Align Documents to Questions: Question-Oriented Document Rewriting for Retrieval-Augmented Generation

Jiaang Li[†], **Zhendong Mao**[†*], **Quan Wang**[‡], **Yuning Wan**[†], **Yongdong Zhang**[†]
[†]University of Science and Technology of China
[‡]Beijing University of Posts and Telecommunications
jali@mail.ustc.edu.cn

Abstract

Retrieval-Augmented Generation (RAG) enhances the factuality of Large Language Models (LLMs) by incorporating retrieved documents and/or generated context. However, LLMs often exhibit a stylistic bias when presented with mixed contexts, favoring fluent but hallucinated generated content over factually grounded yet disorganized retrieved evidence. This phenomenon reveals that the utility of retrieved information is bottlenecked by its presentation. To bridge this gap, we propose **QREAM**, a style-controlled rewriter that aligns retrieved documents with a question-oriented style while preserving facts, better for LLM readers to utilize. Our framework consists of two stages: (1) **QREAM-ICL**, which uses stylistic seeds to guide iterative rewriting exploration; and (2) **QREAM-FT**, a lightweight student model distilled from denoised ICL outputs. QREAM-FT employs dual-criteria rejection sampling, filtering based on answer correctness and factual consistency to ensure high-quality supervision. QREAM seamlessly integrates into existing RAG pipelines as a plug-and-play module. Experiments demonstrate that QREAM consistently enhances advanced RAG pipelines, yielding up to 8% relative improvement with negligible latency overhead, effectively balancing question relevance with factual grounding.

1 Introduction

Large Language Models (LLMs) have achieved remarkable progress in natural language understanding and generation (Brown et al., 2020; Dubey et al., 2024; Jiang et al., 2023a). However, they still face significant challenges in knowledge-intensive tasks such as Open-Domain Question Answering (ODQA), primarily due to limitations in their internal parametric knowledge, including outdated information, knowledge gaps, and hallucinations

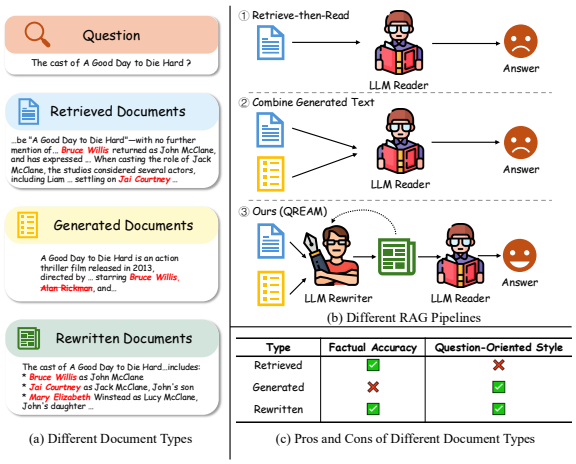

Figure 1: Illustration of different document types in RAG pipelines. Our method integrates the factual accuracy of retrieved documents and the question-oriented style of the generated documents.

(Wen et al., 2024; Sun et al., 2024). Retrieval-Augmented Generation (RAG) has emerged as a powerful paradigm to mitigate these issues by incorporating external knowledge sources to enhance LLMs' factual capabilities (Karpukhin et al., 2020; Asai et al., 2024). As illustrated in Figure 1(a), existing RAG systems typically rely on two types of auxiliary resources: (1) retrieved documents from an authorized knowledge base, which provide reliable factual evidence but often contain redundant or irrelevant content (Liu et al., 2024; Shi et al., 2023), and (2) LLM-generated documents conditioned on the input question (Petroni et al., 2019; Yu et al., 2023; Liu et al., 2022), which tend to be better structured and question-focused but prone to hallucinations (Ji et al., 2023; Rawte et al., 2023).

While integrating retrieval and generation holds promise for balancing factual reliability with contextual relevance (Yu et al., 2023; Zhang et al., 2023), current systems are bottlenecked by *stylistic bias* (Tan et al., 2024). Specifically, LLM readers exhibit a strong preference for the fluent, question-oriented nature of generated content, often priori-

* Corresponding author: Zhendong Mao

Findings of the Association for Computational Linguistics: ACL 2026, pages 17838–17849

tizing it over factually grounded yet disorganized retrieved evidence. This phenomenon reveals that the utility of retrieved information is significantly constrained by its presentation.

Driven by this insight, we aim to restructure retrieved documents to emulate the question-oriented style of generated text, thereby facilitating their usage in RAG. To realize this, we propose **QREAM** (**Q**uestion-oriented document **R**ewriting for **E**ffective **A**nswering **M**odels), a novel framework that integrates the complementary strengths of retrieved and generated documents into a unified context. Functioning as a plug-and-play post-processing module, QREAM transforms raw retrieved documents to align with a question-oriented discourse style while strictly preserving factual correctness. It expands the role of generative LLMs in RAG, shifting them from free-form generators into style-controlled, content-grounded rewriters (Figure 1 (b)), thereby effectively unlocking the potential of retrieved evidence.

QREAM operates in a two-stage *Explore-then-Distill* paradigm. First, **QREAM-ICL** leverages stylistic seeds to guide an iterative exploration of diverse, question-oriented structures. However, this exploration inevitably introduces noise, such as hallucinations or irrelevant content. To extract robust rewriting patterns, we propose a **Bidirectional Denoising** strategy for distillation. We rigorously filter candidates via *dual-criteria rejection sampling*: looking downstream to verify answer utility and upstream to ensure factual fidelity. This derives a purified dataset to train **QREAM-FT**, a student model that learns to produce high-quality rewrites while avoiding the teacher's noise. Consequently, QREAM-FT effectively denoises the retrieval pipeline, combining factual grounding with stylistic fluency as shown in Figure 1(c), while naturally offering superior inference efficiency.

Our contributions are summarized as follows:

- We propose QREAM, a plug-and-play framework that optimizes the documents used by RAG reader. We introduce a bidirectional denoising distillation to filter noisy candidates into purified supervision, enabling a lightweight model for stable rewriting.
- Extensive experiments on four ODQA benchmarks demonstrate that QREAM consistently enhances advanced RAG pipelines, achieving up to 8% relative improvement in accuracy. Notably, QREAM-FT delivers these gains with negligible latency overhead.
- We provide in-depth analyses using a novel style alignment metric (s_{orient}) and quantitative hallucination evaluation. Results verify that QREAM effectively extracts and restructures useful information into a question-oriented style while maintaining factual grounding. Furthermore, our analysis confirms that QREAM effectively mitigates the stylistic bias of LLM readers.

