_i{=}\{r_i, \gamma_i, \zeta_i\}$, representing relevance, precedent similarity, and neighbor similarity. Then, the feature list $\{\text{input}_i\}_{i=1}^k$ is then fed into R^2-Former

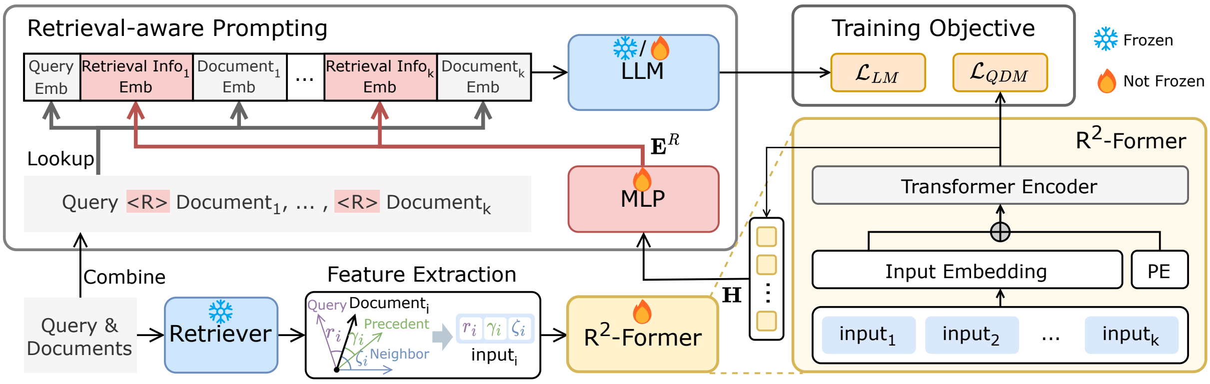

Figure 2: An illustration of R^{2}AG. The R^2-Former is designed to extract retrieval features, acting as an information bottleneck between retrievers and LLMs. Through the retrieval-aware prompting strategy, the retrieval information serves as an anchor to guide LLMs during generation. "Emb" is short for embedding, "PE" stands for positional embeddings, and "<R>" denotes the placeholder for retrieval information.

to further exploit retrieval information.

3.3 R^2-Former

Inspired by Li et al. (2023b), we propose the R^2-Former as the trainable module that bridges between retrievers and LLMs. As shown in the right side of Figure 2, R^2-Former is a pluggable Transformer-based model that accepts list-wise features as inputs and outputs retrieval information.

To better comprehend list-wise features from retrievers, we employ an input embedding layer to linearly transform input features into a higher dimension space. Positional embeddings are then added before attention encoding to maintain sequence awareness. Then, a Transformer (Vaswani et al., 2017) encoder is utilized to exploit the input sequences, which uses a self-attention mask where each position's feature can attend to other positions. Formally, for an input list $\{\text{input}_i\}_{i=1}^k$, the process is formulated by:

$$\mathbf{H} = f_{att}\left[f_{\to h_1}\left(\{\text{input}_i\}_{i=1}^k\right) + \mathbf{p}\right], \quad (2)$$

where f_{att} is the Transformer encoder with h_1 hidden dimension, $f_{\to h_1}$ is a linear mapping layer, and $\mathbf{p} \in \mathbb{R}^{k \times h_1}$ represents trainable positional embeddings. The output embeddings $\mathbf{H} \in \mathbb{R}^{k \times h_1}$ thus contain the deeper retrieval information and will be delivered to the LLM's generation.

3.4 Retrieval-Aware Prompting

In the generation process, it is crucial for the LLM to utilize the retrieval information effectively. As shown in the upper part of Figure 2, we introduce a retrieval-aware prompting strategy that injects the retrieval information extracted by R^2-Former into the LLM's generation process.

First, we employ a projection layer to linearly transform the retrieval information into the same dimension as the token embedding layer of the LLM. Formally, this is represented as:

$$\mathbf{E}^R = f_{\to h_2}(\mathbf{H}) = \{\mathbf{e}_i^R\}_{i=1}^k, \quad (3)$$

where $f_{\to h_2}$ is a linear projection layer via an MLP layer, and h_2 is the dimension of LLM's token embedding layer.

Then, we tokenize the query and documents using LLM's tokenizer and convert them into embeddings. For example, a document d is tokenized into $\boldsymbol{t}^d = \{t_j^d\}_{j=1}^{n_d}$, where t_j^d is the j-th token in the document, n_d is the number of tokens in the document d. And the token embeddings can be transformed by a lookup in the token embedding layer. The process can be expressed as:

$$\mathbf{E}^d = f_{emb}\left(\boldsymbol{t}^d\right) = \{\mathbf{e}_j^d\}_{j=1}^{n_d}, \quad (4)$$

where f_{emb} is the token embedding layer of the LLM, and $\mathbf{E}^d \in \mathbb{R}^{n_d \times h_2}$ is the embeddings of document d. A similar process is applied to obtain the query embeddings $\mathbf{E}^q = \{\mathbf{e}_j^q\}_{j=1}^{n_q}$, where n_q is the number of query tokens.

For nuanced analysis of each document, the corresponding retrieval information embeddings are then prepended to the front of each document's embeddings. They are external knowledge and function as an anchor, guiding the LLM to focus on useful documents. The final input embeddings can be arranged as:

$$\mathbf{E}=[\underbrace{\mathbf{e}_1^q,\cdots,\mathbf{e}_{n_q}^q}_{\text{query}},\mathbf{e}_1^R,\underbrace{\mathbf{e}_1^{d_1},\cdots,\mathbf{e}_{n_{d_1}}^{d_1}}_{\text{document}_1},\cdots,\mathbf{e}_k^R,\underbrace{\mathbf{e}_1^{d_k},\cdots,\mathbf{e}_{n_{d_k}}^{d_k}}_{\text{document}_k}], \tag{5}$$

where $\mathbf{e}_i^R$ denotes the retrieval information embedding for the i-th document. In this way, the retrieval information of corresponding document can be well mixed, reducing the burden of the LLM to process all documents. Finally, we can get the responses by:

$$\hat{y}=f_{\mathbf{G}}(\mathbf{E}), \tag{6}$$

where $\hat{y}$ represents the LLM-generated results. Notably, this part simplifies the instruction prompt, and detailed descriptions and prompt templates can be found in Appendix B.

3.5 Training Strategy

As the interdependence of retrieval and generation, we integrate R^2-Former training and LLM alignment into one stage. The joint training allows R^2-Former to better understand list-wise features from the retriever, ensuring retrieval information can be deeply interpreted by the LLM.

For R^2-Former training, we perform a query-document matching (QDM) task that enforces R^2-Former to learn the relevance relationships from list-wise features. In detail, it is a binary classification task that asks to model each document's relevance to the query. The formula for prediction is as follows:

$$\hat{\mathbf{s}}=f_{\to 1}(\mathbf{H})=\{\hat{s}_i\}_{i=1}^k, \tag{7}$$

where $f_{\to 1}$ is a binary classification head that outputs the relevance predictions $\hat{\mathbf{s}}$. Supporting $\mathbf{s}$=$\{s_i\}_{i=1}^k$ are the ground-truth labels for documents, we use cross-entropy as the loss function, defined as:

$$\mathcal{L}_{QDM}(\mathbf{s},\hat{\mathbf{s}})=-\sum_{i=1}^k s_i\log(\hat{s}_i)+(1-s_i)\log(1-\hat{s}_i). \tag{8}$$

For LLM alignment, we utilize the language modeling (LM) task, which involves learning to generate subsequent tokens based on the preceding context and retrieval information. The language modeling loss $\mathcal{L}_{LM}$ aims to maximize the log-likelihood of the tokens, rewarding the LLM for predicting subsequent words correctly.

