Self-Routing RAG: Binding Selective Retrieval with Knowledge Verbalization

Di Wu*, **Jia-Chen Gu***, **Kai-Wei Chang, Nanyun Peng**
University of California, Los Angeles
{diwu,kwchang,violetpeng}@cs.ucla.edu, gujc@ucla.edu

Abstract

Selective retrieval improves retrieval-augmented generation (RAG) by reducing distractions from low-quality retrievals and improving efficiency. However, existing approaches under-utilize the inherent knowledge of large language models (LLMs), leading to suboptimal retrieval decisions and degraded generation performance. To bridge this gap, we propose Self-Routing RAG (SR-RAG), a novel framework that binds selective retrieval with knowledge verbalization. SR-RAG enables an LLM to dynamically decide between external retrieval and verbalizing its own parametric knowledge. To this end, we design a multi-task objective that jointly optimizes an LLM on knowledge source selection, knowledge verbalization, and response generation. We further introduce dynamic knowledge source inference via nearest neighbor search to improve the accuracy of knowledge source decision under domain shifts. Fine-tuning three LLMs with SR-RAG significantly improves both their response accuracy and inference latency. Compared to the strongest selective retrieval baseline, SR-RAG reduces retrievals by 29% while improving the performance by 5.1%. Data and code will be publicly released at `https://github.com/xiaowu0162/self-routing-rag`.

1 Introduction

Retrieval-augmented generation (RAG) equips large language models (LLMs) with external knowledge sources at inference time, enabling stronger performance on tasks requiring up-to-date or domain-specific information (Khandelwal et al., 2020; Lewis et al., 2020; Borgeaud et al., 2022; Ram et al., 2023; Shi et al., 2024). Recently, selective retrieval—an inference strategy that avoids unnecessary retrieval augmentations—has shown promising results in reducing distractions from low-quality retrievals and improving the efficiency of RAG (He et al., 2021; Mallen et al., 2023; Xu et al., 2024; Wu et al., 2024a).

However, a core question remains overlooked by current selective retrieval research:

Does selective retrieval fully honor the knowledge embedded in the LLM itself?

When retrieval is abstained, existing methods use a standard yet simplistic fallback: letting the LLM directly generate the response (Mallen et al., 2023; Jeong et al., 2024; Asai et al., 2024; Wu et al., 2024a). This design restricts the LLM from explicitly articulating its parametric knowledge before generating a response. We argue that this ability of *knowledge verbalization*, while seemingly subtle, critically impacts the success of selective retrieval. First, verbalizing internal knowledge expands the LLM's capacity to answer without retrieval. Prior work demonstrates the ability of LLMs to directly generate high quality knowledge (Yu et al., 2023) as well as intermediate reasoning paths (Wei et al., 2022; Allen-Zhu & Li, 2025) to benefit the system's performance. This is especially valuable under complex queries, where naive retrieval methods return irrelevant results and compute-intensive retrieval may return noisy contexts. Second, knowledge verbalization enables more accurate selective retrieval

*Equal Contribution

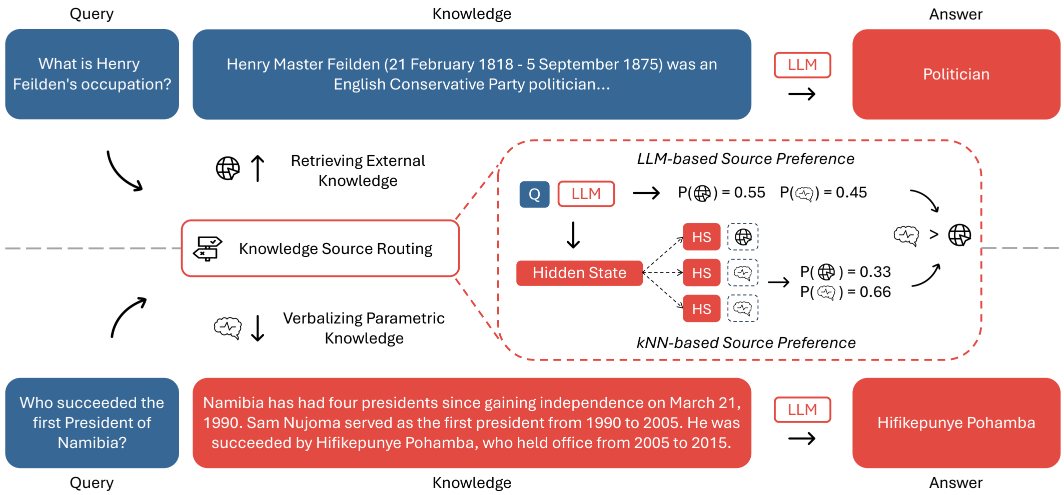

Figure 1: An overview of SR-RAG. Given a user query, the system first selects the most appropriate knowledge source by combining special token prediction with nearest neighbor search. Then, the knowledge is either retrieved from external an external source or self-verbalized by the LLM. Finally, the LLM forms the response based on the query and the knowledge. All the steps are streamlined as a single left-to-right generation pass.

decisions. Existing works train retrieval policies by comparing RAG with direct LLM response (Wang et al., 2023; Wu et al., 2024a) or resorting to likelihood preferences (He et al., 2021; Xu et al., 2024). By contrast, through explicit knowledge elicitation, knowledge verbalization helps characterize the LLM's capabilities more precisely. Therefore, we argue that knowledge verbalization is a core component for selective retrieval to embrace.

We propose Self-Routing RAG (SR-RAG), a selective retrieval framework that tightly integrates knowledge verbalization. By reformulating selective retrieval as a *knowledge source selection* problem, SR-RAG enables an LLM to self-route between retrieving external knowledge and verbalizing its own parametric knowledge, as illustrated in Figure 1. Recognizing the limitations of existing frameworks that fine-tune LLMs for selective retrieval via special token prediction (Asai et al., 2024; Wu et al., 2024a), SR-RAG introduces three key innovations. First, we generate diverse diverse verbalizations of the LLM's internal knowledge to create more accurate training labels for knowledge source selection (§3.3). Second, we introduce a multi-task alignment objective that couples source selection, verbalization, and response generation. We further leverage self-supervised preference alignment over variants of verbalized knowledge to promote high-quality knowledge generation (§3.3). Finally, existing approaches suffer from poor source decision accuracy at inference time due to domain shifts and LLM ability shifts caused by fine-tuning. To bridge this gap, SR-RAG proposes dynamic knowledge source inference via nearest neighbor search, augmenting likelihood-based retrieval decisions with neighboring policy examples in the hidden representation space of the fine-tuned LLM (§3.4). Crucially, SR-RAG's inference remains efficient, requiring only a single left-to-right generation pass.

We evaluate SR-RAG by fine-tuning Llama-2-7B-Chat (Touvron et al., 2023), Phi-3.5-mini-instruct (Abdin et al., 2024), and Qwen2.5-7B-Instruct (Yang et al., 2024) on a mixture of knowledge-intensive tasks to develop the LLMs' ability to select knowledge sources and improve the quality of knowledge verbalization. Extensive experiments across four benchmarks demonstrate that SR-RAG greatly outperforms both always retrieving and the baseline selective retrieval approach. Compared to the latter, SR-RAG achieves 8.5%/2.1%/4.7% higher overall performance while performing 26%/40%/21% fewer retrievals across the three LLMs respectively (§5.2). Further analyses reveal that SR-RAG improves both the accuracy of selective retrieval decisions (§5.3) and the overall inference efficiency (§5.4). Finally, we conduct thorough ablation studies that confirm that the necessity of all three core components to SR-RAG's strong performance (§5.5).

2 Related Work

Selective Retrieval To enhance the efficiency of RAG systems and avoid potentially harmful retrievals, several works have proposed to selectively skip retrieval augmentation, which we call selective retrieval following Xu et al. (2024) and Wu et al. (2024a). One popular approach is to assess whether retrieval augmentation increases the likelihood of the LLM generating the correct answer and distill this observation to a supervised decision model (He et al., 2021; Schick et al., 2023; Xu et al., 2024). Analogously, Wang et al. (2023) and Wu et al. (2024a) directly evaluate the correctness of the answers generated with and without retrieval to create the supervision signal. Recent works have also explored solely examining the question's nature to judge the need for retrieval on question answering tasks (Mallen et al., 2023; Jeong et al., 2024; Asai et al., 2024). To further incorporate LLM's ability and confidence in the retrieval decision, Ding et al. (2024), Yao et al. (2024), and Moskvoretskii et al. (2025b) use the LLM's uncertainty for selective retrieval. By contrast, this paper highlights the benefits of incorporating knowledge verbalization to make precise selective retrieval decisions and boosting the LLM's performance when retrieval is skipped.

Adaptive RAG Inference Strategies This work also relates to the broader field of adaptive RAG, which aims to develop configurable and instance-specific RAG inference strategies. Prior work has introduced *active retrieval*, where the system dynamically refines or re-issues queries if the initially retrieved content is insufficient (Jiang et al., 2023; Su et al., 2024). Other lines of research explore query decomposition and iterative retrieval to better handle complex questions by breaking them down into manageable sub-queries (Shao et al., 2023; Kim et al., 2023; Liu et al., 2024; Lee et al., 2024). Given the retrieval results, Asai et al. (2024) and Yan et al. (2024) propose inference strategies that critique or revise the retrieved knowledge to improve output quality. Parekh et al. (2025) incorporate an initial decision step to adaptively select the most suitable strategy based on the question. While our work focuses on the more targeted problem of selective retrieval, we introduce a novel form of adaptivity: enabling the LLM to *self-route* between external retrieval and internal knowledge verbalization, enhancing both flexibility and efficiency in a principled way.