2 Method

We aim to transform the presentation of retrieved evidence for easier information utilization in RAG. Formally, given a question q and the raw retrieved documents $R = \{r_i\}_{i=1}^{K}$, we introduce a rewriter $\mathcal{M}^{\text{Rew}}$ to refine R into a question-oriented variant $\tilde{R}$, optimized for the utilization of RAG reader $\mathcal{M}^{\text{Read}}$. As shown in Figure 2, our framework follows an *Explore-then-Distill* paradigm. In Stage I, we introduce **QREAM-ICL** to iteratively explore diverse rewriting patterns, guided by synthesized stylistic seeds. We further apply a bidirectional denoising mechanism via **dual-criteria rejection sampling**, which filters out the inherent noise in the output of QREAM -ICL. In Stage II, these high-quality rewrites are used to train **QREAM-FT**, a lightweight student model that ensures efficient and stable rewriting.

2.1 Stage I: In-Context Stylistic Exploration

The first stage aims to generate high-quality, style-aligned document candidates. Given the absence of ground-truth "question-oriented documents", we employ an iterative prompting strategy with stylistic seeds.

Stylistic Seeds Generation. We first construct a set of stylistic seeds to guide the rewriting. We sample M seed questions $\{\hat{q}_i\}_{i=1}^{M}$ from the training set. For each $\hat{q}_i$, we prompt an LLM generator $\mathcal{M}^{\text{Gen}}$ to produce a background-style document g_i using a template $\mathcal{T}^{\text{Gen}}$:

$$g_i = \mathcal{M}^{\text{Gen}}(\mathcal{T}^{\text{Gen}}(\hat{q}_i)) \tag{1}$$

The resulting pairs $\mathcal{E} = \{(\hat{q}_i, g_i)\}_{i=1}^{M}$ serve as few-shot demonstrations. Crucially, by using unrelated questions as seeds, we disentangle *style* from *content*, encouraging the rewriter to mimic the structural pattern without hallucinating the content of the seeds.

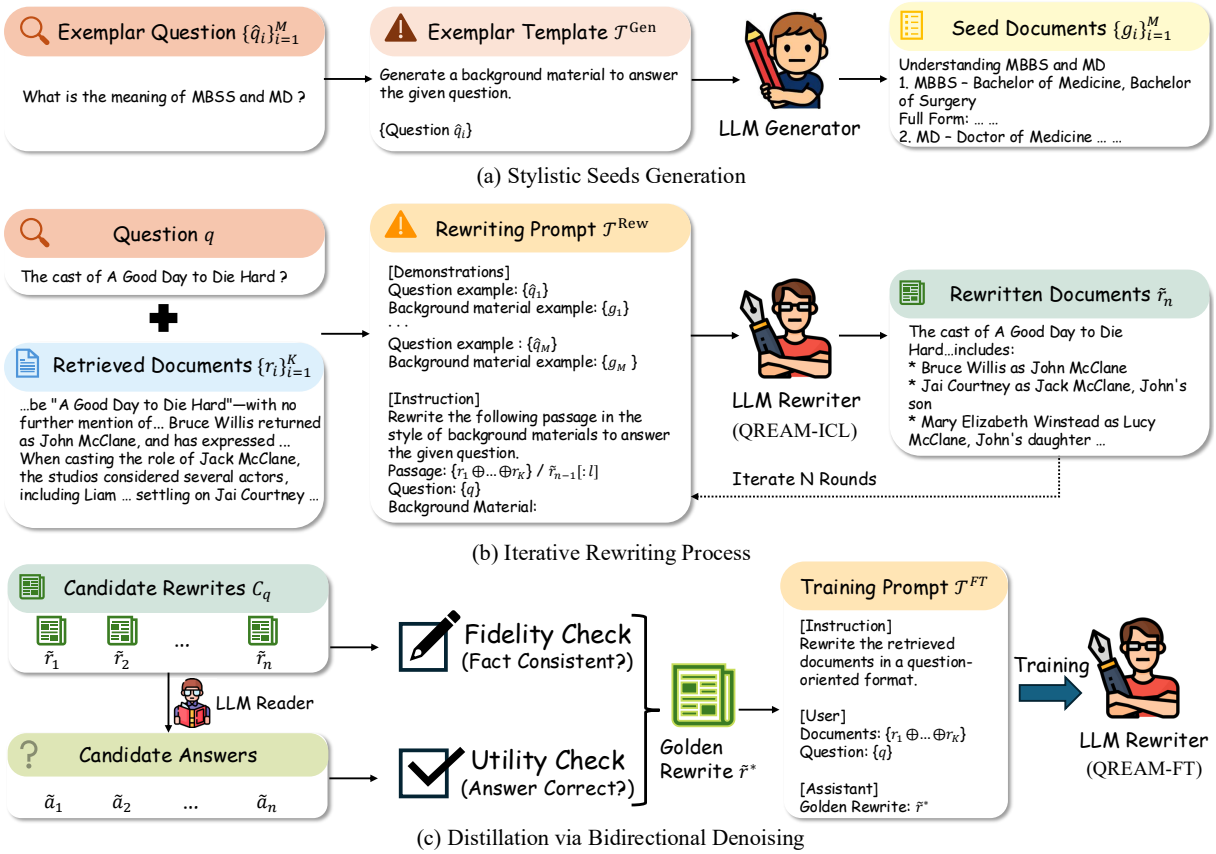

Figure 2: The QREAM framework operating in an *Explore-then-Distill* paradigm. **(a-b) Stage I**: QREAM-ICL explores diverse question-oriented rewrites via iterative in-context learning guided by stylistic seeds. **(c) Stage II**: We employ a Bidirectional Denoising mechanism to filter noisy candidates via dual-criteria rejection sampling, deriving a purified dataset to train the QREAM-FT.

Iterative Rewriting as Candidate Generation. Given a target question q and retrieved documents $R = \{r_k\}_{k=1}^{K}$, we concatenate them into a raw context $r_{\text{raw}} = r_1 \oplus \cdots \oplus r_K$. We employ an iterative rewriting strategy to progressively refine the context. In the first round ($n = 1$), the LLM rewriter $\mathcal{M}^{\text{Rew}}$ is prompted with the exemplars $\mathcal{E}$ and the raw context:

$$\tilde{r}_1 = \mathcal{M}^{\text{Rew}}(\mathcal{T}^{\text{Rew}}(\mathcal{E}, r_{\text{raw}}, q)) \quad (2)$$

For subsequent iterations $n > 1$, the output from the previous step $\tilde{r}_{n-1}$ serves as the input basis. To prioritize high-density information and adhere to context limits, we retain the first l tokens of the previous output:

$$\tilde{r}_n = \mathcal{M}^{\text{Rew}}(\mathcal{T}^{\text{Rew}}(\mathcal{E}, [\tilde{r}_{n-1}]_{1:l}, q)) \quad (3)$$

This process repeats for N rounds, yielding a set of candidate rewrites $\mathcal{C}_q = \{\tilde{r}_1, \ldots, \tilde{r}_N\}$ for the target question q. While the final output $\tilde{r}_N$ can be used for RAG directly, we primarily utilize $\mathcal{C}_q$ as the candidate pool for distilling a robust student model in Stage II.