The joint training involves instruction fine-tuning with a linear combination of QDM and LM tasks. The final loss is expressed as:

$$\mathcal{L}=\mathcal{L}_{QDM}+\mathcal{L}_{LM}. \tag{9}$$

Notably, R^{2}AG offers the flexibility to train the R^2-Former solely while freezing the LLM or to train both together for a deeper understanding of retrieval information. The decision represents a trade-off between lower computational costs and higher accuracy in real-world scenarios.

The R^{2}AG algorithm is detailed in Appendix C.

4 Experiments

4.1 Datasets and Metrics

We evaluate R^{2}AG on five datasets: Natural Questions (NQ) (Kwiatkowski et al., 2019), HotpotQA (Yang et al., 2018), MuSiQue (Trivedi et al., 2021), 2WikiMultiHopQA (2Wiki) (Ho et al., 2020), and DuReader (He et al., 2018). For NQ dataset, we utilize NQ-10, NQ-20, and NQ-30 datasets built by Liu et al. (2023), which contain 10, 20, and 30 total documents, respectively. DuReader is a multiple documents QA version built by Bai et al. (2023b). Detailed introduction and statistics are shown in Appendix D.

Following Mallen et al. (2023); Liu et al. (2023), we adopt accuracy (Acc) as the evaluation metric for NQ datasets. Following Bai et al. (2023b), we adopt accuracy (Acc) and F1 score as evaluation metrics for HotpotQA, MuSiQue, and 2Wiki datasets. For DuReader dataset, we measure performance by F1 score and Rouge (Lin, 2004).

4.2 Baselines

To fully evaluate R^{2}AG, we compared two types of methods: standard RAG using various LLMs, and enhanced RAG using the same foundation LLM.

First, we evaluate standard RAG baselines where LLMs generate responses given the query prepended with retrieved documents. For English datasets, we use several open-source LLMs, including LLaMA2$_{7B}$, LLaMA2$_{13B}$, LLaMA3$_{8B}$ (Touvron et al., 2023), and LongChat1.5$_{7B}$ (Li et al., 2023a). Besides, we adopt ChatGPT (Ouyang et al., 2022) and GPT4 (Achiam et al., 2023) as baselines of closed-source LLMs. For the Chinese dataset, we employ Qwen1.5$_{0.5B}$, Qwen1.5$_{1.8B}$ (Bai et al., 2023a) and InternLM2$_{1.8B}$ (Cai et al., 2024).

Secondly, we experiment with several methods that can enhance RAG, including CoT (Wei et al., 2022), RECOMP (Xu et al., 2023), CRAG (Yan et al., 2024), Self-RAG (Asai et al., 2023), LongLLMLingua (Jiang et al., 2023), and RAFT (Zhang et al., 2024). For NQ-10, HotpotQA, MuSiQue, and 2Wiki datasets, we use LLaMA2$_{7B}$

Methods	NQ-10	NQ-20	NQ-30	HotpotQA		MuSiQue		2Wiki	
	Acc	Acc	Acc	Acc	F1	Acc	F1	Acc	F1
Frozen LLMs									
LLaMA2$_{7B}$	0.3898	-	-	0.2630	0.0852	0.0546	0.0241	0.1205	0.0634
LongChat1.5$_{7B}$	0.6045	0.5782	0.5198	0.5424	0.3231	0.2808	0.1276	0.3882	0.2253
LLaMA3$_{8B}$	0.5141	0.4991	0.5311	0.5901	0.2056	0.2427	0.0891	**0.4723**	0.1952
LLaMA2$_{13B}$	0.7684	-	-	0.3788	0.1000	0.0909	0.0446	0.2405	0.0898
ChatGPT	0.6886	0.6761	0.6347	0.6557	**0.6518**	0.3376	**0.3321**	-	-
GPT4	**0.7759**	**0.7514**	**0.7514**	**0.7673**	0.6026	**0.4853**	0.3270	-	-
CoT	0.4482	0.6026	0.5631	0.2365	0.1028	0.0626	0.0412	0.1627	0.0969
RECOMP	0.0169	0.2222	0.1977	0.2388	0.0265	0.0830	0.0156	0.2666	0.0329
CRAG	0.3974	0.6441	0.6347	0.1194	0.0360	0.0262	0.0047	0.0768	0.0422
LongLLMLingua	0.3635	-	-	0.4174	0.1178	0.1939	0.0477	0.2374	0.0888
R^{2}AG	0.6930	0.7062	0.6704	0.6675	0.3605	0.1864	0.1687	0.3342	**0.3452**
Fine-tuned LLMs									
Self-RAG	0.1883	-	-	0.2475	0.1236	0.0701	0.0378	0.2611	0.1389
RAFT	0.7514	0.8041	0.7307	0.7349	**0.3172**	**0.2529**	0.1502	**0.7555**	0.4869
R^{2}AG+RAFT	**0.8192**	**0.8060**	**0.7458**	**0.7351**	0.3056	0.2295	**0.1533**	0.7444	**0.6351**

Table 1: Main results on four English datasets. Results marked in grey background indicate the performance of foundation LLMs. Results in gray represent the performance of closed-source LLMs. **Results** in bold and results in underlined mean the best and second-best performance among current classified methods.

Methods	DuReader	
	F1	Rouge
Frozen LLMs		
LongChat1.5$_{7B}$	0.0914	0.1181
Qwen1.5$_{0.5B}$	0.1395	0.1656
Qwen1.5$_{1.8B}$	**0.1533**	0.1570
InternLM2$_{1.8B}$	0.1330	0.1391
R^{2}AG	0.1510	**0.1663**
Fine-tuned LLMs		
RAFT	0.2423	**0.2740**
R^{2}AG+RAFT	**0.2507**	0.2734

Table 2: Performance comparison on DuReader dataset.

as the foundation LLM for enhanced RAG methods, which has a maximum context length of 4k tokens. For NQ-20 and NQ-30 datasets, LongChat1.5$_{7B}$ is selected as the foundation LLM, which extends the context window to 32k tokens. For DuReader dataset, Qwen1.5$_{0.5B}$ is the foundation LLM, also with a maximum context length of 32k tokens.

These methods were categorized into two groups – frozen and fine-tuned – based on whether they require training the LLMs.

The implementation details are in Appendix E.

4.3 Main Results

Table 1 and Table 2 provide the main results. We can obtain the following conclusions:

(1) Compared with foundation LLMs using standard RAG, R^{2}AG can significantly increase performance. Even in multi-hot datasets, R^{2}AG improves LLMs' ability for complex reasoning. In DuReader dataset, with a token length of 16k, R^{2}AG remains effective, demonstrating its robustness and efficiency in handling extensive text outputs. These results indicate that R^{2}AG effectively enables LLMs to better understand the retrieval information and boosts their capabilities in handling provided documents. (2) Compared with other LLMs using standard RAG, R^{2}AG generally achieves better performance except for closed-source LLMs. GPT4 shows superior results in most datasets, establishing it as a strong baseline. Notably, R^{2}AG excels ChatGPT in NQ and HotpotQA datasets. Using LLaMA2$_{7B}$ as the foundational LLM, R^{2}AG competes well with LLaMA3$_{8B}$ and LLaMA2$_{13B}$ across most metrics. (3) It is clear that R^{2}AG significantly surpasses other enhanced RAG methods in most results, underscoring the importance of incorporating retrieval information. Although CRAG has a good result in NQ datasets, its perfor-

Methods	NQ-10 LLaMA2$_{7B}$	NQ-20 LongChat1.5$_{7B}$
R^{2}AG	**0.6930**	**0.7062**
w/o r	0.6761 (↓2.45%)	0.6798 (↓3.73%)
w/o γ	0.6723 (↓2.99%)	0.6930 (↓1.87%)
w/o ζ	0.6252 (↓9.78%)	0.6855 (↓2.93%)
w/o $\mathcal{L}_{QDM}$	0.6441 (↓7.07%)	0.7043 (↓0.27%)

Table 3: Ablation studies on NQ-10 and NQ-20 datasets.