LLMs as Knowledge Sources A growing body of work has explored using LLMs to generate auxiliary knowledge for downstream tasks. Early studies demonstrated the potential of LLMs to produce relevant background knowledge in zero-shot settings for commonsense reasoning (Shwartz et al., 2020; Liu et al., 2022). More broadly, Yu et al. (2023) introduced a generate-then-read approach, treating LLMs as context generators to directly replace external retrieval for RAG. In addition, LLMs have been shown to generate effective intermediate reasoning steps for complex reasoning questions (Wei et al., 2022; Kojima et al., 2022). Building on these insights, this paper explores fully leveraging LLMs' knowledge generation ability to benefit selective retrieval systems.

3 Approach

In this section, we first reformulate selective retrieval as a knowledge source selection problem. Then, we introduce the details of the proposed Self-Routing RAG framework.

3.1 Problem Formulation

Knowledge Source-Aware Inference Given a user query q, a knowledge source s is invoked to return relevant information as a text sequence $s(q)$, which is then consumed by an LLM reader M to generate a response $M(q, s(q))$. Standard RAG follows this paradigm, where s typically represents a retriever over an external datastore.

Knowledge Source Selection In practice, multiple knowledge sources may be available, denoted as $\mathcal{S} = \{\phi, s_1, ..., s_N\}$, where ϕ is a null knowledge source that returns nothing for any query. A knowledge source selector P chooses the most appropriate source for a given query, i.e., $P(q, \mathcal{S}) \in \mathcal{S}$. Selective retrieval can be viewed as the special case where $\mathcal{S} = \{\phi, s\}$ with the full pipeline expressed as $M(q, P(q, \mathcal{S})(q))$.

3.2 Self-Routing RAG: Overview

We propose Self-Routing RAG (SR-RAG), a generalization of selective retrieval that treats the LLM itself as a first-class knowledge source. As illustrated in Figure 1, given a query q, the LLM self-determines which knowledge source to use: either retrieving from an external source or verbalizing its parametric knowledge. The final response is then generated based on both the query and the knowledge collected from the selected source.

Building upon traditional selective retrieval methods (Asai et al., 2024; Wu et al., 2024a), SR-RAG fine-tunes the LLM to streamline its inference process with special tokens, enabling efficient inference with a single left-to-right generation pass. Three sets of special tokens are introduced:

1. <EOQ> signals the end of the query and prompts the LLM for knowledge source selection[1].
2. A set of special tokens <s>, each representing a knowledge source s. Our main setup uses two sources: external (S_e) and internal (S_i). This formulation can naturally accommodate more than two knowledge sources as well.
3. <EOK> indicates the end of the knowledge and triggers the LLM for answer generation.

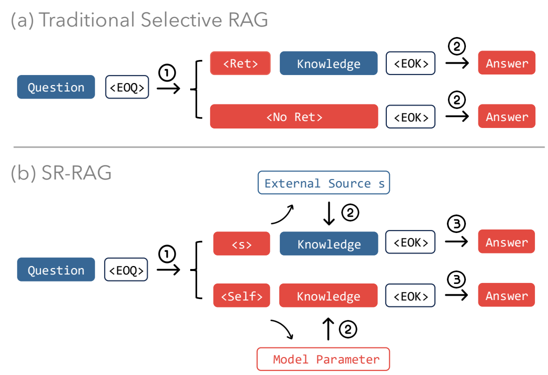

Figure 2: Compared to traditional selective RAG, SR-RAG enables an LLM to self-route between knowledge sources and self-act as a knowledge source. We use blue to represent external information and red to represent the LLM and its self-generated tokens.

As shown in Figure 2, compared to traditional selective retrieval, SR-RAG enables the LLM to actively select the best knowledge source and seamlessly act as a knowledge source itself.

3.3 Self-Routing RAG: Training

To train the backbone LLM for SR-RAG, we propose a pipeline that mines self-supervision from widely available question answering or instruction following data, consisting of pairs of question q and response a. Our pipeline only uses the LLM itself as the internal knowledge source S_i and the external knowledge source S_e, without requiring additional human supervision or synthetic labels from stronger LLMs, demonstrating strong scalability.

Data Construction To enable an LLM to accurately determine whether a question falls within its parametric knowledge and to robustly elicit the knowledge, we argue that thorough and diverse knowledge verbalization is crucial. Following this intuition, SR-RAG collects contexts by exploring each of the available knowledge sources:

- *Parametric Knowledge Verbalization*: We leverage GenRead (Yu et al., 2023) to elicit knowledge from the LLM parametric knowledge source S_i and generate n diverse verbalized contexts, denoted as $c_{i_1}, c_{i_2}, ..., c_{i_n}$.
- *External Knowledge Retrieval*: We retrieve n context chunks from the external knowledge source S_e, denoted as $c_{e_1}, c_{e_2}, ..., c_{e_n}$. In this work, we consider retrieving from Wikipedia with an off-the-shelf dense retriever.

Then, each context c_j is scored by the log-likelihood $l_j = p_M(a|q, c_j)$ of generating the correct answer. Based on the ranking of l_j, we identify the preferred source $s \in \{S_i, S_e\}$ as the one contributing to the majority of the top-n ranked contexts. The resulting $(q, a, s, \{c_j, l_j\})$ tuples are saved for model training. For convenience of later reference, we denote the

[1] This design aligns with Wu et al. (2024a) but diverges from Asai et al. (2024). We argue that this token is necessary for the LLM to allocate probability mass to the tokens for the knowledge sources.

contexts from S_i and S_e that lead to the highest and lowest likelihoods as $c_{i+}, c_{i-}, c_{e+}, c_{e-}$, respectively. We present the formal data creation algorithm in Appendix B.1.

Objective SR-RAG proposes a two-stage multi-task learning framework that jointly optimizes knowledge source selection, knowledge verbalization, and response generation. The first stage performs behavior cloning on three losses:

1. $\mathcal{L}_{src}$: a cross-entropy loss for the preferred knowledge source s following `<EOQ>`:
$$\mathcal{L}_{src} = -\log p_M(\texttt{<s>}|q), \tag{1}$$
where <s> represents the actual token corresponding to the chosen source $s \in \mathcal{S}$.
2. $\mathcal{L}_{verb}$: a cross-entropy loss on the knowledge tokens, only when the LLM itself (S_i) is labeled as the preferred knowledge source:
$$\mathcal{L}_{verb} = \begin{cases} -\log p_M(c_{i+}|q), & \text{if } s = S_i, \\ 0, & \text{if } s = S_e. \end{cases} \tag{2}$$
3. $\mathcal{L}_{ans}$: a cross-entropy loss on generating the answer based on the query and the knowledge from the preferred source:
$$\mathcal{L}_{ans} = \begin{cases} -\log p_M(a|q, c_{i+}), & \text{if } s = S_i, \\ -\log p_M(a|q, c_{e+}), & \text{if } s = S_e. \end{cases} \tag{3}$$

The final loss for the first stage is a simple combination of the three objectives:
$$\mathcal{L}_{stage1} = \mathcal{L}_{src} + \mathcal{L}_{verb} + \mathcal{L}_{ans}. \tag{4}$$

To further boost the LLM's ability to generate useful knowledge, SR-RAG incorporates a second-stage fine-tuning via direct preference optimization (DPO) (Rafailov et al., 2023), pairing self-verbalized knowledge with self-generated preference labels (c_{i+}, c_{i-}).

$$\mathcal{L}_{stage2} = \mathcal{L}_{src} + \mathcal{L}_{verb}^{DPO} + \mathcal{L}_{ans}, \tag{5}$$
$$\mathcal{L}_{verb}^{DPO} = \begin{cases} -\log \sigma\left(\beta \log \frac{p_M(c_{i+}|q)}{p_{ref}(c_{i+}|q)} - \beta \log \frac{p_M(c_{i-}|q)}{p_{ref}(c_{i-}|q)}\right), & \text{if } s = S_i, \\ 0, & \text{if } s = S_e. \end{cases} \tag{6}$$

M and ref are initialized with the LLM fine-tuned on $\mathcal{L}_{stage1}$, and only M is updated.

Overall, this self-supervised pipeline effectively binds knowledge verbalization with the selective retrieval paradigm, enabling the LLM to learn accurate knowledge source preferences through performance-oriented labeling. Analogous to distilling complex "System 2" reasoning into fast "System 1" inference (Yu et al., 2024), the DPO objective leverages the computationally expensive high-quality knowledge to teach the LLM cost-efficient knowledge verbalization at inference time. Finally, SR-RAG naturally extends to more than two knowledge sources, which is useful in practice for distinguishing domain-specific corpora or retrieval methods with varying cost-quality trade-offs.

3.4 Self-Routing RAG: Inference

As shown in Figures 1 and 2, SR-RAG inference unfolds in a single left-to-right pass through three steps: source selection, knowledge collection, and answer generation.

Nearest Neighbor-Enhanced Source Selection A common approach to selecting a knowledge source is to compare the likelihood $p_M(\texttt{<s>}|q)$ for each $s \in \mathcal{S}$, against a fixed threshold (Asai et al., 2024; Wu et al., 2024a). However, this approach fails to account for shifts in the LLM's ability after fine-tuning and lacks fine-grained control over the decision boundary. To make source selection more robust, we propose a dynamic nearest neighbor-based source selection mechanism that builds a *policy datastore* based on *rollout* after the fine-tuning.