2.2 Stage II: Distillation via Bidirectional Denoising

While QREAM-ICL is competitive, its reliance on iterative prompting inevitably contains noise. Therefore, we propose a **Bidirectional Denoising** strategy via dual-criteria rejection sampling: (1) **Downstream Utility Check:** verifying if the rewrite effectively leads the reader to the correct answer; (2) **Upstream Fidelity Check:** ensuring the rewrite remains grounded in the raw retrieved documents. This process derives a purified dataset $\mathcal{D}_{\text{train}}$ to train QREAM-FT

Dual-Criteria Scoring. We evaluate each candidate rewrite $\tilde{r} \in \mathcal{C}_q$ based on two dimensions: *downstream utility* and *factual fidelity*. First, we feed $\tilde{r}$ into the fixed QA reader $\mathcal{M}^{\text{Read}}$ to obtain a predicted answer $\hat{a}$. We apply a hard filter to discard invalid candidates: any rewrite that fails to contain the ground-truth answer strings is rejected. Mathematically, we require $\mathbb{I}(a^* \in \hat{a}) = 1$, where $\in$ denotes the string inclusion operation. For the surviving candidates, we compute a **Performance**

Score S_{perf} based on the token-level F1 match between $\hat{a}$ and a^*:

$$S_{\text{perf}}(\tilde{r}) = \text{F1}(\mathcal{M}^{\text{Read}}(\tilde{r}, q), a^*) \quad (4)$$

Second, to prevent hallucinations, we define a **Consistency Score** S_{fact}. We adopt a decomposition-based verification approach (Min et al., 2023). Specifically, we extract a set of atomic facts $\mathcal{F}$ from the rewrite $\tilde{r}$ and verify each fact $f \in \mathcal{F}$ against the raw retrieved evidence r_{raw}. S_{fact} is calculated as the ratio of supported facts:

$$S_{\text{fact}}(\tilde{r}, r_{\text{raw}}) = \frac{1}{|\mathcal{F}|} \sum_{f \in \mathcal{F}} \mathbb{I}(r_{\text{raw}} \models f) \quad (5)$$

where $\mathbb{I}(\cdot)$ is the indicator function and $\models$ denotes textual entailment (determined by an LLM). Implementation details are provided in Appendix A.

We aggregate these metrics into a composite quality score:

$$S_{\text{total}}(\tilde{r}) = \frac{1}{2}\left(S_{\text{perf}}(\tilde{r}) + S_{\text{fact}}(\tilde{r}, r_{\text{raw}})\right) \quad (6)$$

Golden Data Selection & Student Training. We construct the distilled training set $\mathcal{D}_{\text{train}}$ by selecting the optimal rewrite $\tilde{r}^*$ that maximizes the total quality score for each question:

$$\tilde{r}^* = \underset{\tilde{r} \in \mathcal{C}_q, \mathbb{I}(a^* \in \hat{a})=1}{\arg\max} S_{\text{total}}(\tilde{r}) \quad (7)$$

Questions with no valid candidates passing the hard filter are excluded to avoid noisy supervision.

To train the student model, we wrap the input data into a standardized instruction-tuning template $\mathcal{T}^{\text{FT}}$. Finally, we fine-tune the lightweight student model $\mathcal{M}^{\text{Student}}$ to maximize the likelihood of the golden rewrite $\tilde{r}^*$:

$$\mathcal{L} = -\sum_{(q, r_{\text{raw}}, \tilde{r}^*) \in \mathcal{D}_{\text{train}}} \log P_\theta(\tilde{r}^* \mid \mathcal{T}^{\text{FT}}(q, r_{\text{raw}})) \quad (8)$$

During inference, QREAM-FT generates refined documents in a single forward pass from the question and retrieved documents, significantly reducing latency while denoising the output via the filtered supervision.

3 Experiments

3.1 Experimental Setup

Datasets. We evaluate QREAM on four widely recognized ODQA benchmarks covering both single-hop and multi-hop reasoning: Natural Questions (NQ) (Kwiatkowski et al., 2019), TriviaQA (TQA) (Joshi et al., 2017), HotpotQA (Yang et al., 2018), and 2WikiMultiHopQA (2WikiMQA) (Ho et al., 2020). We follow standard evaluation protocols using the official splits.

Baselines. We compare QREAM against three categories of baselines: (1) *Standard RAG pipelines* utilizing raw retrieved documents, LLM-generated documents (Yu et al., 2023), or their concatenation. (2) *Post-retrieval processing methods*, including **LongLLMLingua** (Jiang et al., 2024) (prompt compression), **CompAct** (Yoon et al., 2024) (active compression), **RECOMP** (Xu et al., 2024) (abstractive compression), and **FaviComp** (Jung et al., 2025) (familiarity-aware fusion). (3) *Advanced RAG frameworks*, where we integrate QREAM into **Self-RAG** (Asai et al., 2024) and **HippoRAG** (Gutiérrez et al., 2024) to demonstrate its plug-and-play capability.

Implementation Details. For consistency with previous works, we employ Contriever-MSMARCO (Izacard et al., 2021) to retrieve the top-5 passages. For the reader backbone, we utilize Llama-3-8B-Instruct and Mistral-7B-Instruct-v0.3, with the same QA prompt template as Jung et al. (2025). QREAM-ICL is employed by using $M = 4$ stylistic seeds and perform $N = 3$ iterations of rewriting, with the truncate length set to $l = 100$. We then construct a training corpus by sampling 1,000 training samples per dataset (4,000 total) and filtering them via dual-criteria rejection sampling. This data is used to fine-tune a Llama-3.2-1B-Instruct student model as QREAM-FT. For integration with advanced RAG pipelines, we utilize the QREAM-FT (1B) distilled from the Llama-3 teacher due to its optimal performance-efficiency trade-off. We report Accuracy (Acc) and token-level F1 score.

In this paper, $M^{gen} = M^{rew}$ for simplicity. However, they can be decoupled in practice. For instance, using a more powerful model for seed generation while using a more efficient one for rewriting.

