BlendFilter: Advancing Retrieval-Augmented Large Language Models via Query Generation Blending and Knowledge Filtering

Haoyu Wang[†], Ruirui Li[⋆], Haoming Jiang[⋆], Jinjin Tian[⋆], Zhengyang Wang[⋆], Chen Luo[⋆], Xianfeng Tang[⋆], Monica Xiao Cheng[⋆], Tuo Zhao[*], Jing Gao[§]
[†]SUNY Albany, [§]Purdue University, [*]Georgia Institute of Technology, [⋆]Amazon
[†]hwang28@albany.edu, [§]jinggao@purdue.edu, [*]tourzhao@gatech.edu ,
[⋆]{ruirul,jhaoming,jinjint,zhengywa,cheluo,xianft,chengxca}@amazon.com

Abstract

Retrieval-augmented Large Language Models (LLMs) offer substantial benefits in enhancing performance across knowledge-intensive scenarios. However, these methods often face challenges with complex inputs and encounter difficulties due to noisy knowledge retrieval, notably hindering model effectiveness. To address this issue, we introduce BlendFilter, a novel approach that elevates retrieval-augmented LLMs by integrating query generation blending with knowledge filtering. BlendFilter proposes the blending process through its query generation method, which integrates both external and internal knowledge augmentation with the original query, ensuring comprehensive information gathering. Additionally, our distinctive knowledge filtering module capitalizes on the intrinsic capabilities of the LLM, effectively eliminating extraneous data. We conduct extensive experiments on three open-domain question answering benchmarks, and the findings clearly indicate that our innovative BlendFilter surpasses state-of-the-art baselines significantly.

1 Introduction

Generative Large Language Models (LLMs) have shown remarkable proficiency in various applications, such as summarization (Zhang et al., 2023; Wang et al., 2023a), dialogue systems (Hudeček and Dušek, 2023; Touvron et al., 2023a), and question answering (Lazaridou et al., 2022; Lu et al., 2022). Nonetheless, the finite scope of their pre-training corpora imposes inherent limitations, preventing LLMs from capturing and maintaining comprehensive worldly knowledge, especially given its dynamic nature. This limitation has spurred interest in retrieval-augmented generation strategies that integrate external knowledge sources, like Wikipedia, to refine the quality of LLM-generated content.

Typically, retrieval-augmented generation methods (Brown et al., 2020; Izacard et al., 2022b; Zakka et al., 2023) feed a task input, such as a user query or a question in open-domain question answering, into a retriever to obtain related knowledge documents. Subsequently, the LLM generates content based on the initial input and the information retrieved. Nevertheless, this direct retrieval strategy faces challenges with intricate task inputs (Shao et al., 2023). While straightforward queries enable effective identification of relevant information, multifaceted and complex questions may not cover some essential keywords, complicating the retrieval of pertinent documents.

To enhance the retrieval for complex task inputs, recent studies have proposed methods to enrich the original input. These approaches encompass question decomposition (Yao et al., 2022; Press et al., 2022), query rewriting (Ma et al., 2023), and query augmentation (Yu et al., 2023; Shao et al., 2023). They utilize knowledge memorized by LLMs or sourced from external databases to supplement the input with additional information, thereby explicitly incorporating additional keywords and substantially facilitating the retrieval process. Among these, query augmentation is particularly noteworthy and achieves state-of-the-art performance because it processes all retrieved knowledge collectively while generating answers and it does not require the training of an additional language model for query rewriting.

However, current query augmentation methods still suffer from some limitations. These techniques have typically relied on a single source of augmentation, either LLM internal knowledge or an external knowledge base. On one hand, for certain complex inputs, this single source of augmentation may not be able to cover all the keywords and thus lead to insufficient augmentation. Furthermore, existing work excludes original input but only rely on the augmented query, which could further exacerbate information loss.

Another major problem of existing methods is

Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, pages 1009–1025
November 12-16, 2024

that the incorporated content fetched by the retriever could contain irrelevant or misleading information. Usually top-K returned documents by the retriever will be used as augmentation, but there is no guarantee that all the top-K documents are relevant and helpful for the task. Correspondingly, incorporating such noise information into the augmented query can potentially lead to inaccuracies in the LLM's output (Wang et al., 2023b). To mitigate the noise in retrieved knowledge documents, previous studies (Yu et al., 2023; Wang et al., 2023b; Asai et al., 2023) have suggested various strategies. Unfortunately, these existing noise reduction methods in knowledge document retrieval are dependent on the LLM's confidence levels, which can be imprecise (Xiong et al., 2023). Additionally, these methods often require an extra language model to determine the need for retrieval, which incurs significant computational costs.

To tackle the aforementioned *complex question* and *noisy retrieved knowledge* challenges, we propose BlendFilter, a novel framework that advances retrieval-augmented large language models by integrating query generation blending and knowledge filtering, as illustrated in Fig. 1. Our framework, BlendFilter, is structured around three core components: 1) **Query Generation Blending** module, 2) **Knowledge Filtering** module, and a 3) Answer Generation module. The **Query Generation Blending** module is dedicated to enhancing input queries through diverse augmentation strategies, essentially forming a composite of queries, to handle the complex question challenge. This module incorporates both external and internal knowledge sources for augmentation. These augmented queries, including the original, external knowledge-augmented, and internal knowledge-augmented, are then employed by the retriever to collect pertinent information. In order to tackle the noise retrieved knowledge challenge, our proposed **Knowledge Filtering** module, aims to eliminate irrelevant retrieved knowledge and could operate autonomously without needing an extra language model, leveraging the innate filtering prowess of the LLM. In the final phase, the LLM integrates the filtered knowledge with the original query to generate the final answer.

The contributions are summarized as follows: 1) We introduce a novel query generation blending approach that integrates various augmentation sources. In contrast to existing work that relies on one source only, the proposed method enriches queries by using a variety of knowledge sources, which lead to a more comprehensive coverage of pertinent knowledge. 2) We present a novel and effective knowledge filtering module designed to eliminate irrelevant knowledge. We are the first to propose the utilization of the LLM itself as a filter in retrieval-augmented generation tasks. 3) We conduct extensive experiments across three open-domain question answering benchmarks. The results demonstrate that our proposed model, BlendFilter, significantly surpasses the baseline models across three distinct backbones.

2 Related Work

Retrieval-augmented generation enhances Large Language Models (LLMs) by leveraging external knowledge to improve generation quality. Initial approaches, as discussed in (Izacard and Grave, 2021; Shao and Huang, 2021; Izacard et al., 2022a; Shi et al., 2023), portrayed LLMs as passive recipients of retrieved knowledge, lacking interactive dynamics with retrievers. However, due to the inherent challenges in accurately capturing relevance between inputs and documents, these direct methods often yield only marginal improvements. Addressing this, recent advancements (Nakano et al., 2021; Trivedi et al., 2022; Jiang et al., 2023; Li et al., 2023b,a; Wang et al., 2023b; Asai et al., 2023; Yu et al., 2023; Ma et al., 2023; Press et al., 2022; Yao et al., 2022) have empowered LLMs to engage actively with retrievers, thereby enhancing relevance modeling. The integration of LLMs into the retrieval process broadly falls into three categories: 1) question decomposition, 2) query rewriting, and 3) query augmentation. For question decomposition, as exemplified by Yao et al. (2022) and Press et al. (2022), LLMs break down a complex question into simpler components, leveraging both previous interactions and retrieved knowledge. This decomposition facilitates more straightforward reasoning by LLMs. However, the success of this approach heavily depends on the LLM's capabilities. Insufficiently powerful LLMs might generate misleading sub-questions. Moreover, this method requires maintaining a historical context, potentially leading to lengthy dialogues and increased computational costs. In the realm of query rewriting, models are trained, often utilizing reinforcement learning, to reformulate the original question into a version more conducive to retrieval (Ma et al., 2023; Li et al., 2023b). These revised questions typ-