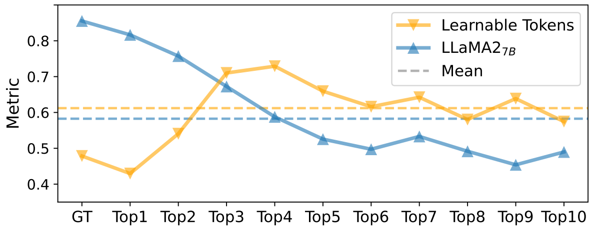

Figure 3: Performance of learnable tokens across different document counts on NQ-10 dataset. "GT" means only retaining ground-true documents.

mance significantly declines in multi-hop datasets. That is because CRAG's simplistic approach of filtering out documents irrelevant to the query can omit crucial connections needed for understanding complex queries. Additionally, our method outperforms compression-based methods (RECOMP and LongLLMLingua). Our case studies reveal their poor performance is mainly because the coordination between the compressors and LLMs tends to result in substantial information loss and even severe hallucinations. (4) RAFT can significantly improve the performance. When combined with R^{2}AG, the results are the best overall, suggesting that a deeper understanding acquired through training benefits generation capabilities.

4.4 Ablation Studies

To demonstrate the effectiveness of R^{2}AG, we create four variants. Specifically, we remove three retrieval features r, γ, ζ, individually. For R^2-Former, we remove the QDM loss $\mathcal{L}_{QDM}$. We conduct the ablation studies on the NQ-10 and NQ-20 datasets, using LLaMA2$_{7B}$ and LongChat1.5$_{7B}$ as foundation LLMs with results shown in Table 3. We can obtain the following observations: First, the performance decreases without any of the three retrieval features, underscoring their effectiveness. The results reveal that utilizing additional retrieval features can help LLMs disentangle irrelevant documents. Secondly, the performance decreases without the QDM loss, showing that the query-document matching task is indeed beneficial for

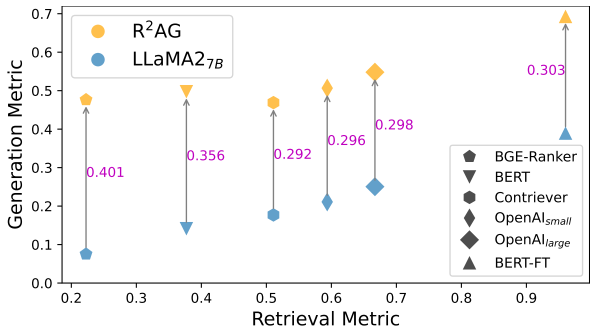

Figure 4: Performance comparison of R^{2}AG with various retrievers on NQ-10 dataset.

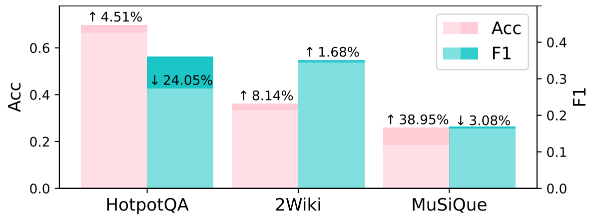

Figure 5: Performance of R^{2}AG$_{7B}$ and R^{2}AG$_{13B}$. Darker parts mean the difference value of R^{2}AG$_{13B}$.

exploiting retrieval information.

To explore the effectiveness of the retrieval-aware prompting strategy, we design an experiment on NQ-10 dataset with various top-k retrieved documents where the retrieval information is set as learnable tokens. This means R^{2}AG only uses these soft prompts without additional features when training and inference. From the results shown in Figure 3, we can find that: (1) When retrieval results are largely relevant, with few or no redundant documents, learnable tokens do not aid the LLM and may instead become redundant information for the generation. (2) As the number of documents increases, it is natural to observe a decline performance. Surprisingly, learnable tokens significantly enhance the performance of the LLM. These findings demonstrate that the retrieval-aware prompting strategy effectively assists LLMs in processing multiple documents, especially when those documents include irrelevant information.

4.5 Discussions

The Impact of Performance of Retrievers and LLMs. As mentioned in Section 1, the quality of retrieved documents can heavily influence the performance of LLMs in RAG. From the main results, R^{2}AG achieves improvements even when the retrieval performance is poor, as observed in MuSiQue and DuReader datasets. Furthermore, we conduct experiments on NQ-10 dataset

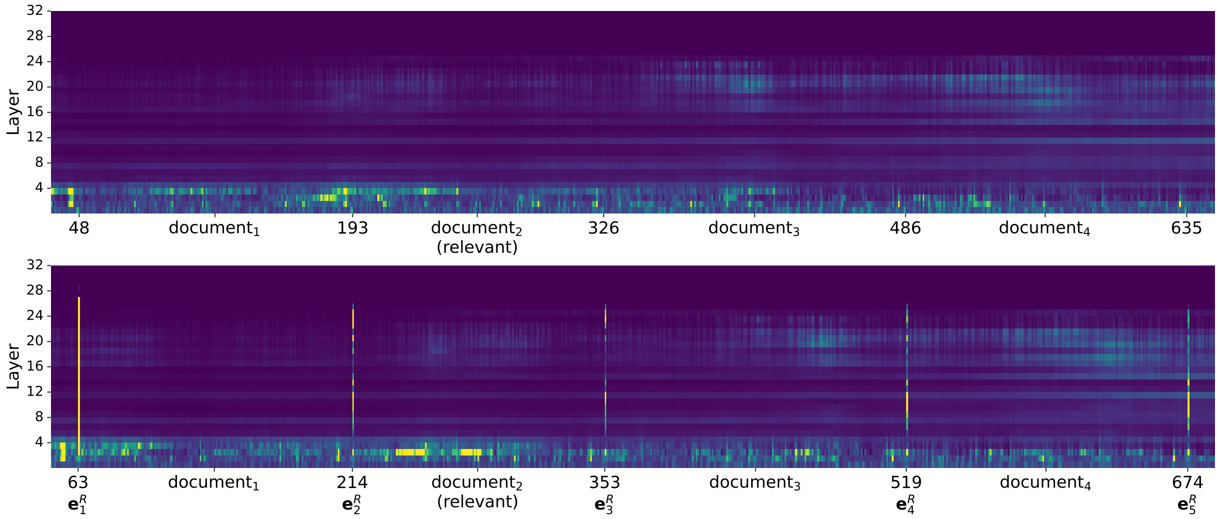

Figure 6: Heatmaps of self-attention distribution of the last token, broken out by token position (X-axis) and layer (Y-axis). Each attention layer comprises 8 heads, and the attention weights are the mean of all the heads. Darker yellow means higher attention weights. $\mathbf{e}_i^R$ is the retrieval information embedding for i-th document.

with five non-trained retrievers, specifically BGE-Reranker (Xiao et al., 2023), BERT (Devlin et al., 2019), Contriever (Izacard et al., 2022), and OpenAI Embedding models (small and large) (Neelakantan et al., 2022), with 1024, 768, 768, 1536, and 3072 dimensions, respectively. Note that OpenAI Embedding models are closed-source. From the results presented in Figure 4, we easily observe that a stronger retriever leads to better performance, both standard RAG and R^{2}AG. Importantly, R^{2}AG significantly enhances the effectiveness of LLMs, even when the retrieval performance is poor.