3.2 Main Results

We evaluate the effectiveness of QREAM across two experimental settings: (1) as a post-processing module in standard RAG pipelines, and (2) as a plug-and-play component in advanced SOTA frameworks. Table 1 summarizes the results.

Performance on Standard RAG Pipelines. We first compare QREAM with various baselines us-

Methods	NQ		TQA		HotpotQA		2WikiMQA		Avg.	
	Acc	F1	Acc	F1	Acc	F1	Acc	F1	Acc	F1
Standard RAG Pipeline with Llama-3-8B-Instruct										
Retrieved Doc.	42.6	47.1	67.6	70.8	30.3	38.7	22.0	26.8	40.6	45.9
Generated Doc.	39.1	23.1	66.1	47.8	32.3	22.1	23.7	17.1	40.3	27.5
Retrieved + Generated	41.8	22.4	68.2	44.8	32.4	20.2	24.9	15.7	41.8	25.8
LongLLMLingua	35.4	40.9	64.8	67.6	25.9	34.7	19.2	24.2	36.3	41.9
CompAct	42.3	46.1	67.0	69.7	29.8	37.5	21.4	26.6	40.1	45.0
RECOMP	41.5	45.8	67.2	70.1	30.5	38.2	23.1	28.5	40.6	45.7
FaviComp	42.3	46.6	68.4	**71.5**	32.3	41.0	27.6	33.6	42.7	48.2
QREAM-ICL	43.8	47.5	**68.7**	**70.7**	38.2	44.5	**30.4**	34.6	45.3	49.3
QREAM-FT	**44.6**	**48.0**	68.5	70.5	**38.4**	**44.9**	30.1	**34.9**	**45.6**	**49.4**
Standard RAG Pipeline with Mistral-7B-Instruct										
Retrieved Doc.	40.2	39.3	66.2	68.6	30.3	37.2	26.6	28.5	40.8	43.4
Generated Doc.	37.5	20.2	65.6	46.3	32.0	21.0	28.4	21.1	40.9	27.2
Retrieved + Generated	39.5	19.3	66.7	43.3	32.7	19.6	29.8	19.9	42.2	25.5
LongLLMLingua	34.3	36.4	63.8	63.8	27.0	34.7	25.5	28.0	37.7	40.7
CompAct	38.8	38.9	65.1	67.1	30.2	37.1	24.9	27.6	39.8	42.7
RECOMP	39.1	39.5	65.5	67.8	30.8	38.0	26.1	29.2	40.4	43.6
FaviComp	40.3	40.4	65.9	**68.9**	32.0	40.5	29.7	35.1	42.0	46.2
QREAM-ICL	41.1	**40.5**	66.3	67.5	35.8	42.5	**34.3**	**36.7**	44.4	**46.8**
QREAM-FT	**41.5**	40.1	**67.6**	67.1	**36.9**	**42.9**	33.6	35.9	**44.9**	46.5
Integration with Advanced RAG Frameworks										
Self-RAG	43.2	42.7	69.3	72.5	37.5	46.5	27.1	32.6	44.3	48.6
+ FaviComp	44.1	42.5	70.1	73.1	39.5	48.2	29.0	34.1	45.7	49.5
+ QREAM-ICL	45.0	44.2	**71.4**	74.0	40.8	50.1	30.5	**36.8**	46.9	51.3
+ QREAM-FT	**45.3**	**44.9**	71.2	**74.5**	**41.2**	**50.3**	**31.4**	36.2	**47.3**	**51.4**
HippoRAG	47.0	41.5	65.6	68.8	45.2	59.2	35.7	62.7	48.6	58.1
+ FaviComp	48.2	42.8	66.5	70.0	47.5	61.7	38.0	65.1	50.1	59.9
+ QREAM-ICL	48.8	43.1	67.2	70.5	50.3	64.4	**39.8**	**66.8**	51.5	61.2
+ QREAM-FT	**49.5**	**43.9**	**67.6**	**71.9**	**50.9**	**65.6**	38.5	65.6	**51.6**	**61.8**

Table 1: **Main Results on ODQA benchmarks.** QREAM-FT achieves comparable or superior performance to the ICL-based Teacher. Notably, the bottom section demonstrates that QREAM serves as an effective plug-and-play module, yielding consistent additive gains when integrated into SOTA frameworks. Best results are bolded.

Method	Rewriter	Latency	Avg. Acc
Standard RAG	None	0.16s	40.6
QREAM-ICL	Llama-3-8B	2.41s	45.3
QREAM-FT	Llama-3.2-1B	0.18s	45.6

Table 2: **Efficiency Analysis on NQ.** QREAM-FT delivers a $\sim$13$\times$ speedup over the QREAM-ICL while maintaining superior accuracy.

ing Llama-3-8B and Mistral-7B. As shown in Table 1, both variants of QREAM outperform all baselines. **QREAM-ICL** demonstrates robust performance (Avg Acc 45.3% on Llama-3), validating the exploration capability of the iterative rewriting strategy. Our distilled student model, **QREAM-FT**, achieves highly competitive results (Avg Acc 45.4%). While employing a significantly smaller backbone (1B vs 8B), QREAM-FT matches or even surpasses the Teacher on datasets like NQ and HotpotQA We attribute this to the *dual-criteria rejection sampling* (Sec 2.2), which acts as a denoising filter to remove suboptimal rewrites generated during exploration, ensuring the student learns from a cleaner distribution. The gains are particularly pronounced on multi-hop datasets (HotpotQA, 2WikiMQA), where QREAM-FT improves over Standard RAG by roughly **8%** in accuracy. Compared to RECOMP and FaviComp, which focus on compression or fusion, QREAM's style-guided rewriting better preserves the logical chain required for complex reasoning.

Integration with Advanced RAG Pipelines. We further validate the plug-and-play capability of QREAM by integrating both ICL and FT variants into Self-RAG and HippoRAG. For this setting, we employ the efficient 1B QREAM-FT model to rewrite documents upstream of the respective frameworks. As shown in the third block of Table 1, both variants yield additive gains over the base frameworks. QREAM-FT generally outperforms or matches QREAM-ICL, suggesting that the distilled model captures the core rewriting patterns essential for question answering. Results demonstrate that QREAM complements the advanced systems and provides additive gains to these SOTA frameworks. Further evaluations on **GPT-5 mini** (Appendix D) demonstrate consistent gains, confirming QREAM's effectiveness even integrated with SOTA proprietary LLM.