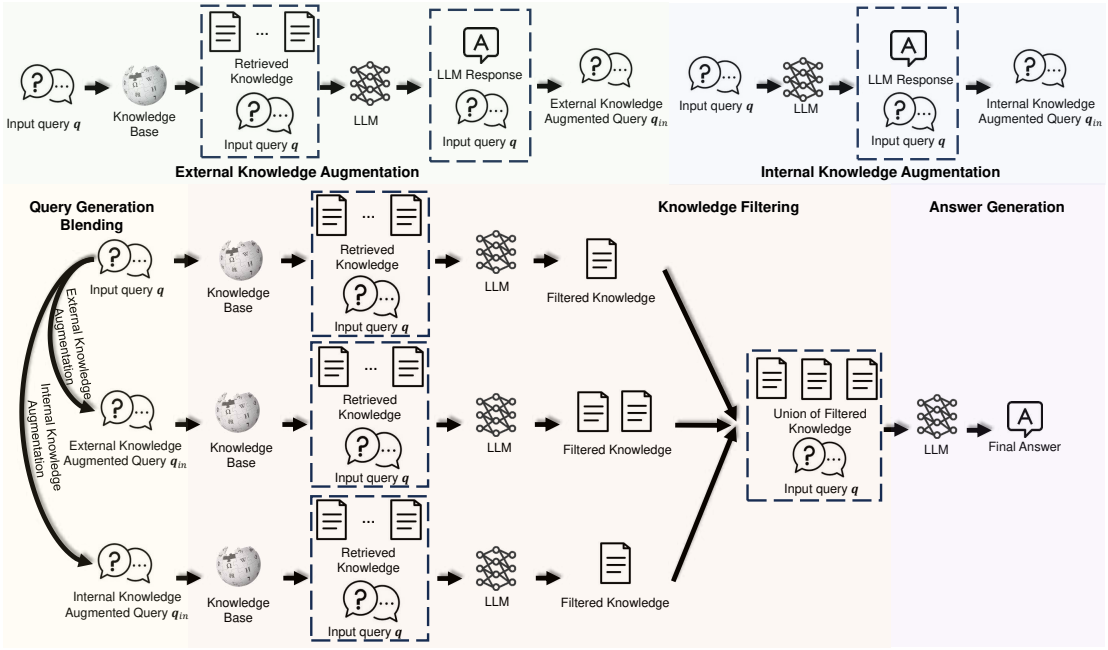


Figure 1: The framework of BlendFilter.

ically yield improved generation outcomes. Nevertheless, training an additional model for rewriting is a resource-intensive process. The third approach, query augmentation, involves enriching queries with knowledge from either LLM internal databases or external sources (Shao et al., 2023; Yu et al., 2023). A limitation of this method is its reliance on a single source of augmentation and often overlooking the original query, thus constraining overall model performance.

The aforementioned studies directly utilize retrieved knowledge, yet recent research (Wang et al., 2023b; Li et al., 2023a) highlights that such knowledge can sometimes be irrelevant or even detrimental to LLMs when answering queries. To solve this challenge, (Wang et al., 2023b) suggests an initial assessment to determine if LLMs need to retrieve knowledge, utilizing a classifier that could be based on BERT-like models or the LLM itself. However, this approach requires additional training data, which poses challenges in zero-shot or few-shot learning scenarios, and the LLM's self-evaluation may not always yield reliable results. (Asai et al., 2023) introduces a self-reflective method to ascertain the necessity of retrieval and to assess the relevance between the retrieved knowledge and the input. A critical limitation of this method, as noted by (Asai et al., 2023), is its dependence on training an auxiliary language model to produce text with reflection tokens, incurring extra costs. Additionally, (Yu et al., 2023) employs a strategy of comparing the average negative likelihood of answers with and without external knowledge to guide decision-making. Nevertheless, this measure may not be a precise indicator of model confidence and is not universally applicable across models, with certain models like GPT-3.5-turbo and GPT-3.5-turbo-Instruct currently unable to access this feature. We summarize the **differences** between the proposed BlendFilter and other baselines in Table 6 in the appendix.

3 Methodology

Given a pre-trained Large Language Model (LLM) $\mathcal{M}(\cdot)$, a knowledge base $\mathcal{K} = \{\mathcal{K}_i\}_{i=1}^{n}$ (where n represents the number of documents), a retriever $\mathcal{R}(\cdot)$, and a query $\boldsymbol{q}$, our objective is to utilize the knowledge base to facilitate accurate responses from the LLM without fine-tuning.

3.1 Overview

To enhance the retrieval quality for retrieval-augmented LLMs, we introduce a framework named BlendFilter, which incorporates query generation blending and knowledge filtering, as depicted in Fig. 1. We begin by presenting query blending, a technique that enhances the original query by incorporating both external knowledge and the LLM's internally memorized knowledge (Section 3.2). Additionally, we propose a knowledge filtering module to effectively remove irrelevant knowledge (Section 3.3). Finally, we demonstrate how the LLM generates answers based on

the filtered knowledge (Section 3.4).

3.2 Query Generation Blending

Numerous studies (Izacard and Grave, 2021; Shao and Huang, 2021; Izacard et al., 2022a; Shi et al., 2023) have validated the effectiveness of utilizing a retriever to enrich questions with relevant knowledge, thereby boosting the performance of LLMs. This process can be represented as follows: $\mathcal{K}_r = \mathcal{R}(\boldsymbol{q}, \mathcal{K}; K)$, $\boldsymbol{a} = \mathcal{M}(\boldsymbol{a}|\texttt{Prompt}(\boldsymbol{q}, \mathcal{K}_r))$, where $\boldsymbol{a}$ represents the generated answer, $\mathcal{K}_r$ denotes the retrieved knowledge, and K serves as the hyper-parameter for the retriever, controlling the quantity of retrieved knowledge items. Nonetheless, in cases where the query is complex, directly inputting it into the retriever often fails to retrieve the correct knowledge documents. As a solution, we advocate for the incorporation of both external and internal knowledge augmentation techniques to refine the query.

External Knowledge Augmentation. For complex questions, such as those in multi-hop question answering (Yang et al., 2018), which often entail implicit sub-problems and span multiple knowledge domains, we utilize an external knowledge base to refine the original query and facilitate document retrieval. Specifically, we initially retrieve relevant knowledge documents using the original query, as follows: $\mathcal{K}_{ex} = \mathcal{R}(\boldsymbol{q}, \mathcal{K}; K)$.

Subsequently, we engage the LLM to derive the answer using the acquired knowledge documents via the Chain-of-Thought (CoT) approach (Wei et al., 2022). This step is depicted as: $\boldsymbol{a}_{ex} = \mathcal{M}(\boldsymbol{a}|\texttt{Prompt}_{\texttt{CoT}}(\boldsymbol{q}, \mathcal{K}_{ex}))$, where $\boldsymbol{a}_{ex}$ represents the reasoning and answer generated by the LLM based on the retrieved knowledge $\mathcal{K}_{ex}$. The generated context $\boldsymbol{a}_{ex}$ contains related keywords and valuable information through CoT reasoning based on retrieved knowledge from the external knowledge base, thereby assisting the retriever in pinpointing relevant knowledge. Subsequently, we integrate the generated context $\boldsymbol{a}_{ex}$ with the initial query $\boldsymbol{q}$ to formulate the enhanced query, as shown below: $\boldsymbol{q}_{ex} = \boldsymbol{a}_{ex} \| \boldsymbol{q}$, where $\|$ represents the concatenation operation.

Remark 1. This process of external knowledge augmentation essentially acts as a two-hop reasoning mechanism to refine the query. In fact, it can be extended to higher-order augmentation, but typically, leveraging two-hop information proves to be sufficiently effective in enhancing retrieval accuracy due to the LLM's strong capabilities. Consequently, we refrain from employing higher-order augmentation in order to strike a balance between efficiency and accuracy.