We conduct experiments on HotpotQA, MuSiQue, and 2Wiki datasets using LLaMA2$_{13B}$ as the foundation LLM. Results shown in Figure 5 indicate that R^{2}AG$_{13B}$ outperforms R^{2}AG$_{7B}$, particularly in the accuracy metric. Specially, there is a decline performance in F1 scores for HotpotQA and MuSiQue datasets. We find this primarily because larger LLMs usually tend to output longer answers with explanations (the average response token count in HotpotQA dataset for R^{2}AG$_{7B}$ is 37.44, compared to 49.71 for R^{2}AG$_{13B}$). This tendency also can be observed from the results of ChatGPT and GPT4.

These results reveal that both a stronger LLM and a more effective retriever lead to better performance, validating that R^{2}AG is a genetic method that can be efficiently applied in various scenarios.

The Effect of Retrieval Information. For a deeper and more intuitive exploration of how retrieval information improves LLMs' generation, we present a visualization of the self-attention distribution in R^{2}AG compared with standard RAG. In detail, we analyze a case in NQ-10 dataset in which the foundation LLM is LLaMA2$_{7B}$. We extract the self-attention weights in different layers from LLM's outputs and visualize the last token's attention distribution for other tokens. The relevant document is ranked in position 2 in our selected case, while the 1st document is potentially confusing. For a clear illustration, we select attention distribution for tokens in top-4 documents. From Figure 6, it is evident that the retrieval information receives higher attention scores even in deeper layers, and the relevant document can get more attention within 1-4 layers. That means the retrieval information effectively acts as an anchor, guiding the LLM to focus on useful documents.

Further discussions and case studies are available in Appendix F and G.

5 Conclusion and Future Work

This paper proposed a novel enhanced RAG framework named R^{2}AG to bridge the semantic gap between the retrievers and LLMs. By incorporating retrieval information from retrievers into LLMs' generation process, R^{2}AG captures a comprehensive understanding of retrieved documents. Experimental results show that R^{2}AG outperforms other competitors. In addition, the robustness and effectiveness of R^{2}AG are further confirmed by detailed analysis. In future work, more retrieval features could be applied to R^{2}AG framework.

Limitations

The following are the limitations associated with R^2AG: First, R^2AG depends on the semantic representations modeled by encoder-based retrievers. The suitability of other types of retrievers, such as sparse and cross-encoder retrievers, requires further exploration. Secondly, as mentioned in Section 4.5, R^2AG relies on the ability of the foundation LLM, and more powerful closed-source LLMs may not be compatible with R^2AG. Thirdly, there may be other informative features besides the three retrieval features - relevance, precedent similarity, and neighbor similarity scores. Lastly, R^2AG is evaluated on five datasets, of which relevant documents are provided. However, situations where no relevant documents are available need to be considered. R^2AG may benefit from integrating techniques like self-RAG to better handle such situations.

Ethics Statement

LLMs can generate incorrect and potentially harmful answers. Our proposed method aims to alleviate this issue by providing LLMs with retrieved documents and retrieval information, thereby enhancing LLMs' capability of generation. In the development and execution of our work, we strictly adhered to ethical guidelines established by the broader academic and open-source community. All the datasets and models used in this work are publicly available. No conflicts of interest exist for any of the authors involved in this work.

References

OpenAI Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, and et al. 2023. Gpt-4 technical report.

Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. 2023. Self-rag: Learning to retrieve, generate, and critique through self-reflection. *ArXiv*, abs/2310.11511.

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenhang Ge, Yu Han, Fei Huang, Binyuan Hui, Luo Ji, Mei Li, and et al. 2023a. Qwen technical report. *ArXiv*, abs/2309.16609.

Yushi Bai, Xin Lv, Jiajie Zhang, Hong Lyu, Jiankai Tang, Zhidian Huang, Zhengxiao Du, Xiao Liu, Aohan Zeng, Lei Hou, Yuxiao Dong, Jie Tang, and Juanzi Li. 2023b. Longbench: A bilingual, multitask benchmark for long context understanding. *ArXiv*, abs/2308.14508.

Scott Barnett, Stefanus Kurniawan, Srikanth Thudumu, Zach Brannelly, and Mohamed Abdelrazek. 2024. Seven failure points when engineering a retrieval augmented generation system. *ArXiv*, abs/2401.05856.

Parishad BehnamGhader, Santiago Miret, and Siva Reddy. 2022. Can retriever-augmented language models reason? the blame game between the retriever and the language model. *ArXiv*, abs/2212.09146.

Zheng Cai, Maosong Cao, Haojiong Chen, Kai Chen, Keyu Chen, Xin Chen, Xun Chen, Zehui Chen, Zhi Chen, Pei Chu, Xiao wen Dong, and et al. 2024. Internlm2 technical report. *ArXiv*, abs/2403.17297.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Matthijs Douze, Alexandr Guzhva, Chengqi Deng, Jeff Johnson, Gergely Szilvasy, Pierre-Emmanuel Mazar'e, Maria Lomeli, Lucas Hosseini, and Herv'e J'egou. 2024. The faiss library. *ArXiv*, abs/2401.08281.

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.

Wei He, Kai Liu, Jing Liu, Yajuan Lyu, Shiqi Zhao, Xinyan Xiao, Yuan Liu, Yizhong Wang, Hua Wu, Qiaoqiao She, Xuan Liu, Tian Wu, and Haifeng Wang. 2018. DuReader: a Chinese machine reading comprehension dataset from real-world applications. pages 37–46.

Xiaoxin He, Yijun Tian, Yifei Sun, N. Chawla, Thomas Laurent, Yann LeCun, Xavier Bresson, and Bryan Hooi. 2024. G-retriever: Retrieval-augmented generation for textual graph understanding and question answering. *ArXiv*, abs/2402.07630.

Xanh Ho, Anh-Khoa Duong Nguyen, Saku Sugawara, and Akiko Aizawa. 2020. Constructing a multi-hop QA dataset for comprehensive evaluation of reasoning steps. pages 6609–6625.

J. Edward Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *ArXiv*, abs/2106.09685.

Gautier Izacard, Patrick Lewis, Maria Lomeli, Lucas Hosseini, Fabio Petroni, Timo Schick, Jane A. Yu, Armand Joulin, Sebastian Riedel, and Edouard Grave. 2022. Atlas: Few-shot learning with retrieval augmented language models. *ArXiv*, abs/2208.03299.

Huiqiang Jiang, Qianhui Wu, Xufang Luo, Dongsheng Li, Chin-Yew Lin, Yuqing Yang, and Lili Qiu. 2023. Longllmlingua: Accelerating and enhancing llms in long context scenarios via prompt compression. *ArXiv*, abs/2310.06839.

Nikhil Kandpal, H. Deng, Adam Roberts, Eric Wallace, and Colin Raffel. 2022. Large language models struggle to learn long-tail knowledge. In *International Conference on Machine Learning*.

Zixuan Ke, Weize Kong, Cheng Li, Mingyang Zhang, Qiaozhu Mei, and Michael Bendersky. 2024. Bridging the preference gap between retrievers and llms. *ArXiv*, abs/2401.06954.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980.

Yuma Koizumi, Yasunori Ohishi, Daisuke Niizumi, Daiki Takeuchi, and Masahiro Yasuda. 2020. Audio captioning using pre-trained large-scale language model guided by audio-based similar caption retrieval. *ArXiv*, abs/2012.07331.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur P. Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, and et al. 2019. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:453–466.