3.3 Efficiency Analysis

We analyze the computational cost of our *Explore-then-Distill* paradigm. Table 2 reports the average end-to-Natural Questions dataset, using Llama3-8B-Instruct as the reader. QREAM-ICL serves as a powerful explorer but incurs higher latency (2.41s) due to iterative prompting with long demonstrations. In contrast, QREAM-FT distills this capability into a lightweight model and refines the raw retrieval within a single forward pass, reducing the latency to **0.18s**. This is comparable to the Standard RAG pipeline (0.16s), confirming that QREAM-FT successfully bridges the gap between high-performance rewriting and real-time deployment requirements.

4 Analysis

In this section, we provide a rigorous analysis to verify the design goals of QREAM: enhancing document quality, validating component contributions, and mitigating stylistic bias.

4.1 Quality of Rewritten Documents

A core motivation of QREAM is to combine the *stylistic relevance* of generated text with the *factual grounding* of retrieved text. We introduce two quantitative metrics to evaluate these properties.

Question-Oriented Style Score (s_{orient}). We use s_{orient} to measure how effectively a document d aligns with the discourse style of the question q. Following the probability ranking principle (Sachan et al., 2022b), we formulate this as the length-normalized log-probability of reconstructing q given d, computed by the LLM reader $\mathcal{M}^{\text{Gen}}$:

$$s_{\text{orient}}(d, q) = \frac{1}{|q|} \sum_{t=1}^{|q|} \log P_{\mathcal{M}^{\text{Gen}}}(q_t \mid q_{<t}, d) \quad (9)$$

Higher scores indicate stronger alignment with the question's intent.

Document Type	Style (s_{orient}) ↑	Fact (r_{inc}) ↓
Retrieved Doc.	-2.53	**0.0%**
Generated Doc.	**-1.35**	33.7%
QREAM-ICL	-1.41	13.8%
QREAM-FT	-1.45	**9.2%**

Table 3: **Document Quality Analysis.** QREAM balances stylistic alignment (approaching generated text) with hallucination rates (approaching retrieved text).

Factual Inconsistency Rate (r_{inc}). To quantify hallucinations, we adapt the FactScore framework (Min et al., 2023). We extract atomic facts from the document d and verify them against the original retrieval r_{raw} using GPT-4o. The rate r_{inc} is the percentage of unsupported facts.

Results. Table 3 presents the results averaged across four datasets. We observe a clear trade-off in existing document types: *Generated Documents* achieve high style scores (-1.35) but suffer from severe hallucinations (33.7%), while *Retrieved Documents* are factual but stylistically misaligned (-2.53). In contrast, QREAM-FT achieves the best of both worlds. Its style score (-1.45) is comparable to free-form generation. Crucially, thanks to the *dual-criteria rejection sampling*, QREAM-FT maintains a low hallucination rate (9.2%), which is even lower than its teacher QREAM-ICL (13.8%). This confirms that our distillation process effectively transfers the style while actively denoising the content.

4.2 Ablation Studies

We conduct ablation studies to verify the necessity of our two-stage design: the *distillation criteria* (Stage II) and the *exploration strategy* (Stage I). All ablation results reported below use Llama-3-8B-Instruct as the QA reader backbone.

Effectiveness of Distillation Criteria (Stage II). The robustness of QREAM-FT stems from the *Dual-Criteria Rejection Sampling*. To isolate its impact, we train student variants using different filtering strategies on the Llama-3.2-1B backbone and

Distillation Strategy	Avg F1	Fact (r_{inc})
QREAM-FT (Dual-Criteria)	**49.4**	**9.2%**
w/o Filtering (Raw ICL)	47.8	14.8%
w/ Utility Check Only	48.9	13.5%
w/ Fidelity Check Only	48.2	12.0%

Table 4: **Ablation on Distillation Criteria (Stage II).** Results are averaged across four datasets using Llama-3-8B-Instruct as the reader.

evaluate their impact on the downstream reader's performance. As shown in Table 4, training on raw ICL outputs without filtering ("No Filter") yields the lowest performance (47.8 Avg F1) and highest hallucination rate (14.8%), as the student blindly imitates the teacher's noise. Using only the *Utility Check* or the *Fidelity Check* improves F1 compared to no filtering but retains high hallucinations (13.5%), as it rewards any rewrite that hits the answer string regardless of factual support, implicitly incentivizing model hallucination. Our dual-criteria approach achieves the optimal balance, yielding the highest F1 score while maintaining a low hallucination rate. This confirms that optimizing on data jointly checked for utility and consistency is essential.

Impact of Stylistic Seeds (Stage I). To validate the exploration mechanism, we analyze the design choices of stylistic seeds as shown in Figure 3. First, removing stylistic seeds ("Zero-shot Rewrite") leads to a performance drop even below raw retrieval. This confirms that the target "question-oriented style" is implicit and difficult to elicit via instructions alone; seeds act as necessary anchors. Second, we investigate the *source* of these seeds. We compare our standard approach (using unrelated questions) against "Self-Rewrite", where seeds are synthesized from the target question itself. Surprisingly, Self-Rewrite performs worse than raw documents and degrades further as more exemplars are added. This is likely because self-generated seeds introduce hallucinations that appear semantically relevant, contaminating the rewriting process. In contrast, our use of unrelated seeds effectively disentangles *style* from *content*, enabling the model to learn the structural pattern without factual interference. Finally, performance improves with the number of exemplars, plateauing around $M = 4$, which validates our default configuration.

4.3 Mitigation of Stylistic Bias

Finally, we verify whether QREAM rewrites effectively adopt the high-utility presentation of generated content to neutralize the reader's inherent stylistic bias. We evaluate under the **Context-Conflicting (CC)** setting (Tan et al., 2024), an adversarial scenario designed to probe reader preference. In this setting, the reader is simultaneously presented with a *correct* retrieved document and an *incorrect* generated document.

	NQ-CC		TQA-CC	
Reader Model	**RAW**	**QREAM**	**RAW**	**QREAM**
Llama-3	19.4	**77.4**	17.2	**69.4**
Mistral-7B	16.1	**72.6**	14.2	**65.8**

Table 5: **QREAM mitigates the stylistic bias of LLM readers.** Compared to using raw documents, QREAM demonstrates superior robustness under the Context-Conflicting (CC) setting.

As evidenced in Table 5, Standard RAG using raw retrievals is highly susceptible to stylistic bias, often prioritizing incorrect generation over factual evidence. For instance, Llama-3's accuracy on NQ-CC is only 19.4% even when the correct retrieval is provided, indicating that the reader is easily misled by the generated context. In contrast, replacing raw retrievals with QREAM rewrites restores the accuracy to 77.4% on NQ-CC and 69.4% on TQA-CC. Consistent trends are also observed with Mistral-7B reader. This phenomenon demonstrates that the style alignment achieved by QREAM effectively mitigates the stylistic bias of LLM readers, thereby unlocking the potential of retrieved evidence.