Concretely, the fine-tuned LLM is first evaluated on a set of question-answer pairs[2]. For each pair, the probabilities of generating the answer a conditioned on different knowledge sources are compared to decide the preferred one. We then build a policy datastore mapping each query to its preferred source, using the hidden representation at `<EOQ>` as the key. At inference, we retrieve k nearest neighbors from the policy datastore and use their source labels to form a distribution over the sources $p_D(\texttt{<s>}|q)$. Finally, to select the best source $s \in \mathcal{S}$, we apply a threshold on the product:

$$p_M(\texttt{<s>}|q) \times p_D(\texttt{<s>}|q). \tag{7}$$

While tackling the challenges of source selection due to the LLM's ability shift, this approach also exhibits better interpretability. Since the policy datastore consists of explicit source assignments, it can be audited, modified, and expanded by human experts to steer SR-RAG's retrieval behavior in a fine-grained manner.

Subsequently, the knowledge from the corresponding source is gathered. If the LLM prefers S_i, we use greedy decoding to directly verbalize a single knowledge context. After SR-RAG fine-tuning, the generated context serves as a compressed yet high-quality articulation of the parametric knowledge, which would otherwise require compute-expensive knowledge verbalization to elicit. If an external source `<s>` is selected instead, we pause the decoding and retrieve from knowledge source s. With the verbalized or retrieved knowledge appended to the context, the LLM proceeds to generate the final response.

4 Experimental Setup

4.1 Implementation Details of SR-RAG

Data Construction The main experiments are performed on two knowledge sources: the 2018 English Wikipedia (`<Wiki>`) as the external knowledge source, and the LLM itself (`<Self>`) as the internal knowledge source. We use the official Wikipedia embeddings released by Karpukhin et al. (2020) and retrieve in the granularity of 100-word chunks. GenRead (Yu et al., 2023) is used to verbalize diverse knowledge contexts. GenRead clusters zero-shot knowledge verbalizations in the same domain as in-context demonstrations to verbalize diverse knowledge. Each piece of verbalized knowledge is limited to a maximum of 150 tokens. From each knowledge source, we collect $n = 5$ knowledge contexts.

Training We fine-tune on a mixture of six short- and long-form knowledge-intensive datasets: Wizard of Wikipedia (Dinan et al., 2019), Natural Questions (Kwiatkowski et al., 2019), FEVER (Thorne et al., 2018), OpenBookQA (Mihaylov et al., 2018), ARC-Easy (Bhakthavatsalam et al., 2021), and ASQA (Stelmakh et al., 2022). This mixture of 53,042 instances is a subset of the RAG instruction tuning data proven effective in Asai et al. (2024). After running the data construction algorithm discussed in §3.3 on three LLMs, 46.9% of the instances are labeled with `<Self>` and the rest are labeled with `<Wiki>` as the preferred knowledge source. Additional training details are provided in Appendix B.1.

Inference To construct the policy datastore, we use a middle layer in the fine-tuned LLM[3], as middle layers are found effective by previous work on LLM faithfulness (Yin et al., 2024). At test time, the datastore index is cached on GPU and similarity search can be achieved via a single matrix multiplication. We retrieve $k = 30$ nearest supporting examples from the datastore and construct $p_D(\texttt{<s>}|q)$ from the counts of each knowledge source as the preferred source. Then, we impose a model-specific threshold τ on $p_M(\texttt{<Wiki>}|q) \times p_D(\texttt{<Wiki>}|q)$ to decide whether retrieval should be triggered[4]. We find that this threshold generally performs well enough and does not require dataset-specific tuning.

[2]Empirically, we reuse the training set so that no additional data or supervision is required.

[3]Layer 15 for Llama-2-7B-Chat and Phi-3.5–Mini-Instruct and layer 11 for Qwen2.5-7B-Instruct. We provide further visualizations and discussions of layer selection in Appendix C.2.

[4]$\tau = 0.1$ for Llama-2-7B-Chat and $\tau = 0.2$ for the other models.

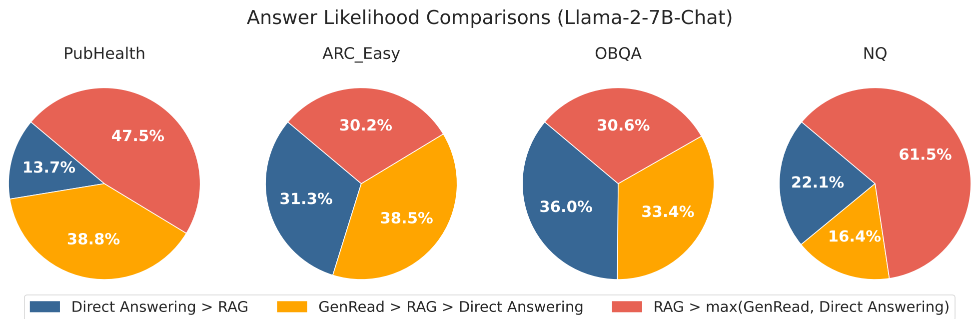

Figure 3: Knowledge verbalization significantly affects the LLM ability boundary. For a large number of instances (16.4% - 38.8%, orange), GenRead reverses the knowledge source preferences: without considering GenRead, RAG dominates over parametric knowledge.

4.2 Evaluation

Datasets and Metrics We evaluate on a diverse set of four knowledge-intensive NLP tasks. **PopQA** (Mallen et al., 2023) is a free-formed long-tail open-domain question answering dataset. Following Asai et al. (2024), we use the subset of 1,399 questions that aims at testing knowledge of long-tail entities. **TriviaQA** (Joshi et al., 2017) is an established open-domain question answering dataset that features relatively complex and diverse questions. We use the same test split and retrieval setup as in Asai et al. (2024). **PubHealth** (Zhang et al., 2023) is a fact-checking dataset focusing on checking claims in the public health domain. Finally, **ARC Challenge** (Bhakthavatsalam et al., 2021) is a multiple-choice question answering dataset featuring grade-school level science questions. Following common practice, we perform lexical postprocessing of the model's output and report accuracy for PubHealth and ARC and substring matching for PopQA and TriviaQA.

Baselines We compare SR-RAG with baselines that cover various training and inference strategies. (1) First, using the LLM before fine-tuning, we compare with either always retrieving or always verbalizing with GenRead. (2) As illustrated in Figure 2, the major baseline we compare with is the state-of-the-art prior *selective retrieval* pipeline, combining the advantage of He et al. (2021), Asai et al. (2024), and Wu et al. (2024a). Specifically, the likelihoods of the LLM generating the answer with and without retrieval are used to create the knowledge source selection label. Then, we fine-tune the LLM for knowledge source selection (among S_e and ϕ) and generate the answer with optional retrieval. At inference, we apply a uniform threshold of 0.2 on the likelihood of the retrieval token for selective retrieval. (3) Always retrieving with the fine-tuned SR-RAG LLM.

5 Results

5.1 Knowledge Verbalization Alters Knowledge Source Preference

To motivate our approach, we first demonstrate that knowledge verbalization has a substantial impact on identifying when an LLM requires retrieval. Figure 3 presents a pilot study on Llama-2-7B-Chat using four datasets from SR-RAG's training mixture. For each instance, we compute the likelihood of the LLM generating the correct answer with no context (blue), with the most helpful GenRead verbalization (c_{i+}, orange), and with the most helpful retrieved passage (c_{e+}, red). Notably, GenRead changes the preferred knowledge source in 16% of Natural Questions examples and over 30% in the other datasets. This suggests that prior selective retrieval methods, which omit verbalization, may significantly underestimate the LLM's capabilities—reinforcing the need to incorporate knowledge verbalization for accurate source preference labeling.

Training	Inference	PopQA		TriviaQA		PubHealth		ARC		Average	
		ACC	%RAG	ACC	%RAG	ACC	%RAG	ACC	%RAG	ACC	%RAG
Llama-2-7B-Chat											
No Fine-tuning	Always RAG	0.529	100%	0.641	100%	0.457	100%	0.546	100%	0.543	100%
	GenRead	0.247	0%	0.616	0%	0.515	0%	0.605	0%	0.496	0%
Selective RAG	Always RAG	0.567	100%	0.640	100%	0.588	100%	0.588	100%	0.596	100%
	Selective RAG	0.565	98%	0.638	100%	0.589	100%	0.594	65%	0.597	86%
SR-RAG	Always RAG	**0.568**	100%	**0.669**	100%	0.689	100%	0.608	100%	0.634	100%
	SR-RAG	0.566	96%	0.664	89%	**0.730**	40%	**0.630**	29%	**0.648**	64%
Phi-3.5-Mini-Instruct											
No Fine-tuning	Always RAG	0.541	100%	0.594	100%	0.549	100%	0.771	100%	0.614	100%
	GenRead	0.331	0%	0.567	0%	0.442	0%	0.840	0%	0.545	0%
Selective RAG	Always RAG	**0.570**	100%	0.645	100%	0.701	100%	0.813	100%	0.682	100%
	Selective RAG	**0.570**	100%	0.638	95%	0.704	91%	0.815	83%	0.682	92%
SR-RAG	Always RAG	0.567	100%	**0.659**	100%	0.689	100%	0.820	100%	0.684	100%
	SR-RAG	0.566	98%	0.657	92%	**0.705**	24%	**0.854**	5%	**0.696**	55%
Qwen2.5-7B-Instruct											
No Fine-tuning	Always RAG	0.563	100%	**0.667**	100%	0.446	100%	**0.916**	100%	0.648	100%
	GenRead	0.334	0%	0.626	0%	0.676	0%	0.875	0%	0.628	0%
Selective RAG	Always RAG	0.555	100%	0.654	100%	0.600	100%	0.827	100%	0.659	100%
	Selective RAG	0.529	88%	0.648	93%	0.608	82%	0.835	78%	0.655	85%
SR-RAG	Always RAG	**0.573**	100%	0.662	100%	0.596	100%	0.821	100%	0.663	100%
	SR-RAG	0.572	99%	0.659	89%	**0.682**	34%	0.830	46%	**0.686**	67%

Table 1: Main evaluation results on four tasks. The best results are boldfaced. Across three LLMs as the backbone models, SR-RAG consistently outperforms selective RAG and always retrieving while consuming a much lower retrieval budget.