Patrick Lewis, Ethan Perez, Aleksandara Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Kuttler, Mike Lewis, and et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *ArXiv*, abs/2005.11401.

Dacheng Li, Rulin Shao, Anze Xie, Ying Sheng, Lianmin Zheng, Joseph E. Gonzalez, Ion Stoica, Xuezhe Ma, and Hao Zhang. 2023a. How long can open-source LLMs truly promise on context length?

Dongxu Li, Junnan Li, Hung Le, Guangsen Wang, Silvio Savarese, and Steven C. H. Hoi. 2022. Lavis: A library for language-vision intelligence. *ArXiv*, abs/2209.09019.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven C. H. Hoi. 2023b. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International Conference on Machine Learning*.

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

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2023. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173.

Yixiao Ma, Qingyao Ai, Yueyue Wu, Yunqiu Shao, Yiqun Liu, M. Zhang, and Shaoping Ma. 2022. Incorporating retrieval information into the truncation of ranking lists for better legal search. *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*.

Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, and Hannaneh Hajishirzi. 2023. When not to trust language models: Investigating effectiveness of parametric and non-parametric memories. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9802–9822, Toronto, Canada. Association for Computational Linguistics.

Arvind Neelakantan, Tao Xu, Raul Puri, Alec Radford, Jesse Michael Han, Jerry Tworek, Qiming Yuan, Nikolas A. Tezak, Jong Wook Kim, and et al. 2022. Text and code embeddings by contrastive pre-training. *ArXiv*, abs/2201.10005.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, and et al. 2022. Training language models to follow instructions with human feedback. *ArXiv*, abs/2203.02155.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, and et al. 2019. Pytorch: An imperative style, high-performance deep learning library. *ArXiv*, abs/1912.01703.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Weijia Shi, Sewon Min, Michihiro Yasunaga, Minjoon Seo, Rich James, Mike Lewis, Luke Zettlemoyer, and Wen tau Yih. 2023. Replug: Retrieval-augmented black-box language models. *ArXiv*, abs/2301.12652.

Hugo Touvron, Louis Martin, Kevin R. Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, and et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *ArXiv*, abs/2307.09288.

H. Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. 2021. Musique: Multi-hop questions via single-hop question composition. *Transactions of the Association for Computational Linguistics*, 10:539–554.

Ashish Vaswani, Noam M. Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Neural Information Processing Systems*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Ed Huai hsin Chi, F. Xia, Quoc Le, and Denny Zhou. 2022. Chain of thought prompting elicits reasoning in large language models. *ArXiv*, abs/2201.11903.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, and et al. 2020. Transformers: State-of-the-art natural language processing. pages 38–45.

Siye Wu, Jian Xie, Jiangjie Chen, Tinghui Zhu, Kai Zhang, and Yanghua Xiao. 2024. How easily do irrelevant inputs skew the responses of large language models? *ArXiv*, abs/2404.03302.

Shitao Xiao, Zheng Liu, Peitian Zhang, and Niklas Muennighoff. 2023. C-pack: Packaged resources to advance general chinese embedding. *ArXiv*, abs/2309.07597.

Fangyuan Xu, Weijia Shi, and Eunsol Choi. 2023. Recomp: Improving retrieval-augmented lms with compression and selective augmentation. *ArXiv*, abs/2310.04408.

Shicheng Xu, Liang Pang, Jun Xu, Huawei Shen, and Xueqi Cheng. 2024. List-aware reranking-truncation joint model for search and retrieval-augmented generation. In *The Web Conference*.

Shi-Qi Yan, Jia-Chen Gu, Yun Zhu, and Zhen-Hua Ling. 2024. Corrective retrieval augmented generation. *ArXiv*, abs/2401.15884.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W. Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. 2018. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. In *Conference on Empirical Methods in Natural Language Processing*.

Michihiro Yasunaga, Armen Aghajanyan, Weijia Shi, Rich James, Jure Leskovec, Percy Liang, Mike Lewis, Luke Zettlemoyer, and Wen tau Yih. 2023. Retrieval-augmented multimodal language modeling. *ArXiv*, abs/2211.12561.

Fuda Ye and Shuangyin Li. 2024. Milecut: A multi-view truncation framework for legal case retrieval. In *The Web Conference*.

Tianjun Zhang, Shishir G. Patil, Naman Jain, Sheng Shen, Matei A. Zaharia, Ion Stoica, and Joseph E. Gonzalez. 2024. Raft: Adapting language model to domain specific rag. *ArXiv*, abs/2403.10131.

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.

Wayne Xin Zhao, Jing Liu, Ruiyang Ren, and Ji rong Wen. 2022. Dense text retrieval based on pretrained language models: A survey. *ACM Transactions on Information Systems*.

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, and et al. 2023. A survey of large language models. *ArXiv*, abs/2303.18223.

Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. 2023a. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *ArXiv*, abs/2304.10592.

Yutao Zhu, Huaying Yuan, Shuting Wang, Jiongnan Liu, Wenhan Liu, Chenlong Deng, Zhicheng Dou, and Ji rong Wen. 2023b. Large language models for information retrieval: A survey. *ArXiv*, abs/2308.07107.

A Retrieval Feature Extraction Details

Formally, the relevance between the query and the i-th document is calculated as:

$$r_i = \mathrm{sim}\left(\mathbf{x}^q, \mathbf{x}_i^d\right), \tag{10}$$

where sim is a similarity function such as dot product or cosine similarity, $\mathbf{x}^q$ and $\mathbf{x}_i^d$ are representations of query and i-th document, respectively.

The precedent similarity computes the similarity score between case representation and its precedent-weighted representations in the ranking list as follows:

$$\gamma_i{=}\mathrm{sim}\left(\mathbf{x}_i^d, \sum_{j=1}^{i-1} w_j \cdot \mathbf{x}_j^d\right), w_j{=}\frac{\exp(r_j)}{\sum_{\ell=1}^{k}\exp(r_\ell)}, \tag{11}$$

where γ_i is the precedent similarity between i-th document and its precedents in the ranking list, and r_i is relevance between the query and i-th document.

Neighbor similarity represents the average similarity of i-th document to its adjacent documents. Specifically, the neighbor similarity of a case in the ranking list is given by:

$$\zeta_i = \begin{cases} \mathrm{sim}(\mathbf{x}_1^d, \mathbf{x}_2^d), & i = 1 \\ [\mathrm{sim}(\mathbf{x}_{i-1}^d, \mathbf{x}_i^d) + \mathrm{sim}(\mathbf{x}_i^d, \mathbf{x}_{i+1}^d)]/2, & i \in [2, k) \\ \mathrm{sim}(\mathbf{x}_{k-1}^d, \mathbf{x}_k^d), & i = k \end{cases}, \tag{12}$$

where ζ_i represents the average similarity of i-th document to its adjacent documents. Such that we can get the list-wise features among documents.

Datasets	Language	# Query	# Train/Test	# Tokens	# Rel/Docs	MAP
NQ-10	English	2655	2124/531	~2k	1/10	0.9602
NQ-20	English	2655	2124/531	~4k	1/20	0.9287
NQ-30	English	2655	2124/531	~6k	1/30	0.9215
HotpotQA	English	97852	90447/7405	~2k	2.36/10	0.9138
MuSiQue	English	22355	19938/2417	~3k	2.37/20	0.5726
2Wiki	English	180030	167454/12576	~2k	2.42/10	0.9637
DuReader	Chinese	200	160/40	~16k	1.82/20	0.7169

Table 4: Statistics of datasets. "# Rel/Docs" denotes the number of relevant documents and the total number of documents for each query. "MAP" represents the Mean Average Precision, a common retrieval metric.