4.4 Qualitative Analysis

Table 6 illustrates the qualitative superiority of QREAM over the strongest baseline (FaviComp). Unlike the baseline, which often retains distracting entities (Case 1) or preserves ambiguous logic (Case 2), QREAM explicitly disentangles critical facts from noise. By restructuring the evidence into a coherent, question-oriented format, our method clears the reasoning path for the reader, ensuring correct predictions even in complex scenarios.

5 Related Works

5.1 Retrieval-Augmented Generation

Regular Retrieval-Augmented Generation (RAG) follows a retrieve-then-read modular pipeline, which involves retrieving relevant information from an external database before generating a final prediction (Fan et al., 2024; Gao et al., 2024). Recently, numerous studies have introduced new modules or refined the original steps. In the *pre-retrieval stage*, methods have been developed to

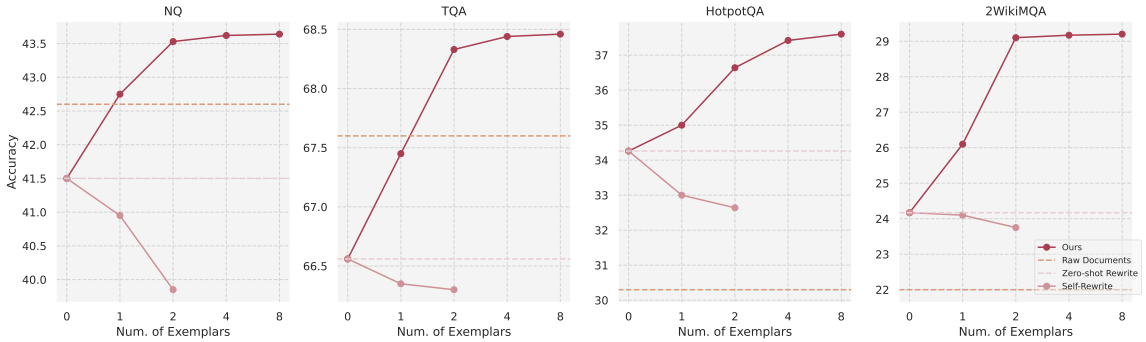

Figure 3: **Ablation on QREAM-ICL.** The QA performance with different numbers and designs of stylistic seeds

Case 1: Distracting Context (Entity Confusion)
Question: *what's the dog's name on tom and jerry?*
[Baseline: Compression]
Doc: Tom, a grey and white domestic shorthair cat, is the main character... while Jerry is a small brown mouse...
Pred: Jerry ✗
[Ours: Question-oriented Rewriting]
Rewrite: Spike is a recurring supporting character... He first appeared in the cartoon "Dog Trouble"...
Pred: Spike ✓
Case 2: Ambiguous Logic (Role Confusion)
Question: *who wrote cant get you out of my head lyrics?*
[Baseline: Compression]
Doc: Kylie Minogue, Cathy Dennis, and Rob Davis **wrote the song** "Can't Get You Out of My Head"...
Pred: Kylie Minogue ✗
[Ours: Question-oriented Rewriting]
Rewrite: The **lyrics** of "Can't Get You Out of My Head" were written by Cathy Dennis and Rob Davis.
Pred: Cathy Dennis ✓

Table 6: **Qualitative Comparisons** with the samples from NQ. While the baseline retains ambiguous or distracting content, QREAM explicitly extracts question-relevant information, correcting the reader's prediction.

determine whether and when retrieval is necessary (Jiang et al., 2023b; Mallen et al., 2023), decompose complex questions into subqueries for multiple retrievals (Dhuliawala et al., 2024), or transform the initial query to improve the retrieval quality (Ma et al., 2023). In the *retrieval stage*, retrievers are often optimized using feedback from LLMs (Shi et al., 2024), or more sophisticated reranking strategies are employed (Sachan et al., 2022a). In the *post-retrieval stage*, most works focus on compressing retrieved documents to improve efficiency and effectiveness (Yoon et al., 2024; Jiang et al., 2024; Jung et al., 2025). Finally, in the *generation stage*, additional training is conducted to make LLMs more robust to noise (Yoran et al., 2024), or automatically judge the correctness of generated outputs (Asai et al., 2024; Yan et al., 2024). We focus on the *post-retrieval stage*. Different from existing works, we aim to imbue the retrieved documents with the question-oriented style unique to generated documents.

More complex RAG frameworks have also emerged. Self-RAG (Asai et al., 2024) introduces self-reflection to adaptively retrieve and generate content, while HippoRAG (Gutiérrez et al., 2024) builds a long-term memory to better integrate knowledge over time. Our plug-and-play rewriting method can be integrated into their post-retrieval stage for further performance gains.

5.2 RAG with Generated Documents

While retrieved documents offer factual grounding, documents generated by an LLM in response to a query are often better aligned with the question in its style and structure, but are notorious for hallucinations (Liu et al., 2022; Petroni et al., 2019). While early methods combine both documents by training a small-scale model (Yu et al., 2023) and (Zhang et al., 2023), a key challenge is the stylistics bias of LLM readers, which favor fluent but potentially inaccurate generated text (Tan et al., 2024). To address this, we propose a novel rewriting mechanism to integrate the merits of both document types.

6 Conclusion

We present QREAM, a framework that transforms retrieved evidence into a question-oriented style to facilitate effective utilization. Operating in an *Explore-then-Distill* paradigm, we distill robust reasoning patterns into a lightweight student via a bidirectional denoising mechanism. Empirical results confirm that this data-centric approach enables a 1B student to output high-quality, factually consistent rewrites, serving as an efficient plug-and-play module for SOTA RAG systems with negligible latency. Crucially, our analysis validates that by

optimizing document presentation, QREAM effectively neutralizes the reader's stylistic bias, ensuring factual evidence is prioritized over hallucinated fluency. We hope this work inspires future research into style-aware optimization for trustworthy RAG.

Limitations

Although QREAM's performance is inherently dependent on the quality of the retrieved documents, this paper assumes that the retrieval process has already been conducted and does not address techniques such as retriever training or document reranking to improve the quality of the retrieved documents. As a result, any deficiencies in the initial retrieval, such as missing information or less relevant content, can still limit QREAM's effectiveness. Future work could explore joint optimization of retrieval and rewriting processes to address this limitation.

Acknowledgments

We would like to thank all reviewing committee for the valuable comments. Thank Ruichen Zheng, who improves this work greatly. This work is supported by the National Natural Science Foundation of China (grants No.62376033 and 62232006).