Internal Knowledge Augmentation. LLMs have memorized a lot of factual knowledge. Some related knowledge is not retrieved in external knowledge augmentation while LLMs may memorize them internally. Consequently, we can prompt the LLM to produce a detailed response to the query, drawing upon its internal knowledge. This internally-sourced response acts as a supplement to the external knowledge. Specifically, the generated text based on LLM internal knowledge can be formulated as $\boldsymbol{a}_{in} = \mathcal{M}(\boldsymbol{a}|\texttt{Prompt}(\boldsymbol{q}))$, and the augmented query is $\boldsymbol{q}_{in} = \boldsymbol{a}_{in} \| \boldsymbol{q}$.

3.3 Knowledge Filtering

By integrating both external and internal knowledge-augmented queries in conjunction with the original query, we are able to retrieve the corresponding knowledge documents separately, as follows: $\mathcal{K}_q = \mathcal{R}(\boldsymbol{q}, \mathcal{K}; K)$, $\mathcal{K}_{q_{ex}} = \mathcal{R}(\boldsymbol{q}_{ex}, \mathcal{K}; K)$, $\mathcal{K}_{q_{in}} = \mathcal{R}(\boldsymbol{q}_{in}, \mathcal{K}; K)$, where $\mathcal{K}_q$ represents knowledge documents retrieved by the original query, $\mathcal{K}_{q_{ex}}$ corresponds to the external knowledge-augmented query, and $\mathcal{K}_{q_{in}}$ pertains to the internal knowledge-augmented query. A direct approach to leveraging this retrieved knowledge involves taking their union: $\mathcal{K}_r^{direct} = \mathcal{K}_q \bigcup \mathcal{K}_{q_{ex}} \bigcup \mathcal{K}_{q_{in}}$.

This method ensures that the synthesized knowledge, $\mathcal{K}_r^{direct}$, encompasses a broader spectrum of relevant documents, thereby enhancing the quality of the retrieved knowledge. Nonetheless, retrieving some unrelated documents is inevitable due to the inherent imperfections of the retrieval process and the selection of the top-K documents, which may include irrelevant information when K exceeds the number of ground truth knowledge documents. This unrelated information can potentially lead to confusion and misguidance for the LLM, resulting in incorrect outputs. Rather than training a separate knowledge filter to identify and eliminate unrelated information, we have observed that the LLM itself serves as an effective knowledge filter. We provide both the original query and the retrieved knowledge to the Large Language Model (LLM) and instruct the LLM to perform knowledge filtering. This can be formulated as follows: $\mathcal{K}_q^f = \mathcal{M}(\mathcal{K}|\texttt{Prompt}(\boldsymbol{q}, \mathcal{K}_q))$, $\mathcal{K}_{q_{ex}}^f = \mathcal{M}(\mathcal{K}|\texttt{Prompt}(\boldsymbol{q}, \mathcal{K}_{q_{ex}}))$, $\mathcal{K}_{q_{in}}^f = \mathcal{M}(\mathcal{K}|\texttt{Prompt}(\boldsymbol{q}, \mathcal{K}_{q_{in}}))$. The final knowledge uti-

lized for generation is obtained by taking the union of the filtered knowledge sets, i.e. $\mathcal{K}_r = \mathcal{K}_q^f \bigcup \mathcal{K}_{q_{ex}}^f \bigcup \mathcal{K}_{q_{in}}^f$, where $\bigcup$ represents taking union operation.

Remark 2. Our method involves filtering knowledge and subsequently combining the filtered information. An alternative option is to reverse the sequence of these two steps. However, we have observed that commencing with the union of knowledge may result in a larger knowledge set, consequently intensifying the challenge of subsequent knowledge filtering. Consequently, we opt to filter knowledge independently for $\mathcal{K}_q$, $\mathcal{K}_{q_{ex}}$, and $\mathcal{K}_{q_{in}}$.

3.4 Answer Generation

In this step, the LLM generates an answer based on both the filtered knowledge and the original query. We employ CoT to enhance the model's reasoning performance, a representation of which is as follows: $\boldsymbol{a} = \mathcal{M}(\boldsymbol{a}|\texttt{Prompt}_{\texttt{CoT}}(\boldsymbol{q}, \mathcal{K}_r))$. The whole algorithm is summarized in Algorithm 1 in the appendix.

4 Experiment

In this section, we evaluate the proposed BlendFilter and answer the following research questions: **RQ1**) How does BlendFilter perform compared to state-of-the-art retrieval-augmented baselines? **RQ2**) Can the proposed BlendFilter generalize well with respect to different backbones and retrievers? **RQ3**) Is the LLM effective to filter unrelated knowledge documents? **RQ4**) What are the roles of the original query, external knowledge-augmented query, and internal knowledge-augmented query in model performance improvements respectively? **RQ5**) How does the performance change with varying numbers of knowledge documents? **RQ6**) Will the proposed BlendFilter be improved by sampling multiple times with different temperatures?

4.1 Datasets and Experiment Settings

4.1.1 Datasets

We conduct experiments on three public benchmarks, including HotPotQA (Yang et al., 2018), 2WikiMultiHopQA (Ho et al., 2020), and StrategyQA (Geva et al., 2021). Examples are illustrated in Fig. 5 in the appendix.

4.1.2 Evaluation Metrics

Following Shao et al. (2023), we evaluate the first 500 questions from the training dataset for StrategyQA and 500 questions from the development dataset for HotPotQA and 2WikiMultiHopQA. For multi-hop question answering datasets, we employ exact match (EM) and F1 as evaluation metrics, and for the commonsense reasoning dataset, we use accuracy, following Yao et al. (2022) and Shao et al. (2023). To evaluate the retrieval performance, we leverage widely used Recall and Precision as evaluation metrics. Additionally, to assess the effectiveness of the proposed knowledge filtering in eliminating irrelevant information, we introduce a new metric called S-Precision. This metric measures the proportion of questions for which the retrieved documents precisely match the golden relevant documents.

4.1.3 Baselines

We adopt following state-of-the-art baselines to evaluate our proposed BlendFilter: 1) Direct Prompting (Brown et al., 2020), 2) CoT Prompting (Wei et al., 2022), 3) ReAct (Yao et al., 2022), 4) SelfAsk (Press et al., 2022), and 5) ITER-RETGEN (Shao et al., 2023). We show the detail information about these baselines in the appendix.

4.1.4 Implementation Details.

We evaluate models with three different LLMs: GPT3.5-turbo-Instruct[1], Vicuna 1.5-13b (Zheng et al., 2023), and Qwen-7b (Bai et al., 2023). We utilize the state-of-the-art efficient retrieval method ColBERT v2 (Santhanam et al., 2022) as the retriever implemented by Khattab et al. (2022, 2023). The knowledge base we employ is the collection of Wikipedia abstracts dumped in 2017 (Khattab et al., 2023). We show the detailed information about implementation details in the appendix.

4.2 Performance Comparison

In this section, we evaluate the performance of both the baseline models and our proposed BlendFilter model using various backbones. The results are displayed in Table 1, Table 2, and Table 3, addressing **RQ1** and **RQ2**.

The performance results in the tables demonstrate that our proposed BlendFilter consistently achieves substantial improvements over the baselines across different backbones and datasets. Remarkably, our BlendFilter model achieves average performance improvements of 9.7%, 7.4%, and 14.2% when using GPT3.5-turbo-Instruct, Vicuna 1.5-13b, and Qwen-7b as backbones, respectively. These results demonstrate the effectiveness of our proposed BlendFilter in enhancing retrieval-

[1] https://platform.openai.com/docs/models/gpt-3-5

Table 1: Performance of BlendFilter with GPT3.5-turbo-Instruct as the backbone. IMP represents the percentage of improvements compared to baselines with respect to Exact Match on HotPotQA and 2WikiMultihopQA and Accuracy on StrategyQA.