B Prompt Templates

In R^{2}AG, retrieval information, we append k special tokens ("<R>") in front of each document to facilitate the incorporation of retrieval information. These tokens do not carry meaningful semantics but serve as placeholders for the retrieval information within the prompt. This special token facilitates the integration of retrieval information into the generation process.

Table 5 shows the prompt templates for R^{2}AG and other baselines. The prompt templates of DuReader dataset can be found in our source code.

C R^{2}AG Algorithm

Algorithm 1 is the implementation of R^{2}AG.

Algorithm 1: R^{2}AG

Input : Query q, retrieved documents $\mathcal{D}=\{d_i\}_{i=1}^k$.
Output : Answer y.
Initialize R^2-Former parameters Θ_R, retriever $f_\mathbf{R}$, and LLM $f_\mathbf{G}$.
Get query and documents semantic representation ($\mathbf{x}^q$ and $\mathbf{x}^d$) from retriever $f_\mathbf{R}$.
`// Retrieval feature extraction`
for d_i *in* D **do**
  Calculate input$_i$. `// {Eq.10, 11, 12}`
end
`// R2-Former`
Calculate hidden states $\mathbf{H}$; `// {Eq. 2}`
`// Retrieval-aware prompting`
Calculate hidden state $\mathbf{E}^R$; `// {Eq. 3}`
Embedding lookup for text tokens; `// {Eq. 4}`
Fuse into final input embeddings $\mathbf{E}$; `// {Eq. 5}`
Get the response y from LLM $f_\mathbf{G}$; `// {Eq. 6}`
`// Training strategy`
if *is training* **then**
  Project the relevance scores S; `// {Eq. 7}`
  Calculate $\mathcal{L}_{QDM}$; `// {Eq. 8}`
  Calculate $\mathcal{L}_{LM}$;
  Calculate final loss $\mathcal{L}$; `// {Eq. 9}`
  Update parameters.
end
return y

D Dataset Introduction

We conduct evaluations on five datasets, including:

Natural Questions (NQ) (Kwiatkowski et al., 2019) is developed from Google Search and contains questions coupled with human-annotated answers extracted from Wikipedia. Further, Liu et al. (2023) collect $k-1$ distractor documents from Wikipedia that do not contain the answers, where k is the total document number for each question. This dataset has three versions: NQ-10, NQ-20, and NQ-30, with total document numbers of 10, 20, and 30, respectively.

HotpotQA (Yang et al., 2018) is a well-known multi-hop question answering dataset based on Wikipedia. This dataset involves questions requiring finding and reasoning over multiple supporting facts from 10 documents. There are two reasoning types of questions: bridging and comparison.

MuSiQue (Trivedi et al., 2021) has questions that involve 2-4 hops and six types of reasoning chains. The dataset is constructed through a bottom–up process by carefully selecting and composing single-hop questions. The final answer to each question in the distractor setting is extracted from 20 documents.

2WikiMultiHopQA (2Wiki) (Ho et al., 2020) consists of up to 5-hop questions, each associated with 10 documents. Unlike HotpotQA, this dataset needs to evaluate the interpretability of models not only with supporting evidence but also with entity-relation tuples.

DuReader (He et al., 2018) is a Chinese dataset developed based on Baidu Search and Baidu Zhidao. To adapt it for assessing long context ability, for each question, Bai et al. (2023b) arbitrarily select several documents from the total corpus as distractors until each question is associated with 20 candidate documents.

Detailed statistics can be found in Table 4.

E Implementation Details

Unlike some works (Li et al., 2023b; Zhu et al., 2023a) built on LAVIS (Li et al., 2022), we completely implement R^{2}AG on PyTorch (Paszke et al., 2019) and Transformers (Wolf et al., 2020) libraries for easy usage.

For the retrieval task, we utilize the Sentence-Transformer (Reimers and Gurevych, 2019) to fine-tune a BERT (Devlin et al., 2019) model as the retriever, which is a siamese dual encoder with shared parameters. The models "`bert-base-uncased`" and "`bert-base-chinese`" are used for English datasets and the Chinese dataset, respectively. All retrievers adopt default hyper-parameter settings with 768 embedding dimensions. Cosine similarity is employed as the scoring function for retrieval and feature extraction. The retrieval performance across datasets is shown in Table 4. Contrary to some works (Liu et al., 2023; Jiang et al., 2023) that artificially place ground truth documents in fixed positions, this paper considers that candidate documents are ranked by the retriever to simulate real-world scenarios.

For R^2-Former, we determine the learning rate as 2e-4 and dropout as 0.1. The number of attention heads and hidden size in Transformer encoder are 4 and 256, respectively. Adam (Kingma and Ba, 2014) is adopted as the optimization algorithm.

For LLMs, all methods use default settings and adopt greedy decoding for fair comparison. The ChatGPT version is "`gpt-3.5-turbo-0125`" with a 16k context window size, and the GPT4 version is "`gpt-4-turbo-2024-04-09`" with a 128k context window size. In CRAG, the retrieval evaluator only triggered `{Correct, Ambiguous}` actions to next knowledge refinement process as there are at least one relevant document in retrieval results. In the RAFT method, we employ LoRA (Hu et al., 2021) to effectively fine-tune LLMs, with LoRA rank set at 16, alpha at 32, and dropout at 0.1.

F Further Discussion

Different Document Number Figure 7 displays the performance comparison with different document numbers on NQ-30 dataset using LongChat1.5$_{7B}$. For both models, there is a notable trend where the performance metric generally decreases as the number of documents increases. However, this decline is steeper for LongChat1.5$_{7B}$ compared to R^{2}AG. It indicates that R^{2}AG is particularly effective in scenarios where the number of documents can vary.

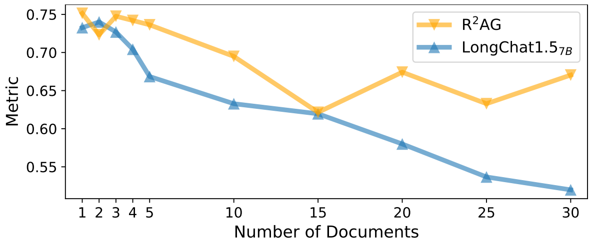

Figure 7: Performance comparison with different document numbers on NQ-30 dataset.

Efficiency of R^{2}AG In default settings, R^{2}AG is designed to train only the R^2-Former and the projection layer. The time complexity primarily depends on the number of the retrieval documents (k), the number of features (3 in our work), and the size of hidden layer in Transformer (h_1). In Table 6, we analyze the parameter count, including trainable and total parameters. The results indicate that the R^2-Former only has few trainable parameters, and resource requirements of R^{2}AG are substantially low. In the inference stage, R^{2}AG incorporates retrieval information into LLMs' embedding space, which theoretically increases the embeddings with k tokens. To provide a practical comparison, we calculate the inference time on 2Wiki dataset using a single A800 GPU. The results indicate that the inference time increases by only 0.8%, making R^{2}AG suitable for real-world applications. In practical scenarios, retrieved documents can be pre-encoded and stored in a vector database using tools like Faiss (Douze et al., 2024) for efficient inference.

G Case Studies

Table 7 presents a case study on HotpotQA dataset, comparing R^{2}AG with compression-based methods. In this case, the 1st and 3rd documents are relevant, and the 2nd document is a hard distractor. Only R^{2}AG successfully provides the correct answer. In contrast, LongLLMLingua compresses the documents but retains their original order, which leads to ineffective outputs. Meanwhile, RECOMP confuses the 2nd and 3rd documents, resulting in hallucinations. More cases compared R^{2}AG with standard RAG are shown in Table 8.