References

Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. 2024. Self-RAG: Learning to retrieve, generate, and critique through self-reflection. In *The Twelfth International Conference on Learning Representations*.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, and 12 others. 2020. Language models are few-shot learners. *Preprint*, arXiv:2005.14165.

Shehzaad Dhuliawala, Mojtaba Komeili, Jing Xu, Roberta Raileanu, Xian Li, Asli Celikyilmaz, and Jason Weston. 2024. Chain-of-verification reduces hallucination in large language models. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 3563–3578, Bangkok, Thailand. Association for Computational Linguistics.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

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*, KDD '24, page 6491–6501, New York, NY, USA. Association for Computing Machinery.

Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, Meng Wang, and Haofen Wang. 2024. Retrieval-augmented generation for large language models: A survey. *Preprint*, arXiv:2312.10997.

Bernal Jiménez Gutiérrez, Yiheng Shu, Yu Gu, Michihiro Yasunaga, and Yu Su. 2024. Hipporag: Neurobiologically inspired long-term memory for large language models. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Xanh Ho, Anh-Khoa Duong Nguyen, Saku Sugawara, and Akiko Aizawa. 2020. Constructing a multi-hop QA dataset for comprehensive evaluation of reasoning steps. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6609–6625, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. 2021. Unsupervised dense information retrieval with contrastive learning.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. 2023. Survey of hallucination in natural language generation. *ACM Comput. Surv.*, 55(12).

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, and 1 others. 2023a. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Huiqiang Jiang, Qianhui Wu, Xufang Luo, Dongsheng Li, Chin-Yew Lin, Yuqing Yang, and Lili Qiu. 2024. LongLLMLingua: Accelerating and enhancing LLMs in long context scenarios via prompt compression. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1658–1677, Bangkok, Thailand. Association for Computational Linguistics.

Zhengbao Jiang, Frank Xu, Luyu Gao, Zhiqing Sun, Qian Liu, Jane Dwivedi-Yu, Yiming Yang, Jamie Callan, and Graham Neubig. 2023b. Active retrieval augmented generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 7969–7992, Singapore. Association for Computational Linguistics.

Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. 2017. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1601–1611, Vancouver, Canada. Association for Computational Linguistics.

Dongwon Jung, Qin Liu, Tenghao Huang, Ben Zhou, and Muhao Chen. 2025. Familiarity-aware evidence compression for retrieval-augmented generation. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 16181–16196, Suzhou, China. Association for Computational Linguistics.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781, Online. Association for Computational Linguistics.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:452–466.

Jiacheng Liu, Alisa Liu, Ximing Lu, Sean Welleck, Peter West, Ronan Le Bras, Yejin Choi, and Hannaneh Hajishirzi. 2022. Generated knowledge prompting for commonsense reasoning. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3154–3169, Dublin, Ireland. Association for Computational Linguistics.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2024. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173.

Xinbei Ma, Yeyun Gong, Pengcheng He, Hai Zhao, and Nan Duan. 2023. Query rewriting in retrieval-augmented large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5303–5315, Singapore. Association for Computational Linguistics.

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.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen-tau Yih, Pang Wei Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2023. FActScore: Fine-grained atomic evaluation of factual precision in long form text generation. In *EMNLP*.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. 2019. Language models as knowledge bases? 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 2463–2473, Hong Kong, China. Association for Computational Linguistics.

Vipula Rawte, Swagata Chakraborty, Agnibh Pathak, Anubhav Sarkar, S.M Towhidul Islam Tonmoy, Aman Chadha, Amit Sheth, and Amitava Das. 2023. The troubling emergence of hallucination in large language models - an extensive definition, quantification, and prescriptive remediations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2541–2573, Singapore. Association for Computational Linguistics.

Devendra Sachan, Mike Lewis, Mandar Joshi, Armen Aghajanyan, Wen-tau Yih, Joelle Pineau, and Luke Zettlemoyer. 2022a. Improving passage retrieval with zero-shot question generation. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 3781–3797, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Devendra Singh Sachan, Mike Lewis, Mandar Joshi, Armen Aghajanyan, Wen-tau Yih, Joelle Pineau, and Luke Zettlemoyer. 2022b. Improving passage retrieval with zero-shot question generation.

Freda Shi, Xinyun Chen, Kanishka Misra, Nathan Scales, David Dohan, Ed Chi, Nathanael Schärli, and Denny Zhou. 2023. Large language models can be easily distracted by irrelevant context. In *Proceedings of the 40th International Conference on Machine Learning*, ICML'23. JMLR.org.

Weijia Shi, Sewon Min, Michihiro Yasunaga, Minjoon Seo, Richard James, Mike Lewis, Luke Zettlemoyer, and Wen-tau Yih. 2024. 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, Mexico City, Mexico. Association for Computational Linguistics.

Kai Sun, Yifan Xu, Hanwen Zha, Yue Liu, and Xin Luna Dong. 2024. Head-to-tail: How knowledgeable are large language models (LLMs)? A.K.A. will LLMs replace knowledge graphs? 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 311–325, Mexico City, Mexico. Association for Computational Linguistics.

Hexiang Tan, Fei Sun, Wanli Yang, Yuanzhuo Wang, Qi Cao, and Xueqi Cheng. 2024. Blinded by generated contexts: How language models merge generated and retrieved contexts when knowledge conflicts? In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6207–6227, Bangkok, Thailand. Association for Computational Linguistics.

Zhihua Wen, Zhiliang Tian, Zexin Jian, Zhen Huang, Pei Ke, Yifu Gao, Minlie Huang, and Dongsheng Li. 2024. Perception of knowledge boundary for large language models through semi-open-ended question answering. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Fangyuan Xu, Weijia Shi, and Eunsol Choi. 2024. RECOMP: Improving retrieval-augmented LMs with context compression and selective augmentation. In *The Twelfth International Conference on Learning Representations*.

Shi-Qi Yan, Jia-Chen Gu, Yun Zhu, and Zhen-Hua Ling. 2024. Corrective retrieval augmented generation. *Preprint*, arXiv:2401.15884.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. 2018. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2369–2380, Brussels, Belgium. Association for Computational Linguistics.

Chanwoong Yoon, Taewhoo Lee, Hyeon Hwang, Minbyul Jeong, and Jaewoo Kang. 2024. CompAct: Compressing retrieved documents actively for question answering. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 21424–21439, Miami, Florida, USA. Association for Computational Linguistics.

Ori Yoran, Tomer Wolfson, Ori Ram, and Jonathan Berant. 2024. Making retrieval-augmented language models robust to irrelevant context. In *The Twelfth International Conference on Learning Representations*.