Method	HotPotQA			2WikiMultihopQA			StrategyQA	
	Exact Match	F1	IMP	Exact Match	F1	IMP	Accuracy	IMP
Without Retrieval								
Direct	0.304	0.410	67.1%	0.282	0.318	43.3%	0.648	14.8%
CoT	0.302	0.432	68.2%	0.300	0.403	34.7%	0.700	6.3%
With Retrieval								
Direct	0.412	0.537	23.3%	0.318	0.371	27.0%	0.634	17.4%
CoT	0.434	0.558	17.1%	0.318	0.396	27.0%	0.616	20.8%
ReAct	0.360	0.475	41.1%	0.374	0.450	8.0%	0.658	13.1%
SelfAsk	0.364	0.481	39.6%	0.334	0.416	21.0%	0.638	16.6%
ITER-RETGEN	0.450	0.572	12.9%	0.328	0.436	23.2%	0.692	7.5%
BlendFilter	0.508	0.624	-	0.404	0.470	-	0.744	-

Table 2: Performance of BlendFilter with Vicuna 1.5-13b as the backbone.

Method	HotPotQA			2WikiMultihopQA			StrategyQA	
	Exact Match	F1	IMP	Exact Match	F1	IMP	Accuracy	IMP
Without Retrieval								
Direct	0.202	0.267	96.0%	0.246	0.288	16.3%	0.604	11.3%
CoT	0.228	0.344	73.7%	0.190	0.279	50.5%	0.660	1.8%
With Retrieval								
Direct	0.336	0.443	17.9%	0.210	0.284	36.2%	0.624	7.7%
CoT	0.362	0.488	9.4%	0.206	0.302	38.8%	0.646	4.0%
ReAct	0.332	0.463	19.3%	0.216	0.323	32.4%	0.588	14.3%
SelfAsk	0.361	0.469	9.7%	0.250	0.376	14.4%	0.618	8.7%
ITER-RETGEN	0.366	0.484	8.2%	0.252	0.3551	13.5%	0.668	0.6%
BlendFilter	0.396	0.527	-	0.286	0.378	-	0.672	-

Table 3: Performance of BlendFilter with Qwen-7b as the backbone.

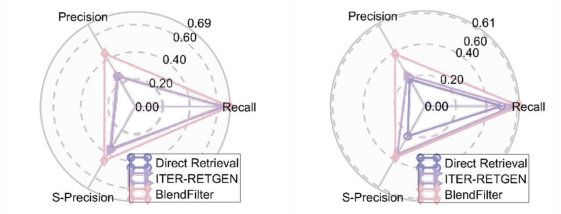
Method	HotPotQA			2WikiMultihopQA			StrategyQA	
	Exact Match	F1	IMP	Exact Match	F1	IMP	Accuracy	IMP
Without Retrieval								
Direct	0.144	0.238	118.1%	0.182	0.244	31.9%	0.630	4.1%
CoT	0.150	0.245	109.3%	0.180	0.246	33.3%	0.658	-0.3%
With Retrieval								
Direct	0.180	0.310	74.4%	0.084	0.200	185.7%	0.572	14.6%
CoT	0.206	0.305	52.4%	0.210	0.292	14.3%	0.604	8.6%
ReAct	0.142	0.239	121.1%	0.158	0.241	51.9%	0.592	10.8%
SelfAsk	0.206	0.307	52.4%	0.106	0.154	126.4%	0.596	10.1%
ITER-RETGEN	0.244	0.364	28.7%	0.200	0.297	20.0%	0.612	7.2%
BlendFilter	0.314	0.442	-	0.240	0.312	-	0.656	-

augmented generation performance and its ability to generalize across various backbones.

It is worth noting that mere retrieval does not consistently enhance accuracy. For instance, when comparing CoT with retrieval and CoT without retrieval using GPT3.5-turbo-Instruct on 2WikiMultihopQA (as shown in Table 1), CoT without retrieval exhibits a higher Exact Match score than CoT with retrieval. This observation suggests that the retrieved knowledge documents may include unrelated information, which can lead to misleading the LLM. This observation aligns with one of our underlying motivations.

4.3 Combining with BM25

In this section, we utilize BM25 (Jones et al., 2000), a widely-used sparse retriever, to explore **RQ2** on the HotPotQA dataset. The results are shown in Table 4. When comparing the results in Table 4



(a) ColBERT v2 (b) BM25

Figure 2: Retrieval performance after knowledge filtering with GPT3.5-turbo-Instruct on HotPotQA.

Table 4: Performance of BlendFilter with GPT3.5-turbo-Instruct and BM25 on HotPotQA.

Method	Exact Match	F1
Without Retrieval		
Direct	0.304	0.410
CoT	0.302	0.432
With Retrieval (BM25)		
Direct	0.342	0.462
CoT	0.348	0.470
ReAct	0.280	0.371
SelfAsk	0.290	0.393
ITER-RETGEN	0.356	0.488
BlendFilter	0.420	0.547

with those in Table 1, it becomes evident that utilizing ColBERT v2, a dense retriever, yields superior performance compared to BM25. Dense retrievers prove more effective in capturing semantic similarities between questions and documents, especially for complex queries. Moreover, our proposed BlendFilter consistently outperforms the baselines when BM25 serves as the retriever as well. The proposed BlendFilter achieves an improvement of approximately 18%, surpassing the performance when ColBERT v2 is employed as the retriever, in comparison to the baseline models. One potential explanation is that BM25 lacks the potency of ColBERT v2, making the application of query blending to ensure the explicit inclusion of keywords in queries a more crucial factor. This highlights the effectiveness of our proposed BlendFilter across different retrievers.

4.4 Effectiveness for Retrieval

In this section, we address **RQ3** by computing Precision, Recall, and S-Precision values after conducting knowledge filtering with GPT3.5-turbo-Instruct on the HotPotQA dataset. Results are presented in Figure 2. As indicated in Fig. 2, the proposed BlendFilter leads to a substantial improvement in retrieval performance. In both ColBERT v2 and BM25 scenarios, the proposed BlendFilter demonstrates superior retrieval accuracy compared to direct retrieval and ITER-RETGEN (multi-hop retrieval). Furthermore, when comparing the Recall between ITER-RETGEN and BlendFilter, it becomes evident that the proposed query blending is effective. This illustrates that combining three queries can recall a greater number of related documents. When comparing the Precision and S-Precision of the baselines with those of BlendFilter, we observe that the proposed knowledge filtering effectively eliminates unrelated documents.

4.5 Effectiveness of Different Queries

In this section, we investigate how performance changes when removing specific queries from the query blending module, addressing **RQ4**. The results are shown in Table 5. According to Table 5, it is evident that removing any query from the query blending process results in thedegradation in model performance. This demonstrates the importance of the original query, the externally augmented query, and the internally augmented query in the answer generation process. Additionally, we can find the internal knowledge-augmented query plays a more important role when BM25 is employed. One possible explanation is that when BM25 is used, the retrieval accuracy is not as robust as that of a dense retriever. Consequently, the externally augmented query may still miss some information. This highlights the importance of complementing it with internal knowledge augmentation.

Table 5: Performance of BlendFilter without different queries with GPT3.5-turbo-Instruct on HotPotQA.

Method	Exact Match	F1
Dense Retriever (ColBERT v2)		
BlendFilter	0.508	0.624
w/o q	0.476	0.604
w/o q_{ex}	0.442	0.565
w/o q_{in}	0.496	0.613
Sparse Retriever (BM25)		
BlendFilter	0.420	0.547
w/o q	0.410	0.532
w/o q_{ex}	0.388	0.506
w/o q_{in}	0.398	0.514

4.6 Number of Retrieved Documents

In this section, we explore how the model's performance varies when employing different numbers of retrieved documents (K), addressing **RQ5**. The re-

sults are presented in Fig. 3. Based on Fig. 3, it can be observed that as the value of K is increased, the performance of both ITER-RETGEN and BlendFilter initially improves and then experiences a slight decline. This indicates that increasing the number of retrieved knowledge documents appropriately can enhance model performance. Notably, it is evident that increasing the value of K from 3 to 8 leads to a substantial improvement in the performance of BlendFilter, while ITER-RETGEN exhibits only marginal performance gains. One possible explanation is that BlendFilter incorporates knowledge filtering, effectively eliminating most unrelated knowledge, whereas ITER-RETGEN lacks this filtering mechanism and incorporates a significant amount of noise knowledge.