Methods	Prompts
RAG	Write a high-quality answer for the given question using only the provided search results (some of which might be irrelevant). Only give me the answer and do not output any other words. [1]{#d_1} [2]{#d_2} ... [k]{#d_k} Only give me the answer and do not output any other words. Question: {#q} Answer:
CoT	Write a high-quality answer for the given question using only the provided search results (some of which might be irrelevant). Only give me the answer and do not output any other words. [1]{#d_1} [2]{#d_2} ... [k]{#d_k} Only give me the answer and do not output any other words. Question: {#q} Let's think it step by step.
Comps	Write a high-quality answer for the given question using only the provided search results (some of which might be irrelevant). Only give me the answer and do not output any other words. {#*Compressed documents*} Only give me the answer and do not output any other words. Question: {#q} Answer:
R^{2}AG	Write a high-quality answer for the given question using only the provided search results (some of which might be irrelevant). Only give me the answer and do not output any other words. The similarity information is provided in front of search results. [1]similarity: <R>{#d_1} [2]similarity: <R>{#d_2} ... [k]similarity: <R>{#d_k} Only give me the answer and do not output any other words. Question: {#q} Answer:

Table 5: Prompt templates of different methods. "Comps" means compression-based methods, including RECOMP and LongLLMLingua. "<R>" is the placeholder for retrieval information.

Methods	# Trainable Params	# Total Params	Inference Time(k=10)
LLaMA2$_{7B}$	0	6.738B	3.154s
R^2-Former	1.319M	1.319M	0.499ms
R^{2}AG	2.372M	6.741B	3.180s($\uparrow$ 0.8%)

Table 6: Efficiency of R^{2}AG.

Case	**Question**:Bordan Tkachuk was the CEO of a company that provides what sort of products? **Relevant Documents**: 1st, 3rd documents **Gold Answer**: IT products and services
R^2AG	**Input Documents** **[1]**similarity: <R>(Title: Bordan Tkachuk) Bordan Tkachuk ( ) is a British business executive, the former CEO of Viglen, also known from his appearances on the BBC-produced British version of "The Apprentice," interviewing for his boss Lord Sugar. **[2]**similarity: <R>(Title: BankUnited) BankUnited, Inc., with total consolidated assets of $27.9 billion at December 31, 2016, is a bank holding company with one wholly owned subsidiary, BankUnited, collectively, the Company. BankUnited, a national banking association headquartered in Miami Lakes, Florida, provides a full range of banking services to individual and corporate customers through 94 banking centers located in 15 Florida counties and 6 banking centers in the New York metropolitan area. The Bank also provides certain commercial lending and deposit products on a national platform. The Company endeavors to provide, through experienced lending and relationship banking teams, personalized customer service and offers a full range of traditional banking products and services to both commercial and retail customers. **[3]**similarity: <R>(Title: Viglen) Viglen Ltd provides IT products and services, including storage systems, servers, workstations and data/voice communications equipment and services. **[4]**similarity: <R>(Title: Cardinal Health) Cardinal Health, Inc. is a Fortune 500 health care services company based in Dublin, Ohio. The company specializes in distribution of pharmaceuticals and medical products, serving more than 100,000 locations. The company also manufactures medical and surgical products, including gloves, surgical apparel and fluid management products. In addition, it operates the nation's largest network of radiopharmacies. Cardinal Health provides medical products to over 75 percent of hospitals in the United States. In December 2013, it was announced that Cardinal Health would team up with CVS Caremark, which would form the largest generic drug sourcing operation in the United States. The venture was named Red Oak Sourcing and began operations in July 2014. **[5]**similarity: <R>(Title: Kingstone Companies) Kingstone Companies, Inc., which has its headquarters ... **[6]**similarity: <R>(Title: Compass Minerals) Compass Minerals International, Inc is a United States ... **[7]**similarity: <R>(Title: Jonathan Michael Ansell) Jonathan M. Ansell (born August 13, 1950 ... **[8]**similarity: <R>(Title: Terren Peizer) Terren Scott Peizer dubbed the "Zelig of Wall Street" ... **[9]**similarity: <R>(Title: Owens Minor) Owens & Minor () is a Fortune 500 company based in ... **[10]**similarity: <R>(Title: Clean Power Finance) Clean Power Finance, headquartered in San Francisco ... **LLM's Response** IT products and services, including storage systems, servers, workstations and data/voice communications equipment and services.
LongLLMLingua	**Compressed Prompt** (Title: Bordan Tkachuk) Bordan Tkachuk ( ) is a British business executive, the former CEO of Viglen, also known from his appearances on the BBC-produced British version of "The Apprentice," interviewing for his boss Lord Sugar. (Title: Cardinal Health) Cardinal Health, Inc. is a Fortune 500 health care services company based in Dublin, Ohio. The company specializes in distribution of pharmaceuticals and medical products, serving more than 100,000 locations. The company also manufactures medical and surgical products, including gloves, surgical apparel and fluid management products. In addition, it operates the nation's largest network of radiopharmacies. Cardinal Health provides medical products to over 75 percent of hospitals in the United States. In December 2013, it was announced that Cardinal Health would team up with CVS Caremark, which would form the largest generic drug sourcing operation in the United States. The venture was named Red Oak Sourcing and began operations in July 2014. (Title Clean Power Finance) Clean Power Finance headquartered in San Francisco, California, is a financial services and software company for the residential solar industry. Clean Power Finance operates the CPF Market, an online business-to-business platform that connects institutional investors and lenders with residential solar professionals who need solar finance products to grow their businesses. Clean Power Finance provides the solar industry with CPF Tools, a solar sales, quoting and design software-as-a-Service (SaaS) solution: qualified residential solar channel partners access finance products through the software. Third-party investors create solar project finance funds; Clean Power Finance provides origination, underwriting and asset management services to the fund investors and markets investor capital to solar professionals as residential finance products, including solar leases and power purchase agreements (PPAs). (: Viglen) Viglen IT products and services, including storage systems, servers, workst and/voice communications and services. Title: Ow amp; Minor) Owens Minor () Fortune 500 company based Mechanicsville, Virginia, just of Richmond Owens Inc is health logistics specializing ining of healthcare products for hospitals Owens andťagline is "Connect the World of Medical Product to Point of Care". The company supply chain services to healthcare providers and manufacturers healthcareable supplies as well devices and impl. Owens Minor logistics operations the United States Europe, where three quarters healthcare spending occurs. Ow Minor' customers span the healthcare market independent hospitals large integrated healthcare networks as well group purchasing organizations, healthcare products manufacturers, the federal government. Owens & Minor is headquartered in Mechanicsville, Virginia and has annualized reven exceeding $9 billion :stone Compies) Kingstone Companies,., which its headquarters in Kingston New York, property and casualty insurance ... :Un) BankUn, Inc., with total consolidated assets $2. billion at December 3 2016, is a bank holding wh owned subsidiary ... : Compass Minerals) Compass Minerals International, Inc a United public company that its subsidiaries is leading producer ... **LLM's Response** *No valid answer.*
RECOMP	**Compressed Prompt** Bordan Tkachuk is a British business executive. He was the CEO of Viglen. He was also known from his appearances on the BBC-produced British version of "The Apprentice," interviewing for his boss Lord Sugar. The company provides a variety of products and services. It offers IT products and services, including storage systems, servers, workstations, and data/voice communications equipment and services. The company also provides health care services, indicating a focus on the medical industry. The company manufactures medical and surgical products, including gloves, surgical apparel and fluid management products. ... **LLM's Response** Answer: health care services company

Table 7: Case studies of R^2AG and compression-based methods.