Wenhao Yu, Dan Iter, Shuohang Wang, Yichong Xu, Mingxuan Ju, Soumya Sanyal, Chenguang Zhu, Michael Zeng, and Meng Jiang. 2023. Generate rather than retrieve: Large language models are strong context generators. In *The Eleventh International Conference on Learning Representations*.

Yunxiang Zhang, Muhammad Khalifa, Lajanugen Logeswaran, Moontae Lee, Honglak Lee, and Lu Wang. 2023. Merging generated and retrieved knowledge for open-domain QA. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 4710–4728, Singapore. Association for Computational Linguistics.

A Prompts for Evaluation Metrics

Since most of the specific prompt templates for QREAM-ICL and QREAM-FT are already illustrated in Figure 2, we omit them here to avoid redundancy. In this section, we provide the specific prompts used for our automated evaluation metrics, including the Stylistic Alignment Score (s_{orient}), the Factual Consistency Score (S_{fact}) and the Factual Inconsistency Rate (r_{inc}).

A.1 Prompt for Style Score Calculation

To compute s_{orient}, we use a fixed generator to calculate the log-probability of reconstructing the question given the document. An instruction is used to wrap the input for the generator, which is formatted as follows:

Reconstruct Question from Document

```
[Instruction] Please generate a question
that can be answered by the following
document.
[Document] {Document}
[Question]: ...
```

A.2 Prompts for Hallucination Evaluation

Since $S_{\text{fact}} = 1 - r_{\text{inc}}$, they are computed using the same process. Following FactScore (Min et al., 2023), we employ a two-step process: atomic fact extraction and verification.

Atomic Fact Extraction. We first extract atomic facts from the rewritten document using the following prompt:

Atomic Fact Extraction

```
[Instruction] Please break down the
following text into independent facts. Each
fact should be a concise, self-contained
statement.
[Document] rewritten document
[Atomic Facts]:
```

Fact Verification. We then verify each atomic fact against the raw retrieved evidence:

Fact Verification Prompt

```
[Instruction] Given the following document
as context, determine if the following
statement is supported by the context...
[Document] {Raw Retrieval}
[Statement] {Atomic Fact}
[Answer] (True/False):
```

ID	Instruction Prompt Variant	Avg Acc	Avg F1
#1	*"Rewrite the following passage in the style of background materials to answer the given question"*	**45.3**	**49.3**
#2	*"Following the examples, rewrite the following passage to serve as background material..."*	45.0	49.1
#3	*"Rewrite the passage to directly answer the question"*	45.0	48.7
#4	*"Transform the following passage. The core objective is to ensure the output is a background..."*	44.7	48.9
#5	*"From the input passage, extract the key information... Then rewrite this information..."*	44.3	48.5

Table 7: **Robustness to Prompt Variations.** Performance comparison using five distinct rewriting instructions. #1 denotes our default prompt. The low variance in metrics indicates that QREAM is robust to specific phrasing.

B Impact of Iteration Rounds in QREAM-ICL

We analyze how the number of rewriting rounds N affects the performance of QREAM-ICL. As illustrated in Figure 4, the performance initially improves as the model progressively refines the document structure. Notably, even a single rewriting iteration already delivers substantial improvement over standard RAG baseline that directly uses raw retrieved documents, demonstrating that QREAM can provide immediate benefits even in its simplest form. We observe that performance saturates at $N = 3$. Therefore, we set $N = 3$ to test the performance of QREAM-ICL and for data construction.

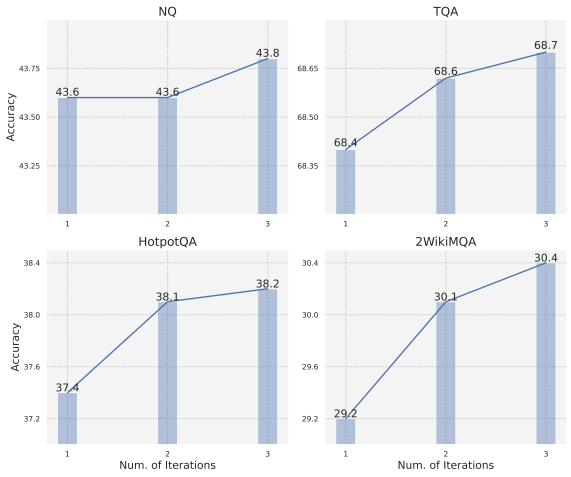

Figure 4: Performance of QREAM-ICL w.r.t. the number of rewriting iterations.

C Robustness to Prompt Variations

To ensure that the effectiveness of QREAM is not an artifact of specific prompt engineering, we evaluate QREAM-ICL with five distinct instruction variants, ranging from concise commands to detailed objective descriptions. Table 7 details the prompts and the results averaged across all datasets.

As observed, QREAM maintains consistent performance across all variations, with a fluctuation of less than 0.8 points in F1 score. Our default prompt (#1) yields the highest results, yet even the most distinct variant (#5, which emphasizes information extraction) significantly outperforms the baselines reported in the main paper. This stability suggests that the stylistic seeds play a more dominant role in guiding the generation than the specific wording of the instruction. Thus, QREAM proves to be robust to prompt phrasing, ensuring its applicability without the need for extensive prompt tuning.

D Effectiveness on Stronger Readers (GPT-5 mini)

We extend our evaluation to **GPT-5 mini** to verify if QREAM's benefits persist with state-of-the-art proprietary models. As shown in Table 8, while GPT-5 mini outperforms open-source baselines, its performance with raw retrieval is merely comparable to specialized frameworks like Self-RAG (e.g., 43.8% vs. 43.2% on NQ). This observation suggests that model scaling alone is insufficient to fully overcome the noise in raw retrieval. Specialized optimization strategies remain essential for SOTA LLMs.

Crucially, integrating **QREAM-FT** yields consistent gains over the raw baseline, particularly on complex multi-hop tasks (e.g., +4.6% Accuracy on HotpotQA). This confirms that QREAM acts as a **universal enhancer**: by restructuring evidence into a format easily usable for LLMs, it unlocks the potential of even the most powerful readers.

	NaturalQuestion		HotpotQA	
Method	**Acc**	**F1**	**Acc**	**F1**
GPT-5 mini	43.8	45.2	40.8	48.2
+ FaviComp	45.1	44.4	44.1	47.3
+ QREAM-FT	**46.6**	**46.2**	**45.4**	**53.8**

Table 8: **Performance with GPT-5 mini Reader.** QREAM consistently improves performance even for the strong proprietary model, particularly on complex multi-hop reasoning tasks.