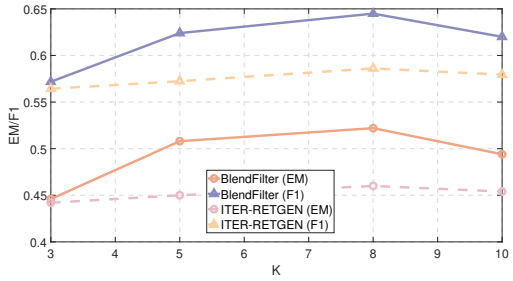


Figure 3: Performance with respect to different K values on HotPotQA with GPT3.5-turbo-Instruct.

4.7 Sampling Times

In this section, we employ various sampling temperatures for the GPT3.5-turbo-Instruct, specifically $top\_p = 0, 0.5, 1$, and sample one answer under each temperature setting on HotPotQA dataset to address **RQ6**. The results are shown in Fig. 4. Based on Fig. 4, it is evident that our proposed BlendFilter consistently outperforms the baselines, whether sampling a single answer or multiple answers. Furthermore, when three answers are sampled, all methods exhibit improvements, albeit the improvements in the case of BlendFilter are notably smaller compared to the other baseline methods. This observation demonstrates that when provided with more opportunities to answer, all these models tend to have a higher probability of answering correctly, whereas our proposed BlendFilter exhibits lower variance.

4.8 Case Study

In this section, we show a concrete example in Fig. 6 in the appendix to show how the proposed BlendFilter works. This example is taken from HotPotQA dataset and we feed it to GPT3.5-turbo-Instruct. The original question is "*superMansion starred the actress who had a recurring role as whom on Workaholics?*". The related knowledge includes the *SuperMasion* document and *Jillian Bell* document. From Fig. 6, we can find both the original query and external knowledge-augmented query retrieved knowledge consists of one correct document *SuperMasion*. Additionally, the internal knowledge-augmented query retrieves another correct knowledge document *Jillian Bell*. This demonstrates the necessity of combining these three queries to retrieve all relevant knowledge documents. Furthermore, following knowledge filtering, our proposed BlendFilter effectively eliminates all irrelevant documents and provides the correct answer to the question.

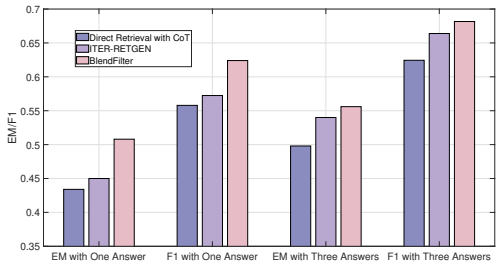


Figure 4: Performance of models with multiple answer sampling on HotPotQA with GPT3.5-turbo-Instruct. For three answers, if one of the answers is correct, its EM will be 1, and the F1 score is the highest one of the three answers.

5 Conclusion

In this paper, we introduce BlendFilter, a comprehensive framework developed to enhance retrieval-augmented generation within LLMs. Our methodology distinctively incorporates query generation blending and knowledge filtering techniques, effectively tackling the intricacies of complex inputs and significantly reducing noise in retrieved knowledge. The amalgamation of external and internal knowledge augmentation fosters a resilient and all-encompassing retrieval mechanism. Additionally, our innovative self-reliant knowledge filtering module exploits the inherent capabilities of the LLM to refine and purify the retrieved knowledge by eliminating extraneous content. We conducted extensive experiments on three benchmarks, and the results demonstrate that BlendFilter outperforms state-of-the-art baselines. Moreover, BlendFilter can be generalized well for different kinds LLMs, including GPT3.5-turbo-Instruct, Vicuna 1.5-13b and Qwen-7b.

Limitations

The proposed BlendFilter framework introduces a hyper-parameter K to control how many documents we need to retrieve, which might require additional effort to tune. Fortunately, we observe that the model performance is not very sensitive to the hyper-parameter and we set it to a fixed value to achieve a good performance in this paper.

Acknowledgement

This work is supported in part by the US National Science Foundation under grant NSF IIS-1747614 and NSF IIS-2141037. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

References

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

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, Binyuan Hui, Luo Ji, Mei Li, Junyang Lin, Runji Lin, Dayiheng Liu, Gao Liu, Chengqiang Lu, Keming Lu, Jianxin Ma, Rui Men, Xingzhang Ren, Xuancheng Ren, Chuanqi Tan, Sinan Tan, Jianhong Tu, Peng Wang, Shijie Wang, Wei Wang, Shengguang Wu, Benfeng Xu, Jin Xu, An Yang, Hao Yang, Jian Yang, Shusheng Yang, Yang Yao, Bowen Yu, Hongyi Yuan, Zheng Yuan, Jianwei Zhang, Xingxuan Zhang, Yichang Zhang, Zhenru Zhang, Chang Zhou, Jingren Zhou, Xiaohuan Zhou, and Tianhang Zhu. 2023. Qwen technical report. *arXiv preprint arXiv:2309.16609*.

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

Mor Geva, Daniel Khashabi, Elad Segal, Tushar Khot, Dan Roth, and Jonathan Berant. 2021. Did aristotle use a laptop? a question answering benchmark with implicit reasoning strategies. *Transactions of the Association for Computational Linguistics*, 9:346–361.

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.

Vojtěch Hudeček and Ondřej Dušek. 2023. Are large language models all you need for task-oriented dialogue? In *Proceedings of the 24th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 216–228.

Gautier Izacard and Édouard Grave. 2021. Leveraging passage retrieval with generative models for open domain question answering. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 874–880.

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

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

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

K Sparck Jones, Steve Walker, and Stephen E. Robertson. 2000. A probabilistic model of information retrieval: development and comparative experiments: Part 2. *Information processing & management*, 36(6):809–840.

Omar Khattab, Keshav Santhanam, Xiang Lisa Li, David Hall, Percy Liang, Christopher Potts, and Matei Zaharia. 2022. Demonstrate-search-predict: Composing retrieval and language models for knowledge-intensive NLP. *arXiv preprint arXiv:2212.14024*.

Omar Khattab, Arnav Singhvi, Paridhi Maheshwari, Zhiyuan Zhang, Keshav Santhanam, Sri Vardhamanan, Saiful Haq, Ashutosh Sharma, Thomas T. Joshi, Hanna Moazam, Heather Miller, Matei Zaharia, and Christopher Potts. 2023. Dspy: Compiling declarative language model calls into self-improving pipelines. *arXiv preprint arXiv:2310.03714*.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*.

Angeliki Lazaridou, Elena Gribovskaya, Wojciech Stokowiec, and Nikolai Grigorev. 2022. Internet-augmented language models through few-shot

prompting for open-domain question answering. *arXiv preprint arXiv:2203.05115*.

Xiaonan Li, Changtai Zhu, Linyang Li, Zhangyue Yin, Tianxiang Sun, and Xipeng Qiu. 2023a. Llatrieval: Llm-verified retrieval for verifiable generation. *arXiv preprint arXiv:2311.07838*.

Xiaopeng Li, Lixin Su, Pengyue Jia, Xiangyu Zhao, Suqi Cheng, Junfeng Wang, and Dawei Yin. 2023b. Agent4ranking: Semantic robust ranking via personalized query rewriting using multi-agent llm. *arXiv preprint arXiv:2312.15450*.