5.2 SR-RAG Improves Accuracy and Reduces Retrieval

Table 1 shows the end-to-end generation performance on three LLMs, demonstrating the advantage of SR-RAG over always retrieving, selective retrieval, and other baselines. Relative to always retrieving with the pretrained model, SR-RAG improves accuracy by 19.3% (Llama-2-7B-Chat), 13.4% (Phi-3.5-Mini-Instruct), and 5.9% (Qwen2.5-7B-Instruct). Against the strongest selective retrieval baseline, SR-RAG yields 8.5%/2.1%/4.7% higher accuracy. Interestingly, despite saving 8–15% retrievals, the selective retrieval baseline achieves accuracy nearly identical to always retrieving. This suggests that selective RAG alone cannot reliably identify retrieval-unnecessary queries. In contrast, SR-RAG reduces 20–40% of low-value retrievals while maintaining or improving performance due to accurate knowledge source selection and high-quality verbalization.

Remarkably, with a uniform inference datastore and threshold, SR-RAG dynamically adapts its retrieval behavior based on dataset characteristics. For a dataset that emphasizes long-tail knowledge like PopQA, SR-RAG tends to retrieve external knowledge most of the time. On the other hand, for PubHealth and ARC where the model's knowledge may suffice for a number of questions, SR-RAG relies on internal knowledge more confidently, resulting in much better performance compared to always retrieving. Qualitative examples illustrating SR-RAG's behavior can be found in Figure 9 and Figure 10 in the appendix.

5.3 SR-RAG Makes Accurate Source Selection Decisions

Can SR-RAG accurately choose the right knowledge source? Table 2 compares SR-RAG with Self-RAG (Asai et al., 2024) and an SR-RAG variant without kNN-based policy. We evaluate accuracy using two criteria: (1) abstaining from retrieval when it does not harm performance (top) and (2) retrieving only when it provides strictly better context (bottom). While all methods perform well under the first criterion, SR-RAG achieves the highest overall accuracy. Un-

Method	PopQA	TriviaQA	PubHealth	ARC	Average
Accuracy (Verbalization $\geq$ Retrieval)					
Self-RAG	0.957	0.936	0.867	0.908	0.917
SR-RAG w/o. kNN	**0.959**	0.930	0.869	0.888	0.912
SR-RAG	**0.959**	**0.943**	**0.880**	**0.910**	**0.923**
AUROC (Retrieval $>$ Verbalization)					
Self-RAG	0.489	0.503	0.438	**0.557**	0.497
SR-RAG w/o. kNN	0.490	**0.567**	0.564	0.513	0.534
SR-RAG	**0.577**	0.565	**0.606**	0.533	**0.570**

Table 2: Source selection accuracy measured on Llama-2-7B-Chat. SR-RAG achieves the best averaged performance in both settings.

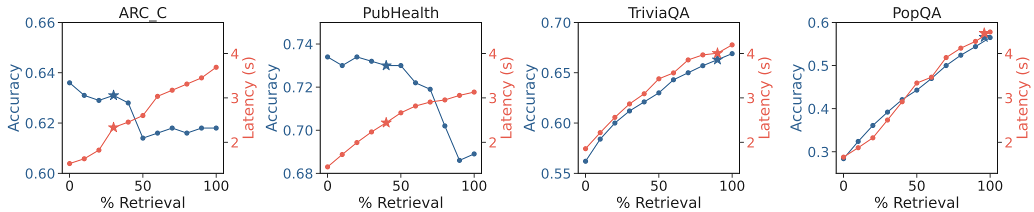

Figure 4: Accuracy and system latency of SR-RAG fine-tuned Llama-2-7B-Chat with different verbalization frequencies. SR-RAG's source selection policy (marked with stars) achieves near-optimal accuracy-efficiency trade-off without dataset-specific thresholds.

Training	Inference	PopQA ACC	PopQA %RAG	TriviaQA ACC	TriviaQA %RAG	PubHealth ACC	PubHealth %RAG	ARC ACC	ARC %RAG	Average ACC	Average %RAG
SR-RAG	SR-RAG	0.566	96%	0.664	89%	**0.730**	40%	**0.630**	29%	**0.648**	64%
SR-RAG	SR-RAG w/o. kNN	0.558	94%	0.658	77%	0.694	72%	0.627	56%	0.634	75%
SR-RAG w/o. kv. label	SR-RAG	**0.568**	100%	0.644	100%	0.598	100%	0.629	84%	0.610	96%
SR-RAG w/o. $\mathcal{L}_{verb}^{DPO}$	SR-RAG	0.564	98%	0.645	100%	0.674	100%	0.581	66%	0.616	86%

Table 3: Ablation studies on SR-RAG. Llama-2-7B-Chat is used as the LLM.

der the second, stricter definition, SR-RAG demonstrates substantially better discrimination, outperforming Self-RAG by 14.7% in average AUROC. Removing kNN-based policy (w/o. kNN) leads to noticeable drops in both accuracy and AUROC, highlighting the benefit of adapting to fine-tuning-induced shifts in LLM ability.

5.4 System Efficiency

We assess SR-RAG's end-to-end latency using Llama-2-7B-Chat under batched inference[5]. Figure 4 shows the trade-off between accuracy and latency under varying retrieval proportions. As expected, latency improves as fewer instances invoke retrieval. Due to dataset-specific difficulty, the optimal retrieval proportion varies per dataset. However, SR-RAG's learned source selection policy achieves a near-optimal accuracy-efficiency trade-off across all datasets, without any dataset-specific threshold tuning. This confirms that SR-RAG not only improves performance but does so efficiently.

5.5 Ablation Study

To understand the contribution of each SR-RAG component, we perform ablations on Llama-2-7B-Chat (Table 3). Removing kNN-based source selection (w/o. kNN) results in lower accuracy and higher retrieval rates, confirming that kNN inference helps the model adapt to ability shifts introduced by fine-tuning. In addition, disabling knowledge verbalization during preference labeling (w/o. kv. label) causes the LLM to over-rely on retrieval and reduces performance, showing that verbalization is crucial for effective labeling. Finally, $\mathcal{L}_{stage2}$ is ablated and the $\mathcal{L}_{stage1}$ loss is kept for stage 2 training. The result shows a significant increase in retrieval proportion as the model's knowledge verbalization ability degrades. These findings highlight that all three components are vital to SR-RAG's success. In Appendix C, we include more detailed analyses on the `<EOQ>` representations, other SR-RAG hyperparameters, as well as qualitative examples.

[5] Detailed latency modeling formulation in Appendix B.2. In our setting, latency is linearly correlated with retrieval frequency under small batch sizes.

6 Conclusion

We present SR-RAG, a novel retrieval-augmented generation (RAG) framework that tightly integrates selective retrieval with knowledge verbalization. By reformulating selective retrieval as a knowledge source selection problem, SR-RAG enables the LLM to not only choose between external and internal knowledge sources but also to serve as a knowledge source itself. During inference, SR-RAG leverages internal hidden states and a nearest-neighbor policy to make accurate, adaptive source selection decisions. Extensive experiments show that SR-RAG significantly improves answer accuracy while reducing retrieval frequency and latency, offering a scalable and reliable path forward for more efficient, knowledge-aware RAG systems.

References

Marah Abdin, Jyoti Aneja, Hany Awadalla, Ahmed Awadallah, Ammar Ahmad Awan, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Jianmin Bao, Harkirat Behl, et al. Phi-3 technical report: A highly capable language model locally on your phone. *ArXiv preprint*, abs/2404.14219, 2024. URL https://arxiv.org/abs/2404.14219.

Zeyuan Allen-Zhu and Yuanzhi Li. Physics of language models: Part 3.2, knowledge manipulation. In *The Thirteenth International Conference on Learning Representations, ICLR 2025*, 2025. URL https://openreview.net/forum?id=oDbiL9CLoS.

Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. Self-rag: Learning to retrieve, generate, and critique through self-reflection. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=hSyW5go0v8.

Sumithra Bhakthavatsalam, Daniel Khashabi, Tushar Khot, Bhavana Dalvi Mishra, Kyle Richardson, Ashish Sabharwal, Carissa Schoenick, Oyvind Tafjord, and Peter Clark. Think you have solved direct-answer question answering? try arc-da, the direct-answer AI2 reasoning challenge. *CoRR*, abs/2102.03315, 2021. URL https://arxiv.org/abs/2102.03315.

Sebastian Borgeaud, Arthur Mensch, Jordan Hoffmann, Trevor Cai, Eliza Rutherford, Katie Millican, George van den Driessche, Jean-Baptiste Lespiau, Bogdan Damoc, Aidan Clark, Diego de Las Casas, Aurelia Guy, Jacob Menick, Roman Ring, Tom Hennigan, Saffron Huang, Loren Maggiore, Chris Jones, Albin Cassirer, Andy Brock, Michela Paganini, Geoffrey Irving, Oriol Vinyals, Simon Osindero, Karen Simonyan, Jack W. Rae, Erich Elsen, and Laurent Sifre. Improving language models by retrieving from trillions of tokens. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvári, Gang Niu, and Sivan Sabato (eds.), *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, volume 162 of *Proceedings of Machine Learning Research*, pp. 2206–2240. PMLR, 2022. URL https://proceedings.mlr.press/v162/borgeaud22a.html.

Emily Dinan, Stephen Roller, Kurt Shuster, Angela Fan, Michael Auli, and Jason Weston. Wizard of wikipedia: Knowledge-powered conversational agents. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019. URL https://openreview.net/forum?id=r1l73iRqKm.

Hanxing Ding, Liang Pang, Zihao Wei, Huawei Shen, and Xueqi Cheng. Retrieve only when it needs: Adaptive retrieval augmentation for hallucination mitigation in large language models. *CoRR*, abs/2402.10612, 2024. doi: 10.48550/ARXIV.2402.10612. URL https://doi.org/10.48550/arXiv.2402.10612.

Junxian He, Graham Neubig, and Taylor Berg-Kirkpatrick. Efficient nearest neighbor language models. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih (eds.), *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 5703–5714, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.emnlp-main.461. URL https://aclanthology.org/2021.emnlp-main.461/.

Soyeong Jeong, Jinheon Baek, Sukmin Cho, Sung Ju Hwang, and Jong Park. Adaptive-RAG: Learning to adapt retrieval-augmented large language models through question complexity. In Kevin Duh, Helena Gomez, and Steven Bethard (eds.), *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 7036–7050, Mexico City, Mexico, June 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.naacl-long.389. URL https://aclanthology.org/2024.naacl-long.389/.

Zhengbao Jiang, Frank Xu, Luyu Gao, Zhiqing Sun, Qian Liu, Jane Dwivedi-Yu, Yiming Yang, Jamie Callan, and Graham Neubig. Active retrieval augmented generation. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 7969–7992, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.495. URL https://aclanthology.org/2023.emnlp-main.495/.

Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In Regina Barzilay and Min-Yen Kan (eds.), *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1601–1611, Vancouver, Canada, July 2017. Association for Computational Linguistics. doi: 10.18653/v1/P17-1147. URL https://aclanthology.org/P17-1147/.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. Dense passage retrieval for open-domain question answering. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu (eds.), *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 6769–6781, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.550. URL https://aclanthology.org/2020.emnlp-main.550/.

Urvashi Khandelwal, Omer Levy, Dan Jurafsky, Luke Zettlemoyer, and Mike Lewis. Generalization through memorization: Nearest neighbor language models. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020. URL https://openreview.net/forum?id=HklBjCEKvH.

Gangwoo Kim, Sungdong Kim, Byeongguk Jeon, Joonsuk Park, and Jaewoo Kang. Tree of clarifications: Answering ambiguous questions with retrieval-augmented large language models. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 996–1009, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.63. URL https://aclanthology.org/2023.emnlp-main.63/.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners. In Sanmi Koyejo, S. Mohamed, A. Agarwal, Danielle Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022. URL http://papers.nips.cc/paper_files/paper/2022/hash/8bb0d291acd4acf06ef112099c16f326-Abstract-Conference.html.

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. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:452–466, 2019. doi: 10.1162/tacl_a_00276. URL https://aclanthology.org/Q19-1026/.

Myeonghwa Lee, Seonho An, and Min-Soo Kim. PlanRAG: A plan-then-retrieval augmented generation for generative large language models as decision makers. In Kevin Duh, Helena Gomez, and Steven Bethard (eds.), *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language*

Technologies (Volume 1: Long Papers), pp. 6537–6555, Mexico City, Mexico, June 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.naacl-long.364. URL https://aclanthology.org/2024.naacl-long.364/.

Patrick S. H. Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive NLP tasks. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/6b493230205f780e1bc26945df7481e5-Abstract.html.

Jiacheng Liu, Alisa Liu, Ximing Lu, Sean Welleck, Peter West, Ronan Le Bras, Yejin Choi, and Hannaneh Hajishirzi. Generated knowledge prompting for commonsense reasoning. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio (eds.), *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3154–3169, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-long.225. URL https://aclanthology.org/2022.acl-long.225/.

Yanming Liu, Xinyue Peng, Xuhong Zhang, Weihao Liu, Jianwei Yin, Jiannan Cao, and Tianyu Du. RA-ISF: Learning to answer and understand from retrieval augmentation via iterative self-feedback. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 4730–4749, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-acl.281. URL https://aclanthology.org/2024.findings-acl.281/.

Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, and Hannaneh Hajishirzi. When not to trust language models: Investigating effectiveness of parametric and non-parametric memories. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki (eds.), *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 9802–9822, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.546. URL https://aclanthology.org/2023.acl-long.546/.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. Can a suit of armor conduct electricity? a new dataset for open book question answering. In Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii (eds.), *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 2381–2391, Brussels, Belgium, October-November 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1260. URL https://aclanthology.org/D18-1260/.

Viktor Moskvoretskii, Maria Lysyuk, Mikhail Salnikov, Nikolay Ivanov, Sergey Pletenev, Daria Galimzianova, Nikita Krayko, Vasily Konovalov, Irina Nikishina, and Alexander Panchenko. Adaptive retrieval without self-knowledge? bringing uncertainty back home. *CoRR*, abs/2501.12835, 2025a. doi: 10.48550/ARXIV.2501.12835. URL https://doi.org/10.48550/arXiv.2501.12835.

Viktor Moskvoretskii, Maria Lysyuk, Mikhail Salnikov, Nikolay Ivanov, Sergey Pletenev, Daria Galimzianova, Nikita Krayko, Vasily Konovalov, Irina Nikishina, and Alexander Panchenko. Adaptive retrieval without self-knowledge? bringing uncertainty back home. *CoRR*, abs/2501.12835, 2025b. doi: 10.48550/ARXIV.2501.12835. URL https://doi.org/10.48550/arXiv.2501.12835.

Tanmay Parekh, Pradyot Prakash, Alexander Radovic, Akshay Shekher, and Denis Savenkov. Dynamic strategy planning for efficient question answering with large language models. In *Findings of the Association for Computational Linguistics: NAACL 2025*, 2025. URL https://arxiv.org/abs/2410.23511.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D. Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine (eds.), *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. URL http://papers.nips.cc/paper_files/paper/2023/hash/a85b405ed65c6477a4fe8302b5e06ce7-Abstract-Conference.html.

Ori Ram, Yoav Levine, Itay Dalmedigos, Dor Muhlgay, Amnon Shashua, Kevin Leyton-Brown, and Yoav Shoham. In-context retrieval-augmented language models. *Transactions of the Association for Computational Linguistics*, 11:1316–1331, 2023. doi: 10.1162/tacl_a_00605. URL https://aclanthology.org/2023.tacl-1.75/.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Eric Hambro, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. Toolformer: Language models can teach themselves to use tools. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine (eds.), *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. URL http://papers.nips.cc/paper_files/paper/2023/hash/d842425e4bf79ba039352da0f658a906-Abstract-Conference.html.

Zhihong Shao, Yeyun Gong, Yelong Shen, Minlie Huang, Nan Duan, and Weizhu Chen. Enhancing retrieval-augmented large language models with iterative retrieval-generation synergy. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 9248–9274, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-emnlp.620. URL https://aclanthology.org/2023.findings-emnlp.620/.

Weijia Shi, Sewon Min, Michihiro Yasunaga, Minjoon Seo, Richard James, Mike Lewis, Luke Zettlemoyer, and Wen-tau Yih. REPLUG: Retrieval-augmented black-box language models. In Kevin Duh, Helena Gomez, and Steven Bethard (eds.), *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 8371–8384, Mexico City, Mexico, June 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.naacl-long.463. URL https://aclanthology.org/2024.naacl-long.463/.

Vered Shwartz, Peter West, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. Unsupervised commonsense question answering with self-talk. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu (eds.), *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 4615–4629, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.373. URL https://aclanthology.org/2020.emnlp-main.373/.

Ivan Stelmakh, Yi Luan, Bhuwan Dhingra, and Ming-Wei Chang. ASQA: Factoid questions meet long-form answers. In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang (eds.), *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 8273–8288, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.emnlp-main.566. URL https://aclanthology.org/2022.emnlp-main.566/.

Weihang Su, Yichen Tang, Qingyao Ai, Zhijing Wu, and Yiqun Liu. DRAGIN: Dynamic retrieval augmented generation based on the real-time information needs of large language models. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 12991–13013, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.702. URL https://aclanthology.org/2024.acl-long.702/.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. FEVER: a large-scale dataset for fact extraction and VERification. In Marilyn Walker, Heng Ji, and Amanda Stent (eds.), *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pp. 809–819, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-1074. URL https://aclanthology.org/N18-1074/.

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

Yile Wang, Peng Li, Maosong Sun, and Yang Liu. Self-knowledge guided retrieval augmentation for large language models. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 10303–10315, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-emnlp.691. URL https://aclanthology.org/2023.findings-emnlp.691/.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In Sanmi Koyejo, S. Mohamed, A. Agarwal, Danielle Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022. URL http://papers.nips.cc/paper_files/paper/2022/hash/9d5609613524ecf4f15af0f7b31abca4-Abstract-Conference.html.

Di Wu, Wasi Uddin Ahmad, Dejiao Zhang, Murali Krishna Ramanathan, and Xiaofei Ma. Repoformer: Selective retrieval for repository-level code completion. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net, 2024a. URL https://openreview.net/forum?id=moyG54Okrj.

Di Wu, Jia-Chen Gu, Fan Yin, Nanyun Peng, and Kai-Wei Chang. Synchronous faithfulness monitoring for trustworthy retrieval-augmented generation. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 9390–9406, Miami, Florida, USA, November 2024b. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.527. URL https://aclanthology.org/2024.emnlp-main.527/.

Fangyuan Xu, Weijia Shi, and Eunsol Choi. RECOMP: improving retrieval-augmented lms with context compression and selective augmentation. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=mlJLVigNHp.

Shi-Qi Yan, Jia-Chen Gu, Yun Zhu, and Zhen-Hua Ling. Corrective retrieval augmented generation. *CoRR*, abs/2401.15884, 2024. doi: 10.48550/ARXIV.2401.15884. URL https://doi.org/10.48550/arXiv.2401.15884.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, et al. Qwen2. 5 technical report. *ArXiv preprint*, abs/2412.15115, 2024. URL https://arxiv.org/abs/2412.15115.

Zijun Yao, Weijian Qi, Liangming Pan, Shulin Cao, Linmei Hu, Weichuan Liu, Lei Hou, and Juanzi Li. Seakr: Self-aware knowledge retrieval for adaptive retrieval augmented generation. *CoRR*, abs/2406.19215, 2024. doi: 10.48550/ARXIV.2406.19215. URL https://doi.org/10.48550/arXiv.2406.19215.

Fan Yin, Jayanth Srinivasa, and Kai-Wei Chang. Characterizing truthfulness in large language model generations with local intrinsic dimension. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=7DbIyQlfaO.

Ping Yu, Jing Xu, Jason Weston, and Ilia Kulikov. Distilling system 2 into system 1. *CoRR*, abs/2407.06023, 2024. doi: 10.48550/ARXIV.2407.06023. URL https://doi.org/10.48550/arXiv.2407.06023.

Wenhao Yu, Dan Iter, Shuohang Wang, Yichong Xu, Mingxuan Ju, Soumya Sanyal, Chenguang Zhu, Michael Zeng, and Meng Jiang. Generate rather than retrieve: Large language models are strong context generators. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023. URL https://openreview.net/forum?id=fB0hRu9GZUS.

Tianhua Zhang, Hongyin Luo, Yung-Sung Chuang, Wei Fang, Luc Gaitskell, Thomas Hartvigsen, Xixin Wu, Danny Fox, Helen Meng, and James R. Glass. Interpretable unified language checking. *CoRR*, abs/2304.03728, 2023. doi: 10.48550/ARXIV.2304.03728. URL https://doi.org/10.48550/arXiv.2304.03728.

A List of Notations

In Table 4, we present the major notations and parameters used throughout the paper.

Notation	Description
q	User query input to the system.
a	The expected answer.
M	The LLM.
$\mathcal{S}$	Set of all knowledge sources.
S_e	The external knowledge source.
S_i	The internal knowledge source (parametric knowledge).
c_{i+}	Most helpful verbalized knowledge context from S_i.
c_{i-}	Least helpful verbalized knowledge context from S_i.
c_{e+}	Most helpful retrieved knowledge context from S_e.
c_{e-}	Least helpful retrieved knowledge context from S_e.
`<EOQ>`	End-of-query special token.
`<EOK>`	End-of-knowledge special token.
`<s>`	Special token representing knowledge source $s \in \mathcal{S}$.
`<Wiki>`	Special token representing Wikipedia.
`<Self>`	Special token representing the LLM itself as knowledge source.
k	number of neighbors retrieved for source policy inference

Table 4: A summary of the key symbols and parameters used in the paper.

B SR-RAG: Further Details

B.1 Training Details

Dataset Construction Algorithm Algorithm 1 presents the full algorithm for constructing training data and labeling knowledge source preferences. GenRead is executed independently on each training data subset. We adopt instance-level notation for clarity. The pipeline naturally scales to additional knowledge sources by applying knowledge collection and likelihood evaluation in parallel across sources.

Algorithm 1 **SR-RAG Training Data Construction**

Require: LLM M, External Retriever $\mathcal{R}$, Dataset $\mathcal{D}$, Number of contexts n
for $(q, a) \in \mathcal{D}$ **do**
  // Retrieving External Knowledge
  $\mathcal{C}_{S_e} \leftarrow \mathcal{R}(q, n)$
  // Knowledge Verbalization
  $\mathcal{C}_{S_i} \leftarrow M.\text{GenRead}(q, n)$
  // Compute Likelihoods
  for $c \in \mathcal{C}_{S_e} \cup \mathcal{C}_{S_i}$ **do**
    $l_c \leftarrow p_M(a|q, c)$
  end for
  $s \leftarrow \arg\max_{s \in \{S_e, S_i\}} \sum_{c \in \mathcal{C}_s} l_c$
  Store $(q, a, s, \{c, l_c\})$
end for
return Processed dataset with labeled knowledge sources

GenRead Prompt We implement GenRead following the setup in Yu et al. (2023). In the second verbalization round, five clusters with five in-context examples each are used. All datasets except ASQA follow the general prompt shown in Figure 5. For ASQA, we include an additional instruction to handle ambiguity: "`If the question is ambiguous, generate multiple documents for each possibility.`"

```
Generate a background document from Wikipedia to help answer the following question.
Directly start with document content and do not generate URL.

Question: {question}

Background document:
```

Figure 5: Prompt used for knowledge verbalization data collection via GenRead.

Training data Table 5 summarizes the training and validation splits, including the proportion of examples where verbalization is preferred over retrieval.

Dataset	Train	Validation	Total	%Verbalization		
				Llama	Phi	Qwen
ARC_Easy	2037	107	2144	61%	84%	66%
NQ	14753	776	15529	28%	33%	41%
OBQA	4462	234	4696	61%	77%	61%
FEVER	9467	498	9965	52%	58%	68%
WoW	16493	868	17361	13%	55%	32%
ASQA	3700	194	3894	13%	25%	16%

Table 5: Statistics of the training and validation data with verbalization percentages. Llama = Llama-2-7B-Chat, Phi = Phi-3.5-Mini-Instruct, and Qwen = Qwen2.5-7B-Instruct.

Training process To fully leverage the backbone LLM's ability to follow natural language instructions, both SR-RAG fine-tuning and inference use the following prompt that interleaves special tokens with natural language:

```
Question: {question} Background knowledge: <EOQ>
     <s> {knowledge} <EOK> Answer: {answer}
```

During training, the loss is computed on the knowledge part only if <s> is <Self>. If <s> is <Wiki>, we augment c_{e+} by randomly appending $\max(p-1, 0)$ retrieved contexts, where p is sampled from a Poisson distribution with $\lambda = 2$. This data augmentation improves the LLM's robustness to different retrieval strategies and various levels of retrieval quality. For stage 1 training, we use batch size 64, learning rate 1e-5, and fine-tune for 1 epoch. For stage 2 training, we use batch size 64, learning rate 5e-7, β = 0.3 for DPO, and train for another epoch. All the experiments are performed on a machine with eight A800 (80GB) GPUs and a machine with eight A6000 GPUs. On eight A800 (80GB) GPUs, the two-staged training takes approximately 10 hours for a 7B-sized model.

B.2 Latency Formulation

To evaluate the inference efficiency of SR-RAG, we measure the latency in a realistic batched inference setup, where the system handles a batch of B = 10 queries and returns the results for all of them together. For the latency experiments presented in the paper, we decompose the system latency as follows:

- **Source Selection Time** (T_d): The time taken by the knowledge source selector to determine whether to retrieve from external sources or rely on parametric knowledge.
- **Retrieval Latency** (T_{rs}): The time taken to fetch external knowledge from the database if the model chooses to retrieval from an external knowledge source s. In our batched setting, we calculate T_{rs} by performing a batched retrieval for all the B instances that require retrieval and report the per-item latency.
- **Verbalization Latency** (T_v): The time taken for the LLM to verbalize parametric knowledge if itself is selected as the knowledge source.

- **Generation Latency** (T_g): The time required for the LLM to generate the response, conditioned on either retrieved or verbalized knowledge.

Thus, the total per-item latency T_{total} is given by:

$$T_{\text{total}} = \begin{cases} T_d + T_v + T_g, & \text{if verbalize,} \\ T_d + T_{rs} + T_g, & \text{if retrieve from knowledge source } s. \end{cases}$$

We have the following remarks:

- This formulation assumes that both the retrieval index and the source selection datastore are pre-constructed. This assumption is reasonable as these indices are only constructed once and then constantly reused.
- We choose the batched setup due to the complexity of the retrieval system. For instance, in our implementation of Wikipedia search, it takes around five to ten seconds per instance to encode the query and retrieve the most relevant context chunks. In the batched setting with queries executed in parallel, the throughput better measures the advantage of SR-RAG.
- As the nearest neighbor only involves one matrix product for similarity calculation and one top-k operation, T_d is generally very small. In fact, we find T_d (0.01s) $<<$ T_g (0.1s) $<$ T_v (1s). On the other hand, T_{rs} is the major bottleneck of the pipeline. As a result, in an online setting, the system's efficiency gain directly converges to the percentage of retrievals it is able to avoid.

C Further Analyses

C.1 Self-Routing RAG: Hyperparameters

In this section, we analyze other hyperparameters in SR-RAG.

Source Labeling Heuristics In SR-RAG, the default design for source preference labeling is collecting the knowledge from each source and selecting the top-ranked source in top 50% knowledge contexts in terms of contribution to the likelihood of the answer. In this section, we compare this strategy to a number of alternative heuristics:

- **Best Single Likelihood:** Selecting the source that produces the knowledge leading to the highest answer likelihood.
- **Best Average Likelihood:** Selecting the source that leads to the highest answer likelihood, averaged over all knowledge contexts.
- **Best All Rank:** Selecting top-ranked knowledge source using all the knowledge context instead of top-50%.

In Table 6, we present the F1 score and AUROC using Llama-2-7B-Chat as the LLM. Correctness is defined as preferring self-verbalized knowledge when it is better than or equal to performing retrieval in terms of the downstream question answering performance. Compared to three baseline methods, the heuristics SR-RAG uses achieves the best F1 and AUROC over the short-form and closed-set subset of the training set. Building upon this paper's results, future work can further study leveraging more advanced uncertainty quantification methods (Wu et al., 2024b; Moskvoretskii et al., 2025a) to build more accurate source selection labels.

Datastore Size To study the influence of the kNN policy datastore size on the performance of SR-RAG, we randomly sample a subset of the training set (50k samples in total) to construct the datastore. As shown in Figure 6 (blue), reducing the kNN policy datastore to 25k (half) has minimal impact. Even with just 1k examples, performance drops only slightly, suggesting potential for lightweight and memory-efficient deployments.

Method	ARC_Easy		NQ		OBQA		Fever		Average	
	F1	AUROC	F1	AUROC	F1	AUROC	F1	AUROC	F1	AUROC
Best Single Likelihood	0.640	0.572	0.320	0.551	0.660	0.606	0.560	0.650	0.545	0.585
Best Average Likelihood	0.730	**0.692**	0.420	0.626	**0.720**	**0.643**	0.630	0.703	0.630	0.666
Best All Rank	0.700	0.691	**0.470**	0.632	0.700	0.621	0.630	0.674	0.625	0.655
SR-RAG	**0.740**	0.691	0.440	**0.635**	**0.720**	0.635	**0.670**	**0.711**	**0.643**	**0.668**

Table 6: F1 and AUROC scores for different source labeling methods across datasets. Llama-2-7B-Chat is used as the LLM. The best score per column is boldfaced.

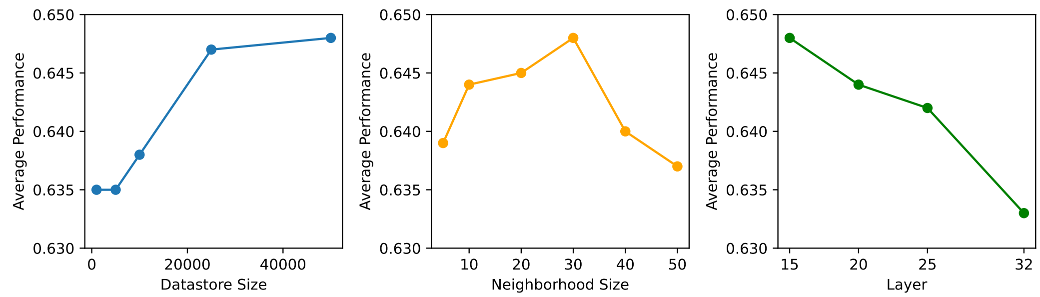

Figure 6: Hyperparameter settings for the kNN policy datastore with Llama-2-7B-Chat.

Number of Neighbors The number of neighbors retrieved from the datastore determines the localness of the focus in the hidden representation space. While task-specific neighborhood size can be tuned, SR-RAG generally works well with 10 to 30 neighboring data points (Figure 6 middle), indicating its robustness. However, using a too small (e.g., 5) or a too large (e.g., more than 50) neighborhood harms the performance of SR-RAG.

Layer for Hidden State We further study the impact of layer selection on SR-RAG's performance in Figure 6 part 3 (green). Overall, we observe that using a middle layer's hidden states has more superior performance over using the last layer's. In the next section, we visualize the hidden state space across LLMs and show that middle layers tend to learn better representations for the model uncertainty as well as task types.

C.2 Hidden State Space of the Source Reflection Token

Does SR-RAG training allow the model to condense their knowledge of task, model uncertainty, and source characteristics into the representation of `<EOQ>`?

Uncertainty Encoding in `<EOQ>` In Figure 7, we visualize `<EOQ>` hidden states on PopQA using t-SNE. Clear separation emerges between instances where self-verbalized knowledge is helpful (green) and unhelpful (red). Middle layers show stronger clustering than final layers, supporting the use of these representations for kNN-based source selection.

Implicit Task Clustering We further randomly sample 500 data points from the SR-RAG train set and visualize their hidden states. Figure 8 shows that SR-RAG also learns to cluster instances by task type (e.g., fact-checking, closed QA, long-form generation). This suggests that source reflection tokens carry semantically meaningful and task-sensitive representations on top of uncertainty information. These behaviors serve as the foundation of our nearest neighbor-based source selection approach.

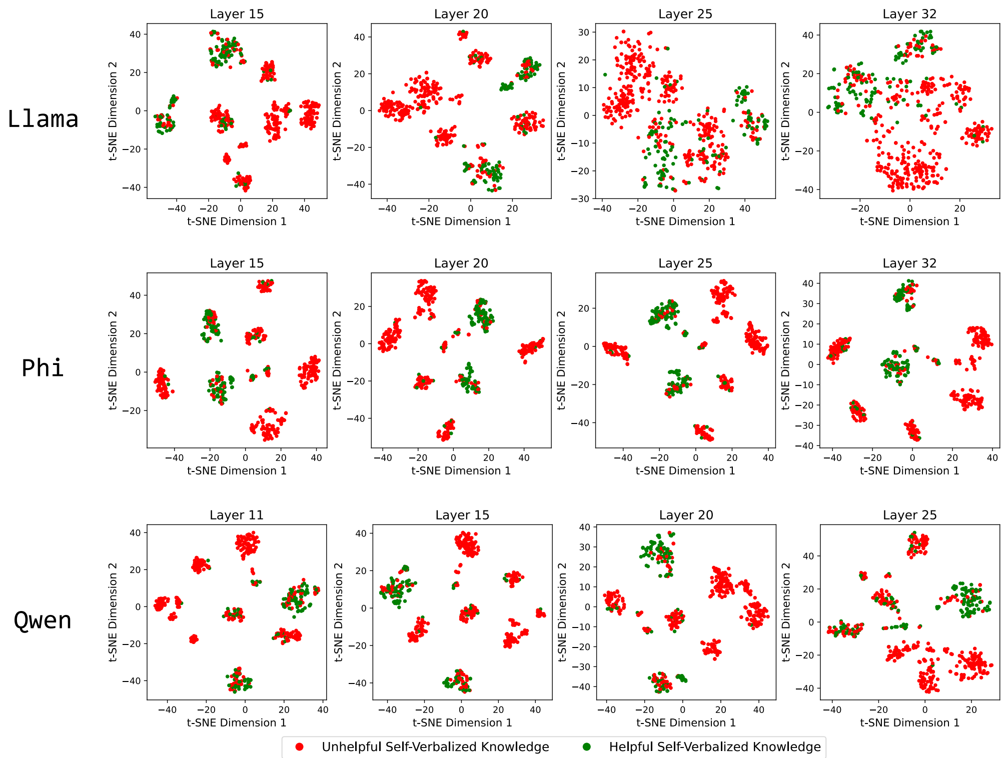

Figure 7: Hidden states of <EOQ> across three LLMs on PopQA, visualized using t-SNE. Llama = Llama-2-7B-Chat, Phi = Phi-3.5-Mini-Instruct, and Qwen = Qwen2.5-7B-Instruct.

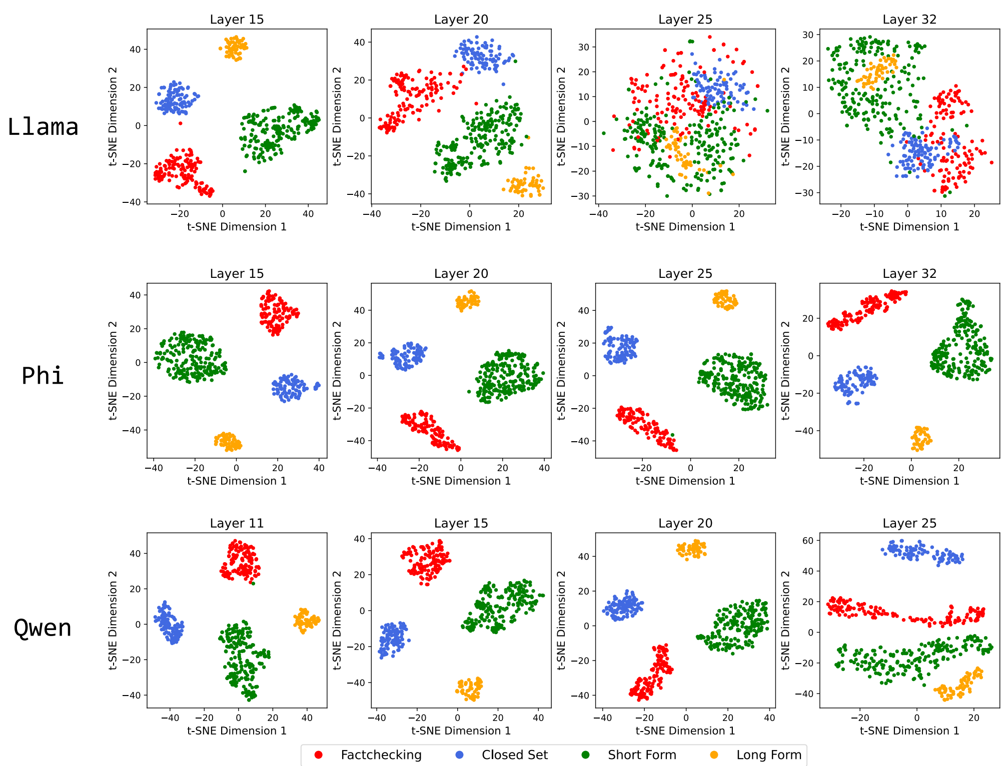

Figure 8: Hidden states of <EOQ> across three LLMs on a random sample of 500 data points in the SR-RAG training set, visualized using t-SNE. Llama = Llama-2-7B-Chat, Phi = Phi-3.5-Mini-Instruct, and Qwen = Qwen2.5-7B-Instruct.

C.3 Can SR-RAG learn more from self-generated knowledge?

In SR-RAG training, self-verbalized knowledge contexts from the LLM are utilized to fine-tune the model via behavior cloning (stage 1) or preference alignment (stage 2). Alternatively, is it possible that the model can learn to verbalize even higher-quality knowledge from off-policy samples generated by a stronger LLM? We investigate this hypothesis by running GenRead-based knowledge verbalization on Llama-3.3-70B-Instruct and directly use the knowledge to prepare SR-RAG training data with Llama-2-7B-Chat. Table 7 compares the performance of fine-tuning Llama-2-7B-Chat with this data versus using self-generated knowledge for SR-RAG. Interestingly, using self-generated positive and negative knowledge pairs is more beneficial for unlocking the model's ability to generate useful knowledge, outperforming the alternative by 4.9% higher answer accuracy and 5.0% higher win rate over retrieval. It is also possible that the ability of the teacher model needs to be close to the fine-tuned model for its knowledge to be useful. We leave this investigation to future work.

Source for c_{i+} and c_{i-}	**PopQA**	**TriviaQA**	**PubHealth**	**ARC**	**Average**
%Verb $\geq$ RAG ($\uparrow$)					
Self-Generated Knowledge	**33.0%**	**56.6%**	**73.4%**	**63.5%**	**56.6%**
Distilled Knowledge	28.2%	55.3%	69.7%	62.3%	53.9%
Verbalization Performance ($\uparrow$)					
Self-Generated Knowledge	**0.302**	**0.572**	**0.734**	**0.634**	**0.561**
Distilled Knowledge	0.282	0.553	0.697	0.606	0.535

Table 7: Comparison between training with self-generated knowledge versus knowledge distillation for the verbalization branch in SR-RAG. Llama-2-7B-Chat is fine-tuned either with self-verbalized knowledge or the knowledge verbalized from Llama-3.3-70B-Instruct.

C.4 Qualitative Study

In Figure 9 and Figure 10, we show two qualitative examples from TriviaQA with the fine-tuned Llama-2-7B-Chat as the backbone LLM. For both examples, we show the LLM's prediction conditioned on both retrieved knowledge and the verbalized knowledge. While the verbalized knowledge is more concise than retrieved paragraphs, it better identifies the correct answers for both questions. This is because the core information (i.e., "first name", "island") only takes up a small portion of the query. As a result, dense retrieval is noise-prone, bringing in only distantly relevant information. By contrast, the LLM as a knowledge generator dynamically adjusts its focus, guiding the knowledge generation to include the core information. As a result, the LLM can correctly answer both questions with self-verbalized knowledge but provides incorrect answers with the low-quality retrieved knowledge. In the source selection process, the LLM directly predicts the source preference $p_M(<\text{s}>|q)$. In addition, questions from the train set sharing the same topic and difficulty level are retrieved to form the source preference distribution $p_D(<\text{s}>|q)$. In the first example (Figure 9), both distributions prefer the LLM itself as the knowledge source. By contrast, in the second example (Figure 10), $p_M(<\text{s}>|q)$ has a high uncertainty and $p_D(<\text{s}>|q)$ corrects the source distribution to select the LLM itself as the knowledge source.

Question: In the book David Copperfield what was Mr. Micawber's first name?
Answer: Wilkins

Retrieved Knowledge: [1] David Copperfield (1999 film) a visit from Mr. Peggotty, back from Australia. He brings news that Emily has made a full recovery and that Mr. Micawber has established himself as a successful magistrate and bank manager. The story closes with the birth of David and Agnes's third child - a girl. Betsey Trotwood's wish finally comes true after nearly 30 years, as David decides that the baby will be christened Betsey Trotwood Copperfield, in honour of her godmother. [2]... (following knowledge chunks omitted)
Prediction (with retrieved knowledge): John Micawber, Sr., is a fictional character in Charles Dickens's 1850 novel David Copperfield. He is the father of David Copperfield's godfather, Mr. Micawber, Jr., and is portrayed as a kind and gentle man who is often in financial difficulties. Mr. Micawber's first name is Charles, not John, as is often mistaken.

Verbalized Knowledge: David Copperfield is a novel by Charles Dickens, published in 1849-50. The novel follows the life of David Copperfield, a young man who grows up in Victorian England. Mr. Micawber is a character in the novel, a kind and well-meaning man who is a friend of David's father. Mr. Micawber's first name is Wilkins.
Prediction (with verbalized knowledge): Wilkins Micawber. Mr. Micawber is a character in Charles Dickens 's novel David Copperfield. He is a kind and well - meaning friend of David 's father, Mr. Copperfield. Mr. Micawber is known for his optimistic outlook on life and his tendency to get into financial difficulties.

Neighbor Questions:
- What was Mr. Smith's first name in Mr. Smith goes to Washington? (<Self> > <Wiki>)
- Who was the original singer of Knocking on Heaven's Door? (<Self> > <Wiki>)
- When did sir Arthur Conan Doyle wrote his first book? (<Wiki> > <Self>)
... (27 questions omitted)
$p_D(\text{<s>}|q)$: P(<Wiki>) = 9/30 = 0.3, P(<Self>) = 21/30 = 0.7.
$p_M(\text{<s>}|q)$: P(<Wiki>) = 0.146, P(<Self>) = 0.854.
Final Source Decision: <Self>

Figure 9: Example of SR-RAG inference where $p_D(\text{<s>}|q)$ agrees with $p_M(\text{<s>}|q)$. As the model itself is selected as the knowledge source, the retrieval will be abstained. The retrieved knowledge and the corresponding prediction are shown only for illustration.

Question: From 1791 to 1803, Toussaint L'Ouverture led a revolt against France on which island?
Answer: Haiti

Retrieved Knowledge: [1] Saint-Domingue expedition slave revolt that led to the abolition of slavery in 1793 by the civil commissioners Sonthonax and Polverel, in a decision endorsed and spread to all the French colonies by the National Convention 6 months later. Toussaint Louverture, a black former slave who had been made Governor by France, re-established peace, fought off Spanish and British attempts to capture the island, and reestablished prosperity by daring measures. [2]... (following knowledge chunks omitted)
Prediction (with retrieved knowledge): Saint-Pierre et Miquelon (Saint-Pierre and Miquelon), a French overseas collectivity in the North American Atlantic Ocean.

Verbalized Knowledge: Toussaint L'Ouverture was a Haitian leader who led a successful revolt against French colonial rule in Haiti from 1791 to 1803. L'Ouverture was a former slave who became a prominent figure in the Haitian Revolution, which was a series of uprisings and battles fought between Haitian slaves and their French colonial oppressors. After years of fighting, L'Ouverture and his forces were able to secure Haiti's independence from France on January 1, 1804.
Prediction (with verbalized knowledge): Haiti (Saint-Domingue).

Neighbor Questions:
- The battle of Hastings in 1066 was fought in which country? (<Self> > <Wiki>)
- When Belgium declared its independence in 1830 it broke away from ___ control? (<Self> > <Wiki>)
- When did the French come to the new world? (<Self> > <Wiki>)
... (27 questions omitted)
$p_D(\text{<s>}|q)$: P(<Wiki>) = 24/30 = 0.8, P(<Self>) = 6/30 = 0.2.
$p_M(\text{<s>}|q)$: P(<Wiki>) = 0.576, P(<Self>) = 0.424.
Final Source Decision: <Self>

Figure 10: Example of SR-RAG inference where $p_D(\text{<s>}|q)$ corrects the source selection from $p_M(\text{<s>}|q)$. The model itself is selected as the knowledge source.