Case	**Question**:When did the animated series Kent Scott wrote end after beginning in September of 2002 on "Nick on CBS"? **Relevant Documents**: 1st, 4th documents **Gold Answer**: November
R^{2}AG	**Input Documents** **[1]**similarity: <R>(Title: Hawaii Five-0 (2010 TV series)) Hawaii Five-0 is an American action police procedural television series, which premiered on Monday, September 20, 2010 on CBS. The series is a reboot of the original series, which aired on CBS from 1968 to 1980. Like the original, it follows an elite state police task force set up to fight crime in the state of Hawaii. The series is produced by K/O Paper Products and 101st Street Television in association with CBS Productions, originally an in-name-only unit of but folded into CBS Television Studios, which has produced the series since the beginning of season three. On March 25, 2016, CBS renewed the series for a seventh season, which premiered on September 23, 2016. On March 23, 2017, CBS renewed the series for an eighth season, which premiered on September 29, 2017. **[2]**similarity: <R>(Title: Pelswick) Pelswick is an animated television series co-produced ... **[3]**similarity: <R>(Title: Kenn Scott) Kenn Scott is a Toronto-based screenwriter noted for ... **[4]**similarity: <R>(Title: Scott McCord) William Scott McCord (born April 19, 1971) is a Canadian actor, voice actor, musician, and composer currently based in Brooklyn. A versatile performer, he has appeared in the films "16 Blocks", "Shoot Ém Up", and the animated feature "The Nut Job". On television, he is best known for his voice over work in popular animated series. He plays Dan Kuso in "Bakugan Battle Brawlers", Tetsuya Watarigani in "Beyblade Metal Fusion", Owen and Trent in Fresh Animationś "Total Drama" series, Skull Boy in "Ruby Gloom", McGee in "Camp Lakebottom", and Jake in the animated TVOKids/Nickelodeon television series, "PAW Patrol". He is also Co-Artistic Director of Toronto-based Criminal Theatre, along with playwright and actor Rosa Laborde. In 2016, he won the Canada Screens Award for Best Performance in an Animated Series for his role as Squidgy on Guruś Justin Time. In 2002 he was nominated for a Dora Mavor Moore Award with Best Performance by a Male in Independent Theatre for his performance in Jacob Richmondś The Qualities of Zero. He has released two albums, the solo effort Blues For Sunshine (2009) and Scott McCord and the Bonafide Truth (2012). The latter is also the name of the soul/jazz/blues/rock eight piece band that was created for touring the first album. Scott McCord and the Bonafide Truth were nominated for Best Artist of the Year at the 2010 Maple Blues Awards. **[5]**similarity: <R>(Title: Project G.e.e.K.e.R.) Project G.e.e.K.e.R. is an animated television ... **[6]**similarity: <R>(Title: Dora the Explorer) Dora the Explorer is an American educational ... **[7]**similarity: <R>(Title: Duane Capizzi) Duane Capizzi is an American writer and television ... **[8]**similarity: <R>(Title: Horseland (TV series)) Horseland is an American animated series ... **[9]**similarity: <R>(Title: Henry Gilroy) Henry Gilroy is an American television screenwriter ... **[10]**similarity: <R>(Title: David W. Zucker) David W. Zucker is President of Television ... **LLM's Response** November of that year.
LLaMA2$_{7B}$	**Input Documents** *As above but without retrieval information.* **LLM's Response** Pelswick
Case	**Question**:When was the female co-founder of the organization that published the "Daily Word" devotional born? **Relevant Documents**: 2nd, 3rd documents **Gold Answer**: August 6, 1845
R^{2}AG	**Input Documents** **[1]**similarity: <R>(Title: The Word For Today) The Word For Today (known as The Word For You Today in some countries) ... **[2]**similarity: <R>(Title: Unity Church) Unity, known informally as Unity Church, is a New Thought Christian organization that publishes the "Daily Word" devotional publication. It describes itself as a "positive, practical Christianity" which "teach[es] the effective daily application of the principles of Truth taught and exemplified by Jesus Christ" and promotes "a way of life that leads to health, prosperity, happiness, and peace of mind." **[3]**similarity: <R>(Title: Myrtle Fillmore) Mary Caroline "Myrtle" Page Fillmore (August 6, 1845 - October 6, 1931) was co-founder of Unity, a church within the New Thought Christian movement, along with her husband Charles Fillmore. Prior to that time, she worked as a schoolteacher. **[4]**similarity: <R>(Title: Daily Word) The Daily Word is a daily inspirational message provided ... **[5]**similarity: <R>(Title: My Utmost for His Highest) My Utmost for His Highest is a daily devotional by ... **[6]**similarity: <R>(Title: Our Daily Bread Ministries) Our Daily Bread Ministries (formerly RBC Ministries) is ... **[7]**similarity: <R>(Title: Daily Light on the Daily Path) Daily Light on the Daily Path or Daily Light is ... **[8]**similarity: <R>(Title: Our Daily Bread) Our Daily Bread is a devotional calendar-style booklet ... **[9]**similarity: <R>(Title: The Clear Word) The Clear Word, originally published in March 1994 as ... **[10]**similarity: <R>(Title: Weili Dai) Weili Dai () is a Chinese-born American businesswoman. She is the Director, ... **LLM's Response** August 6, 1845"
LLaMA2$_{7B}$	**Input Documents** *As above but without retrieval information.* **LLM's Response** The female co-founder of the organization that published the "Daily Word" devotional was born on 1952-09-20.
Case	**Question**:What pop band features an Argentine actress that appeared in the television series "Chiquititas"? **Relevant Documents**: 1st, 4th documents **Gold Answer**: Erreway
R^{2}AG	**Input Documents** **[1]**similarity: <R>(Title: Luisana Lopilato) Luisana Lopilato] (born May 18, 1987) is an Argentine actress and model. She has appeared in the television series "Chiquititas", "Rebelde Way", "Alma Pirata", "Casados con Hijos" and "Atracción x4". **[2]**similarity: <R>(Title: Celeste Cid) María Celeste Cid (] ; born 19 January 1984) is an ... **[3]**similarity: <R>(Title: Nadia Di Cello) Nadia Mariel Di Cello (] ; born January 20, 1989 ... **[4]**similarity: <R>(Title: Camila Bordonaba) Camila Bordonaba Roldán (] ; born September 4, 1984 in El Palomar, Buenos Aires, Argentina) (nicknamed Cami or Cato) is an Argentine actress, singer-songwriter, dancer, musician, Theatre director and former occasional model. She is best known for her roles in "Chiquititas", "Rebelde Way", "Son de Fierro" and "Atracción x4" and as a member of Argentine band Erreway. **[5]**similarity: <R>(Title: Milagros Flores) María de los Milagros Flores (born 26 April 1990 ... **[6]**similarity: <R>(Title: Romina Yan) Romina Yankelevich (5 September 1974 – 28 September 2010) ... **[7]**similarity: <R>(Title: Sweet Baby (Erreway song)) "Sweet Baby" is the debut single ... **[8]**similarity: <R>(Title: Marcela Kloosterboer) Marcela Kloosterboer (born 5 July 1983) is ... **[9]**similarity: <R>(Title: Mili Urién) Milagros Urién, simply known as Mili, is a fictional ... **[10]**similarity: <R>(Title: Rincón de Luz (orphanage)) Rincón de Luz is a fictional orphanage and ... **LLM's Response** Erreway
LLaMA2$_{7B}$	**Input Documents** *As above but without retrieval information.* **LLM's Response** Luisana Lopilato

Table 8: Case studies of R^{2}AG.