Pan Lu, Swaroop Mishra, Tanglin Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. 2022. Learn to explain: Multimodal reasoning via thought chains for science question answering. *Advances in Neural Information Processing Systems*, 35:2507–2521.

Xinbei Ma, Yeyun Gong, Pengcheng He, Hai Zhao, and Nan Duan. 2023. Query rewriting for retrieval-augmented large language models. *arXiv preprint arXiv:2305.14283*.

Reiichiro Nakano, Jacob Hilton, Suchir Balaji, Jeff Wu, Long Ouyang, Christina Kim, Christopher Hesse, Shantanu Jain, Vineet Kosaraju, William Saunders, et al. 2021. Webgpt: Browser-assisted question-answering with human feedback. *arXiv preprint arXiv:2112.09332*.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744.

Ofir Press, Muru Zhang, Sewon Min, Ludwig Schmidt, Noah A Smith, and Mike Lewis. 2022. Measuring and narrowing the compositionality gap in language models. *arXiv preprint arXiv:2210.03350*.

Keshav Santhanam, Omar Khattab, Jon Saad-Falcon, Christopher Potts, and Matei Zaharia. 2022. Colbertv2: Effective and efficient retrieval via lightweight late interaction. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3715–3734.

Zhihong Shao, Yeyun Gong, Yelong Shen, Minlie Huang, Nan Duan, and Weizhu Chen. 2023. Enhancing retrieval-augmented large language models with iterative retrieval-generation synergy. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 9248–9274. Association for Computational Linguistics.

Zhihong Shao and Minlie Huang. 2021. Answering open-domain multi-answer questions via a recall-then-verify framework. *arXiv preprint arXiv:2110.08544*.

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 preprint arXiv:2301.12652*.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023a. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023b. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. 2022. Interleaving retrieval with chain-of-thought reasoning for knowledge-intensive multi-step questions. *arXiv preprint arXiv:2212.10509*.

Jiaan Wang, Yunlong Liang, Fandong Meng, Beiqi Zou, Zhixu Li, Jianfeng Qu, and Jie Zhou. 2023a. Zero-shot cross-lingual summarization via large language models. In *Proceedings of the 4th New Frontiers in Summarization Workshop*, pages 12–23.

Yile Wang, Peng Li, Maosong Sun, and Yang Liu. 2023b. Self-knowledge guided retrieval augmentation for large language models. In *The 2023 Conference on Empirical Methods in Natural Language Processing*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35:24824–24837.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 conference on empirical methods in natural language processing: system demonstrations*, pages 38–45.

Miao Xiong, Zhiyuan Hu, Xinyang Lu, Yifei Li, Jie Fu, Junxian He, and Bryan Hooi. 2023. Can llms express their uncertainty? an empirical evaluation of confidence elicitation in llms. *arXiv preprint arXiv:2306.13063*.

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.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R Narasimhan, and Yuan Cao. 2022. React: Synergizing reasoning and acting in language models. In *The Eleventh International Conference on Learning Representations*.

Ori Yoran, Tomer Wolfson, Ben Bogin, Uri Katz, Daniel Deutch, and Jonathan Berant. 2023. Answering questions by meta-reasoning over multiple chains of thought.

Wenhao Yu, Zhihan Zhang, Zhenwen Liang, Meng Jiang, and Ashish Sabharwal. 2023. Improving language models via plug-and-play retrieval feedback. *arXiv preprint arXiv:2305.14002*.

Cyril Zakka, Akash Chaurasia, Rohan Shad, Alex R Dalal, Jennifer L Kim, Michael Moor, Kevin Alexander, Euan Ashley, Jack Boyd, Kathleen Boyd, et al. 2023. Almanac: Retrieval-augmented language models for clinical medicine. *Research Square*.

Tianyi Zhang, Faisal Ladhak, Esin Durmus, Percy Liang, Kathleen McKeown, and Tatsunori B Hashimoto. 2023. Benchmarking large language models for news summarization. *arXiv preprint arXiv:2301.13848*.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *arXiv preprint arXiv:2306.05685*.

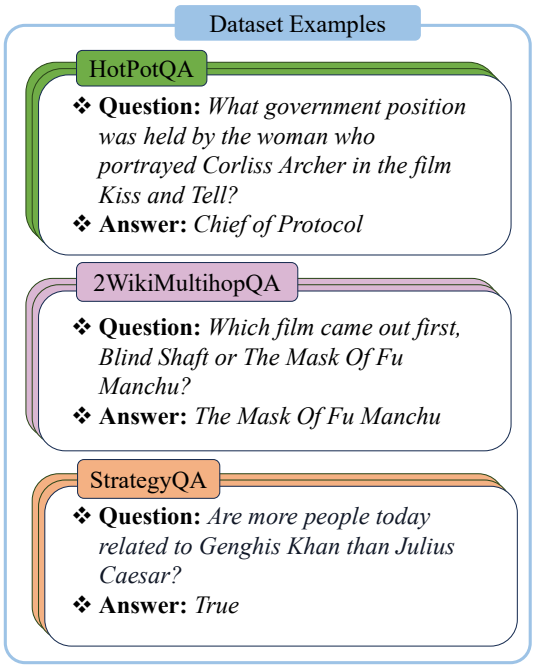


Figure 5: Examples of datasets.

A Related Work

We the differences bettwen the proposed BlendFilter and existing baselines in Table 6.

B Algorithm

C Baselines

We adopt following state-of-the-art baselines to evaluate our proposed BlendFilter:

- Direct Prompting (Brown et al., 2020) instructs the LLM to provide direct answers to questions without offering explanations or explicit reasoning steps. We evaluate both Direct Prompting with and without retrieval as our baseline approaches, referring to them as Direct for brevity.

- CoT Prompting (Wei et al., 2022) instructs the LLM to generate answers accompanied by explicit reasoning steps. Similar to Direct Prompting, we evaluate CoT Prompting with and without retrieval, referring to them as CoT in our experiments.

- ReAct (Yao et al., 2022) incorporates reasoning, action, and observation steps. The generation process concludes upon reaching the finishing state. The action can involve either generating a query to retrieve knowledge or finalizing the generation. The observation entails the retrieved knowledge documents.

- SelfAsk (Press et al., 2022) comprises steps for follow-up question generation, retrieval, and an-

Table 6: The differences between the proposed BlendFilter and existing methods.

	Query Decomposition	Query Rewriting	Query Augmentation		Knowledge Selection			Need Traing
			External Knowledge	Internal Knowledge	Predicting Before Retrieval	Model Confidence	Filtering	
ReAct Yao et al. (2022)	✓	–	–	–	–	–	–	✗
Ma et al. (2023)	–	✓	–	–	–	–	–	✓
Yu et al. (2023)	–	–	–	✓	–	✓	–	✗
ITER-RETGEN (Shao et al., 2023)	–	–	✓	–	–	–	–	✗
Asai et al. (2023)	–	–	–	–	✓	–	–	✓
Wang et al. (2023b)	–	–	–	–	✓	–	–	✓
BlendFilter	–	–	✓	✓	–	–	✓	✗

Question: *superMansion starred the actress who had a recurring role as whom on Workaholics?*

Knowledge Preparation

Original Query: *superMansion starred the actress who had a recurring role as whom on Workaholics?*
Retrieved Knowledge:
❖ **SuperMansion** | SuperMansion is an American stop-motion … The series premiered on Crackle on October 8, 2015.
❖ **Superman** (1987 film) | Superman is a … Puneet Issar in lead role as Superman.
❖ **Joan Alexander** | Joan Alexander … radio serial "The Adventures of Superman" (1940–1951).
❖ **Superman and the Mole Men** | Superman and the Mole Men … The film was released by Lippert Pictures Inc.
❖ **Sarah Douglas** | Sarah Douglas (born 12 December 1952) is an English actress … drama series "Falcon Crest" (1983–85).

External Knowledge Augmentation Query: *SuperMansion starred Bryan Cranston, who had a recurring role as the boss on Workaholics. superMansion starred the actress who had a recurring role as whom on Workaholics?*
Retrieved Knowledge:
❖ **SuperMansion** | SuperMansion is an American stop-motion … The series premiered on Crackle on October 8, 2015.
❖ **Superman and the Mole Men** | Superman and the Mole Men … The film was released by Lippert Pictures Inc.
❖ **Superman (1987 film)** | Superman is a … Puneet Issar in lead role as Superman.
❖ **Atom Man vs. Superman** | Atom Man vs. Superman (1950), … to cover the story.
❖ **Superman Returns** | Superman Returns is a 2006 American superhero film … Superman and the world.

Internal Knowledge Augmentation Query: *The actress who had a recurring role as whom on Workaholics … superMansion starred the actress who had a recurring role as whom on Workaholics?*
Retrieved Knowledge:
❖ **Gillian Jacobs** | Gillian MacLaren Jacobs (; born October 19, 1982) is an American actress … and "Brother Nature" (2016).
❖ **Jillian Bell** | Jillian Leigh Bell (born April 25, 1984) is an American comedian, actress, and screenwriter. She is best known for her recurring roles as Jillian Belk on "Workaholics“ … "Fist Fight" (2017).
❖ **Gillian Vigman** | Gillian Vigman (born January 28, 1972) is an American comic actress. … role on "The Defenders".
❖ **Gillian Jones** | Gillian Jones … drama "Packed to the Rafters" since 2009.
❖ **Jan Hooks** | Janet Vivian "Jan" Hooks … roles in film and television.

Answer Generation

Question: *superMansion starred the actress who had a recurring role as whom on Workaholics?*
Knowledge:
SuperMansion | SuperMansion is an American stop-motion … The series premiered on Crackle on October 8, 2015.
Jillian Bell | Jillian Leigh Bell (born April 25, 1984) is an American comedian, actress, and screenwriter. She is best known for her recurring roles as Jillian Belk on "Workaholics“ … "Fist Fight" (2017).
Answer: Jillian Belk

Figure 6: Case study.

swering follow-up questions. Each retrieval operation relies on the generated follow-up questions. When no further follow-up questions are generated, the LLM provides the answer to the original question. We prepend newly retrieved knowledge to the original question following the approach of Yoran et al. (2023). In the context of this paper, SelfAsk shares similarities with ReAct, albeit differing in the location of retrieved knowledge.

- ITER-RETGEN (Shao et al., 2023), a state-of-the-art retrieval-augmented generation method,

Algorithm 1: BlendFilter

Input: An input query $\boldsymbol{q}$, a knowledge base $\mathcal{K}$, a retriever $\mathcal{R}(\cdot)$, and a LLM $\mathcal{M}(\cdot)$.

// query blending

Direct retrieval by feeding $\boldsymbol{q}$ into retriever $\mathcal{R}(\cdot)$;

Generate external knowledge-augmented query according to $\boldsymbol{a}_{ex} = \mathcal{M}(\boldsymbol{a}|\texttt{Prompt}_{\texttt{CoT}}(\boldsymbol{q}, \mathcal{K}_{ex}))$ and $\boldsymbol{q}_{ex} = \boldsymbol{a}_{ex}\|\boldsymbol{q}$;

Generate internal knowledge-augmented query according to $\boldsymbol{a}_{in} = \mathcal{M}(\boldsymbol{a}|\texttt{Prompt}(\boldsymbol{q}))$ and $\boldsymbol{q}_{in} = \boldsymbol{a}_{in}\|\boldsymbol{q}$;

// Knowledge filtering

Retrieve knowledge with different queries based on Eqn. **??**;

Filter retrieved knowledge based on

$$\mathcal{K}_q = \mathcal{R}(\boldsymbol{q}, \mathcal{K}; K),$$
$$\mathcal{K}_{q_{ex}} = \mathcal{R}(\boldsymbol{q}_{ex}, \mathcal{K}; K),$$
$$\mathcal{K}_{q_{in}} = \mathcal{R}(\boldsymbol{q}_{in}, \mathcal{K}; K);$$

Union filtered knowledge according to $\mathcal{K}_r = \mathcal{K}_q^f \bigcup \mathcal{K}_{q_{ex}}^f \bigcup \mathcal{K}_{q_{in}}^f$;

// Answer generation

Generate answer according to $\boldsymbol{a} = \mathcal{M}(\boldsymbol{a}|\texttt{Prompt}_{\texttt{CoT}}(\boldsymbol{q}, \mathcal{K}_r))$.

introduces the iterative augmentation of questions using an external knowledge base and employs knowledge distillation to enhance retriever performance. To ensure a fair comparison, we exclude retrieval training and employ the same retriever as other methods in the case of ITER-RETGEN.

D Dataset Exmples

D.0.1 Implementation Details.

We evaluate our approach with three different LLMs: GPT3.5-turbo-Instruct[2], Vicuna 1.5-13b (Zheng et al., 2023), and Qwen-7b (Bai et al., 2023). GPT3.5-turbo-Instruct is a refined version of InstructGPT (Ouyang et al., 2022), Vicuna 1.5-13b is trained based on Llama 2 (Touvron et al., 2023b) continually, and Qwen-7b is a Transformer-based model trained from scratch. Vicuna 1.5-13b and Qwen-7b are open-source models. We utilize the state-of-the-art efficient retrieval method ColBERT v2 (Santhanam et al., 2022) as the retriever implemented by Khattab et al. (2022, 2023) which applies quantization to accelerate approximate nearest neighbor search. We conduct experiments using Vicuna 1.5-13b with vLLM Kwon et al. (2023) and Qwen-7b with Transformers (Wolf et al., 2020), respectively. The knowledge base we employ is the collection of Wikipedia abstracts dumped in 2017 (Khattab et al., 2023). In all experiments, we utilize a 3-shot in-context learning setting following the approach of Shao et al. (2023). The value of k is set to 5 for all methods. The detailed prompts are provided in the Appendix.

E Case Study

We show an example about how the proposed BlendFilter works in Fig. 6.

[2] https://platform.openai.com/docs/models/gpt-3-5

F Prompt

In this section, We show the prompt we use on three benchmarks for GPT3.5-turbo-Instruct, including prompts for external knowledge augmentation, internal knowledge augmentation, knowledge filtering, and answer generation. Among them, the prompt for external knowledge augmentation is the same for all datasets.

Prompt for External Knowledge Augmentation on HotPotQA

Answer questions following the given format.

Knowledge:{Example_Knowledge}
Question:Are It Might Get Loud and Mr. Big both Canadian documentaries?
Let's think step by step.
Mr. Big is a 2007 documentary which examines the "Mr. Big" undercover methods used by the Royal Canadian Mounted Police. However, It Might Get Loud is a 2008 American documentary film.
So the answer is no.

Knowledge:{Example_Knowledge}
Question:Were László Benedek and Leslie H. Martinson both film directors?
Let's think step by step.
László Benedek was a Hungarian-born film director and Leslie H. Martinson was an American film director.
So the answer is yes.

Knowledge:{Example_Knowledge}
Question:Lucium was confimed to be an impure sample of yttrium by an English chemist who became the president of what?
Let's think step by step.
Lucium was confimed to be an impure sample of yttrium by William Crookes. William Crookes is Sir William Crookes. Sir William Crookes became the president of the Society for Psychical Research.
So the answer is Society for Psychical Research.

Knowledge:{Knowledge}
Question:{question}
Let's think step by step.

Prompt for Internal Knowledge Augmentation

Please write a passage to answer the question.

Question:{question}
Passage:

Prompt for Knowledge Filtering on HotPotQA and 2WikiMultihopQA

What general topic is Question {question} related to?
Answer:The topic is related to

———————————————————— forget your knowledge about {topic}. Please only consider the knowledge below.
knowledge 0 : {Retrieved_knowledge0}
knowledge 1 : {Retrieved_knowledge1}
knowledge 2 : {Retrieved_knowledge2}
knowledge 3 : {Retrieved_knowledge3}
knowledge 4 : {Retrieved_knowledge4}
Please check the relevance between {question} and knowledges 0-4 one by one, remove the irrelevant ones and show me the relevant ones. There may be multiple relevent ones. Please take a deep breath and do it step by step.

———————————————————— Please check the relevance between the given question and knowledges 0-4 one by one based on the given context. ONLY output the relevant knowledge ids (0-4). There may be multiple relevent ones.

Context:{LLM_Last_Generated_Context}

Question:{question}

knowledge 0 : {Retrieved_knowledge0}
knowledge 1 : {Retrieved_knowledge1}
knowledge 2 : {Retrieved_knowledge2}
knowledge 3 : {Retrieved_knowledge3}
knowledge 4 : {Retrieved_knowledge4}

Answer:

Prompt for Answer Generation on HotPotQA

Answer questions following the given format.

Knowledge:{Example_Knowledge}
Question:Are It Might Get Loud and Mr. Big both Canadian documentaries?
Let's think step by step.
Mr. Big is a 2007 documentary which examines the "Mr. Big" undercover methods used by the Royal Canadian Mounted Police. However, It Might Get Loud is a 2008 American documentary film.
So the answer is no.

Knowledge:{Example_Knowledge}
Question:Were László Benedek and Leslie H. Martinson both film directors?
Let's think step by step.
László Benedek was a Hungarian-born film director and Leslie H. Martinson was an American film director.
So the answer is yes.

Knowledge:{Example_Knowledge}
Question:Lucium was confimed to be an impure sample of yttrium by an English chemist who became the president of what?
Let's think step by step.
Lucium was confimed to be an impure sample of yttrium by William Crookes. William Crookes is Sir William Crookes. Sir William Crookes became the president of the Society for Psychical Research.
So the answer is Society for Psychical Research.

Knowledge:{Filtered_Knowledge}
Question:{question}
Let's think step by step.

———————————————————— Answer the following question based on the given context with one or few words.

Context:{LLM_Last_Generated_Context}
Question:{question}
Answer:

Prompt for External Knowledge Augmentation on 2WikiMultihopQA

Answer questions following the given format.

Knowledge:{Example_Knowledge}
Question:Do both films The Falcon (Film) and Valentin The Good have the directors from the same country?
Let's think step by step.
Valentin The Good is directed by Martin Frič. Martin Frič was a Czech film director. The Falcon (Film) is directed by Vatroslav Mimica. Vatroslav Mimica is a Croatian film director. Czech is different from Croatia.
So the answer is no.

Knowledge:{Example_Knowledge}
Question:What nationality is the director of film Wedding Night In Paradise (1950 Film)?
Let's think step by step.
Wedding Night In Paradise (1950 film) is directed by Géza von Bolváry. Géza von Bolváry was a Hungarian actor, screenwriter and film director.
So the answer is Hungarian.

Knowledge:{Example_Knowledge}
Question:Who is Rhescuporis I (Odrysian)'s paternal grandfather?
Let's think step by step.
The father of Rhescuporis I (Odrysian) is Cotys III. The father of Cotys III is Raizdos.
So the answer is Raizdos.

Knowledge:{Knowledge}
Question:{question}
Let's think step by step.

Prompt for Answer Generation on 2WikiMultihopQA

Answer questions following the given format.

Knowledge:{Example_Knowledge}
Question:Do both films The Falcon (Film) and Valentin The Good have the directors from the same country?
Let's think step by step.
Valentin The Good is directed by Martin Frič. Martin Frič was a Czech film director. The Falcon (Film) is directed by Vatroslav Mimica. Vatroslav Mimica is a Croatian film director. Czech is different from Croatia.
So the answer is no.

Knowledge:{Example_Knowledge}
Question:What nationality is the director of film Wedding Night In Paradise (1950 Film)?
Let's think step by step.
Wedding Night In Paradise (1950 film) is directed by Géza von Bolváry. Géza von Bolváry was a Hungarian actor, screenwriter and film director.
So the answer is Hungarian.

Knowledge:{Example_Knowledge}
Question:Who is Rhescuporis I (Odrysian)'s paternal grandfather?
Let's think step by step.
The father of Rhescuporis I (Odrysian) is Cotys III. The father of Cotys III is Raizdos.
So the answer is Raizdos.

Knowledge:{Filtered_Knowledge}
Question:{question}
Let's think step by step.

———————————————————————— Answer the following question based on the given context with one or few words.

Context:{LLM_Last_Generated_Context}
Question:{question}
Answer:

Prompt for External Knowledge Augmentation on StrategyQA

Answer questions following the given format.

Knowledge:{Example_Knowledge}
Question:Do people take laxatives because they enjoy diarrhea?
Let's think step by step.
Laxatives are substances that loosen stools and increase bowel movements. People take laxatives to treat and/or prevent constipation.
So the answer is No.

Knowledge:{Example_Knowledge}
Question:Could Durian cause someone's stomach to feel unwell?
Let's think step by step.
Durian has a pungent odor that many people describe as being similar to feet and onions. Unpleasant smells can make people feel nauseous.
So the answer is Yes.

Knowledge:{Example_Knowledge}
Question:Did the swallow play a role in a famous film about King Arthur?
Let's think step by step.
Monty Python and the Holy Grail was a famous film about King Arthur. In Monty Python and the Holy Grail, swallows are mentioned several times.
So the answer is Yes.

Knowledge:{Knowledge}
Question:{question}
Let's think step by step.

Prompt for Knowledge Filtering on StrategyQA

Please check the relevance between the given question and knowledges 0-4 one by one carefully, remove all the irrelevant ones and only show me the relevant ones. There may be no relevant one.

Question:`{question}`

knowledge 0 : `{Retrieved_knowledge0}`
knowledge 1 : `{Retrieved_knowledge1}`
knowledge 2 : `{Retrieved_knowledge2}`
knowledge 3 : `{Retrieved_knowledge3}`
knowledge 4 : `{Retrieved_knowledge4}`

Please take a deep breath and do it step by step.

Please check the relevance between the given question and knowledges 0-4 one by one based on the given context. ONLY output the relevant knowledge ids (0-4). There may be no relevant one.

Context:`{LLM_Last_Generated_Context}`

Question:`{question}`

knowledge 0 : `{Retrieved_knowledge0}`
knowledge 1 : `{Retrieved_knowledge1}`
knowledge 2 : `{Retrieved_knowledge2}`
knowledge 3 : `{Retrieved_knowledge3}`
knowledge 4 : `{Retrieved_knowledge4}`

Answer:

Prompt for Answer Generation on StrategyQA

Answer questions following the given format.

Knowledge:`{Example_Knowledge}`
Question:Do people take laxatives because they enjoy diarrhea?
Let's think step by step.
Laxatives are substances that loosen stools and increase bowel movements. People take laxatives to treat and/or prevent constipation.
So the answer is No.

Knowledge:`{Example_Knowledge}`
Question:Could Durian cause someone's stomach to feel unwell?
Let's think step by step.
Durian has a pungent odor that many people describe as being similar to feet and onions. Unpleasant smells can make people feel nauseous.
So the answer is Yes.

Knowledge:`{Example_Knowledge}`
Question:Did the swallow play a role in a famous film about King Arthur?
Let's think step by step.
Monty Python and the Holy Grail was a famous film about King Arthur. In Monty Python and the Holy Grail, swallows are mentioned several times.
So the answer is Yes.

Knowledge:`{Filtered_Knowledge}`
Question:`{question}`
Let's think step by step.

Answer the following question based on the given context. The final answer to a question should always be either Yes or No, and NOTHING ELSE.

Context:`{LLM_Last_Generated_Context}`
Question:`{question}`
Answer: