

STEPPER: Step-wise Knowledge Distillation for Enhancing Reasoning Ability in Multi-Step Retrieval-Augmented LM

Anonymous ACL submission

Abstract

To answer complex real-world questions, it is crucial to retrieve and integrate relevant information step-by-step to generate well-grounded responses. However, existing methods struggle to effectively distill step-specific reasoning abilities, as they do not account for the varying amount of information accessed at each reasoning step. To address this limitation, we propose **Step-wise Knowledge Distillation for Enhancing Reasoning Ability in Multi-Step Retrieval-Augmented LM (STEPPER)**. STEPPER leverages step-wise datasets and reasoning difficulty-aware training to enhance reasoning abilities essential for multi-step retrieval-augmented LM. Moreover, STEPPER is adaptable to various multi-step retrieval-augmented LM frameworks, including reasoning path-based retrieval and question decomposition-based approaches. Extensive experiments demonstrate that STEPPER outperforms existing methods on multi-hop QA datasets, with an 8B model achieving performance on par with a 70B teacher model.

1 Introduction

Large language models (LLMs) have demonstrated strong reasoning abilities across various tasks (Rae et al., 2021; Hoffmann et al., 2022; Chowdhery et al., 2023), leveraging Chain-of-Thought prompting (CoT). However, these reasoning abilities are primarily observed in large models (Wei et al., 2022; Chung et al., 2024), requiring substantial inference cost. Therefore, Knowledge Distillation (KD) methods have been introduced to effectively transfer these abilities to smaller models (Hsieh et al., 2023; Mitra et al., 2023; Lee et al., 2024).

Beyond mathematical problem-solving tasks, reasoning ability is also essential in retrieval-augmented LMs, which retrieve relevant information and generate accurate responses based on the retrieval results. Existing KD approaches train a student model to mimic the teacher’s reasoning

process, often using CoT prompting to generate rationales that guide the student through intermediate reasoning steps (Luo et al., 2023; Yu et al., 2023). However, these methods struggle with complex real-world tasks that require multi-step retrieval for effective reasoning.

To answer complex questions, a model must develop multiple reasoning abilities. For example, consider a doctor diagnosing a patient with ankle pain. The reasoning process can be broken down into three stages: (1) *Reasoning Initialization*, where the doctor identifies potential diseases based on symptoms; (2) *Reasoning Expansion*, where additional tests, such as an X-ray to check for fractures and an ultrasound for muscle damage, are conducted; and (3) *Reasoning Aggregation*, where all the information is reviewed to make a final diagnosis and treatment plan. To solve complex problems, a model needs to learn step-by-step reasoning and adapt to the varying amount of information required at each stage.

Existing KD methods are limited in these scenarios, as they fail to account for reasoning abilities and the varying amounts of information required at each step (Kang et al., 2023; Luo et al., 2023; Yu et al., 2023; Li et al., 2024). In general, these methods train the student model to generate the entire reasoning path from accumulated retrieval results, without considering the differences in the information available at each step. As shown in Figure 1, the vanilla-KD model fails to initialize the reasoning path properly, attempting to generate the entire path in the First-step with minimal information, which limits its performance in multi-step retrieval settings.

To address this limitation, we propose **Step-wise Knowledge Distillation for Enhancing Reasoning Ability in Multi-Step Retrieval-Augmented LM (STEPPER)**. STEPPER constructs a step-wise dataset using a teacher multi-step retrieval-augmented LM, enabling the model to learn reasoning abilities spe-

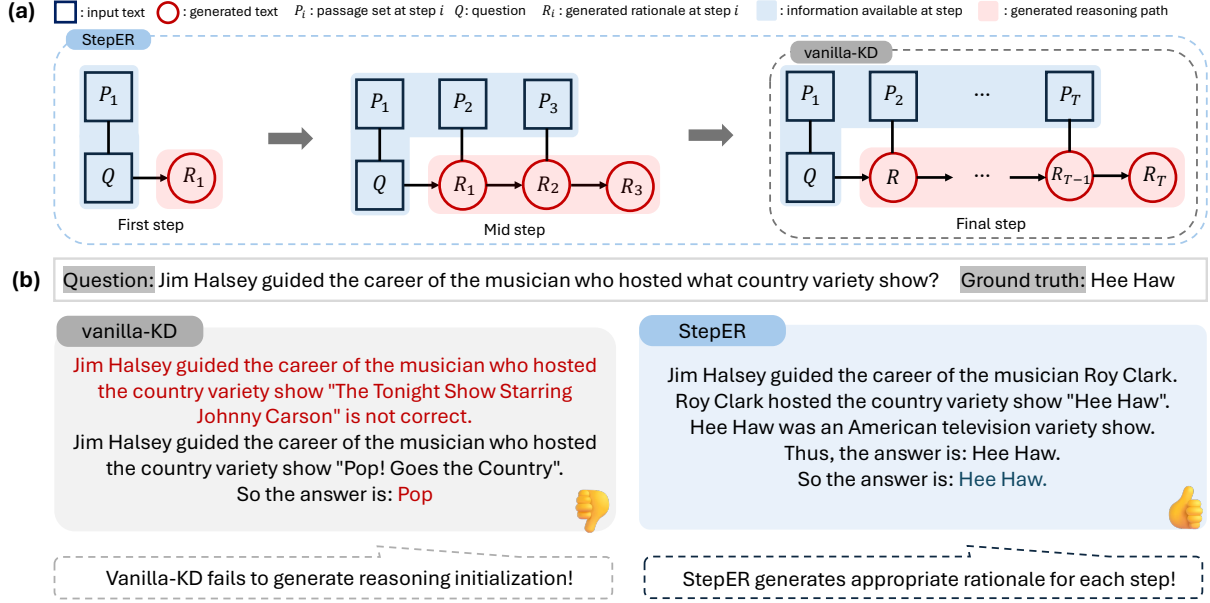


Figure 1: Comparison of vanilla-KD and STEP-ER. In (a), we show the conceptual differences in training data. First-step data represents initial reasoning based on retrieved passages, Mid-step data includes intermediate reasoning steps, and Final-step data corresponds to the step where the model generates the final answer. In (b), we provide answer examples from the vanilla-KD and STEP-ER models. vanilla-KD generates an answer in one step, often resulting in errors, while STEP-ER expands reasoning step by step to produce the correct answer.

cific to each step. First-step data helps the model initiate reasoning based on initial retrieval results, Mid-step data facilitates reasoning expansion, and Final-step data supports reasoning aggregation. This approach allows the model to acquire reasoning capabilities for complex questions while considering the information required at each step.

To further enhance reasoning ability learning, we introduce reasoning difficulty-aware training. Initially, the model focuses on tasks that are suitable for learning, gradually increasing the focus on more challenging tasks as training progresses. This adaptive approach allows the model to effectively learn reasoning abilities, optimizing the learning process according to its current state. As shown in Figure 1, a model trained with StepER successfully identifies the artist, the show hosted by the artist, the country where it aired, and ultimately produces the correct answer.

STEPER offers several advantages for answering complex questions. First, it outperforms vanilla-KD methods, with experiments showing an average accuracy improvement of approximately 9.5%. G-Eval results confirm that step-wise is crucial for enhancing reasoning abilities. Second, STEP-ER is flexible and can be applied to various multi-step retrieval-augmented LM frameworks. Further, STEP-ER is model-scalable, achieving performance comparable to a 70B teacher model with a 8B

model.

Our main contributions are as follows: (1) We categorize the essential reasoning abilities required for multi-retrieval settings and demonstrate the need for methods to enhance each ability. (2) We propose STEP-ER, a method that leverages step-wise data and reasoning difficulty-aware training to effectively learn the necessary reasoning abilities. (3) Through extensive analysis, we show that STEP-ER outperforms existing KD approaches, improving both overall performance and scalability across various model sizes.

2 Related Work

Retrieval-Augmented LM Retrieval-augmented LM has significantly improved performance in knowledge-intensive tasks such as Open-Domain Question Answering (Lewis et al., 2020; Guu et al., 2020). This model typically consists of a retriever that selects relevant documents and a generator that constructs responses based on the retrieved information (Borgeaud et al., 2022; Izacard et al., 2023; Shi et al., 2023). To answer based on documents most relevant to the question, Kim et al. (2024), Xu et al. (2023) have explored approaches that refine retrieved documents before generation, by summarizing evidence. However, Jiang et al. (2024) shows that improving the quality of retrieval results alone remains insufficient for multi-hop QA tasks,

indicating the need for more effective methods to facilitate complex reasoning in question answering.

Multi-Step Retrieval-Augmented LM To address the limitations of single-step retrieval in handling complex queries, multi-step retrieval-augmented LMs have been introduced (Trivedi et al., 2022a; Shao et al., 2023; Jeong et al., 2024). These models iteratively retrieve information throughout the reasoning process. Trivedi et al. (2022a), Shao et al. (2023) leverage previously generated rationales as queries for subsequent retrieval, while Press et al. (2022) decomposes the original question into sub-questions and answers them independently.

KD for Retrieval-Augmented LM Several studies have explored the use of teacher-generated rationales to improve the training of retrieval-augmented language models (Xu et al., 2024). In addition to simply utilizing teacher rationales, recent studies have been proposed to enhance search result quality using rationales (Kang et al., 2023) or to improve answer generation by reflecting the relevance between the retrieved passages and the question (Luo et al., 2023; Yu et al., 2023). However, these methods primarily focus on single-step retrieval settings, which limits their performance in multi-hop question answering tasks.

Recently, Asai et al. (2023) has been introduced to enhance the training of multi-step retrieval-augmented LMs by learning when to retrieve and which documents to incorporate into responses. This approach focuses on integrating high-quality search results into answers but overlooks the step-wise reasoning abilities needed for complex questions and requires additional models for training, increasing the cost.

3 Preliminaries

We formalize retrieval-augmented generation (RAG) in the context of multi-step reasoning. Specifically, let q denote the original input question, and let the reasoning process proceed over S steps. During the first $S - 1$ steps, the model produces intermediate reasoning outputs $\{r_1, r_2, \dots, r_{S-1}\}$ and in the Final-step, it generates the answer, denoted by $r_S = a$.

Single-Step RAG In the single-step RAG, the model accesses an external knowledge source only once before generating both its reasoning chain and final answer. Let P_1 be the top- K passages

retrieved from the knowledge source given the original question q . The generation process is then factorized as

$$P(R | q, P_1) \cdot P(a | q, P_1, R). \quad (1)$$

Here, the model first generates the intermediate reasoning steps R conditioned on $\{q, P_1\}$, and then produces the final answer a based on $\{q, P_1, R\}$. Although this approach simplifies the pipeline, previous works have demonstrated that it is inadequate for complex multi-hop queries that require additional (Trivedi et al., 2022a; Jeong et al., 2024; Gao et al., 2023; Shao et al., 2023; Jiang et al., 2023).

Multi-Step RAG Multi-step RAG extends single-step RAG by iteratively retrieving new passages over multiple steps. At step s , let q_s be a step search query, which is constructed based on the partial chain of reasoning $R_{<s} = \{r_1, \dots, r_{s-1}\}$. Using q_s to query the external knowledge source, we retrieve the top- K relevant passages P_s . We denote by $P_{\leq s} = \bigcup_{i=1}^s P_i$ the collection of all passages retrieved up to step s . For S total steps, the generation process is factorized as

$$\left[\prod_{s=1}^{S-1} P(r_s | q, P_{\leq s}, R_{<s}) \right] \cdot P(a | q, P_{\leq S}, R_{<S}), \quad (2)$$

By repeatedly retrieving and integrating new evidence, Multi-step RAG is naturally suited to address complex or multi-hop questions.

4 STEPER Framework

We propose a novel framework, **STEPER**, to enhance the step-specific reasoning abilities of student models.

4.1 Data Construction

According to equation (2), the accessible information in the Multi-Step RAG changes as the steps progress, and accordingly the reasoning ability required by the model varies (see Figure 1). We categorize these reasoning abilities into three groups: (a) *reasoning initialization*, (b) *reasoning expansion*, and (c) *reasoning aggregation*. To enable the student to learn these three reasoning skills from a teacher, we construct a step-wise dataset, denoted as D_{stepwise} , from the original dataset D . Given a complex QA dataset $\mathcal{D} = \{(q^{(i)}, a^{(i)})\}_{i=1}^n$, where each $q^{(i)}$ is a question and $a^{(i)}$ is its correct answer, we construct a stepwise dataset $\mathcal{D}_{\text{steps}}$ in which every sample explicitly records multiple intermediate reasoning steps with each accessible information.

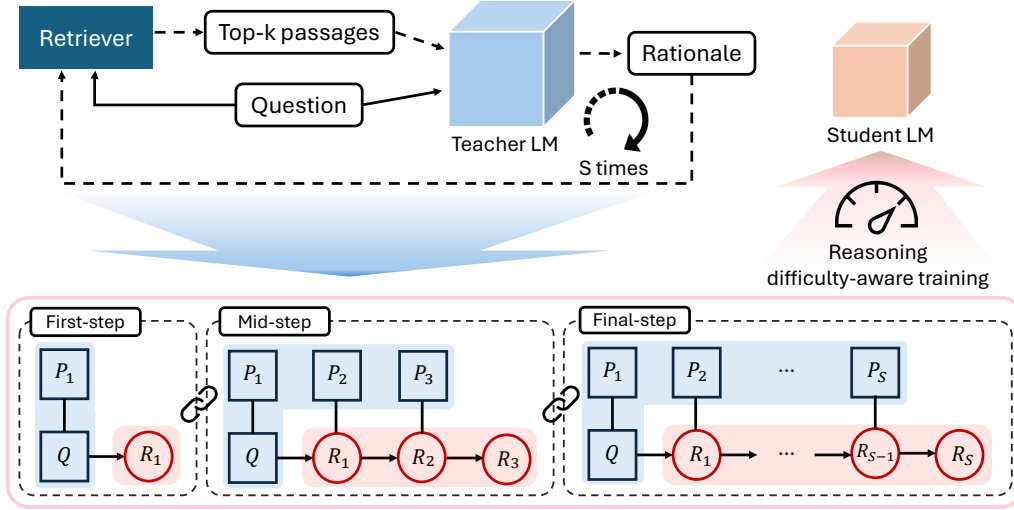


Figure 2: STEPER framework. This illustration conceptually depicts the construction of training data and the student model’s learning process. A teacher LM constructs a step-wise dataset, and the student model effectively acquires the necessary reasoning abilities for multi-step retrieval-augmented LM through reasoning difficulty-aware training using step-wise data.

Reasoning Initialization For each question $q^{(i)}$, we retrieve a first passages $P_1^{(i)}$ by querying an external knowledge source with $q^{(i)}$. We then prompt the teacher model $\mathcal{T}$ to produce the initial reasoning step $r_1^{(i)}$ from $(q^{(i)}, P_1^{(i)})$. We retain the initial reasoning step r_1 and then proceed to the next step.

Reasoning Expansion Based upon the initial rationale, we prompt the teacher model $\mathcal{T}$ to generate the next reasoning step. Specifically, at step $s > 1$, we retrieve additional passages $P_s^{(i)}$ using q_s as a step search query. Then, the cumulative information $(q^{(i)}, P_{\leq s}^{(i)}, R_{\leq s-1}^{(i)})$ is provided as input, from which $\mathcal{T}$ produces the next reasoning step $r_s^{(i)}$. Each expansion step is designed to elaborate existing reasoning by integrating new evidence while maintaining coherence with previous contexts. This iterative process continues up to a maximum of $S - 1$ steps. If at any point $r_s^{(i)}$ includes the answer flag (e.g., beginning with “So the answer is:”), we record the reasoning chain constructed up to that step and terminate the expansion step early.

Reasoning Aggregation Upon reaching the last step S , we prompt $\mathcal{T}$ to aggregate all previous reasoning steps and passages. Concretely, $\mathcal{T}$ is instructed to append a concluding statement like “So the answer is:” and explicitly provide $a^{(i)}$.

Filtering Dataset After generating all reasoning steps for each $(q^{(i)}, a^{(i)})$, we filter out samples where the teacher’s final statement does not match the ground truth $a^{(i)}$, ensuring that $\mathcal{D}_{\text{steps}}$ only con-

tains the correct reasoning processes. Ultimately, every sample in $\mathcal{D}_{\text{steps}}$ illustrates how $\mathcal{T}$ (i) *initializes* reasoning from limited context, (ii) *expands* partial reasoning with newly retrieved evidence, and (iii) *aggregates* all partial results into a final answer.

4.2 Learning Objectives

Multi-task Learning We train the student model $\mathcal{M}$ on the stepwise dataset $\mathcal{D}_{\text{steps}}$ to distill multi-step reasoning abilities. Formally, we minimize the following objective:

$$\mathcal{L} = \frac{1}{3n} \sum_{i=1}^n \left[\underbrace{\ell(\mathcal{M}(q^{(i)}, P_{\leq 1}^{(i)}), R_{\leq 1}^{(i)})}_{\text{(a) reasoning initialization}} + \underbrace{\sum_{s=2}^{S-1} \ell(\mathcal{M}(q^{(i)}, P_{\leq s}^{(i)}), R_{\leq s}^{(i)})}_{\text{(b) reasoning expansion}} + \underbrace{\ell(\mathcal{M}(q^{(i)}, P_{\leq S}^{(i)}), (R_{\leq S}^{(i)} || a^{(i)}))}_{\text{(c) reasoning aggregation}} \right], \quad (3)$$

where $\ell(\cdot, \cdot)$ is the cross-entropy between predicted and target tokens, n is the total number of samples, and $||$ in (c) denotes string concatenation.

Reasoning Difficulty-Aware Training Since each task has a different difficulty level, the model should prioritize learning the reasoning abilities that are most suitable for its current training stage (Liang and Zhang, 2020; Guo et al., 2018; Murugesan and Carbonell, 2017). To achieve this,

we apply an adaptive weighting scheme (Kendall et al., 2017; Chen et al., 2021), allowing the model to focus on adequate tasks while dynamically adjusting learning priorities at each training step. The difficulty of each task is represented as a trainable parameter σ . In Equation (3), (a), (b), and (c) correspond to L_{init} , L_{exp} , and L_{agg} respectively, then the final objective is then formulated as:

$$\mathcal{L}_{\text{final}} = \sum_{j \in \{\text{init}, \text{exp}, \text{agg}\}} \left(\frac{1}{2\sigma_j^2} L_j + \log \sigma_j \right), \quad (4)$$

where $\log \sigma_j$ functions as a regularization term. The model is adaptively trained so that tasks requiring more challenging reasoning are guided to have higher σ values, whereas less demanding tasks are guided to have lower σ values, enabling the model to dynamically reweight its training focus based on the difficulty of each task, leading to more effective multi-step reasoning.

5 Experiments

5.1 Experimental Setup

Backbone Model We use Llama3.1-Instruct 70B (Dubey et al., 2024) as our teacher model $\mathcal{T}$, with Llama3.1-Instruct 8B as the student model $\mathcal{M}$. Unless otherwise specified, all baseline methods employ Llama3.1-Instruct.

Datasets and Metrics We evaluate on three widely used multi-hop QA benchmarks that involve complex queries: 2WikiMultiHopQA (2Wiki) (Ho et al., 2020), HotpotQA (Yang et al., 2018), and MuSiQue (Trivedi et al., 2022b) that are recognized for requiring more complex and multi-step reasoning (Welbl et al., 2018; Yang et al., 2018). We report Exact Match (EM), F1, and Accuracy (Acc), where Acc measures whether the ground-truth answer is present in the model’s generated text.

Baselines We compare a wide range of retrieval-augmented generation (RAG) methods that cover both few-shot in-context learning (ICL) and knowledge distillation, while varying the number of retrieval times (0, 1, or multiple).

In ICL, we include a non-retrieval few-shot baseline for reference, since LLMs already encode a large amount of knowledge (Zhao et al., 2023). Next, We evaluate vanilla-RAG (Lewis et al., 2020), where a retriever retrieves relevant documents and a generator produces the answer conditioned on this

retrieved context. We compare SuRE (Kim et al., 2024), an advanced variant that retrieves and summarizes before verifying the final prediction. For multi-step retrieval in ICL, we compare two ways of updating the step search query: one in which the query is updated with previously generated context, as in ITER-RETGEN (Shao et al., 2023) and IRCOT (Trivedi et al., 2022a), and another where the model decomposes the original question into multiple sub-queries, as in Self-Ask (Press et al., 2022) and ReAct (Yao et al., 2023).

In knowledge distillation, SAIL (Luo et al., 2023) and CoN (Yu et al., 2023) distill context filtering strategies, helping the student identify irrelevant passages. KARD (Kang et al., 2023) distills the teacher’s reasoning while leveraging its rationale to improve retrieval. We refer to vanilla-KD, trained to generate the full reasoning path sequentially from all retrieved documents until the final answer. We compare Self-RAG (Asai et al., 2023), which learns when to retrieve and reflect the outputs in a multi-step setting. STEP-ER utilizes IRCOT-style reasoning-path-based retrieval in our experiments.

Implementation Details We follow the corpus selection and data preprocessing setup from the previous work Trivedi et al. (2022a). For passage retrieval, we adopt an off-the-shelf retriever BM25 (Robertson and Zaragoza, 2009) with a maximum of $S = 5$ retrieval steps, retrieving the top- $K = 4$ passages at each step. We train the models using a learning rate of 5×10^{-6} for total 2 epochs, along with a cosine scheduler and linear warmup. Experiments run on 4×A100 GPUs with DeepSpeed ZeRO Stage 3 and gradient checkpointing to reduce memory consumption.

5.2 Main Results

Table 1 shows the performance of various approaches on 2Wiki, HotpotQA, and MuSiQue with Llama3.1-Instruct. We first note that single-time retrieval methods struggle to address complex queries, and even recent improvements (Kim et al., 2024) exhibit a noticeable gap compared to multi-time retrieval. In addition, an accuracy gap persists between 8B and 70B models under multi-step RAG ICL, highlighting the importance of model size in complex reasoning tasks.

STEP-ER stands out as it delivers the best performance among knowledge distillation methods, achieving a 9.5% average accuracy improvement

Ret. times		2Wiki			HotpotQA			MuSiQue			Avg.		
		EM	F1	Acc	EM	F1	Acc	EM	F1	Acc	EM	F1	Acc
<i>In-Context Learning</i>													
<i>No</i>	Llama3.1 8B	29.83	35.59	33.69	29.18	38.76	35.01	8.68	17.91	13.22	22.56	30.75	27.31
	Llama3.1 70B	45.47	51.09	47.89	40.61	51.25	45.86	16.19	25.94	23.28	34.09	42.76	39.01
	gpt-4o-mini	25.51	40.80	27.09	28.15	41.25	35.81	11.84	24.03	15.89	21.83	35.36	26.26
	gpt-4o	52.26	65.88	53.70	40.69	57.24	48.05	21.62	35.22	28.50	38.19	52.78	43.42
<i>Single</i>	vanilla-RAG 8B	35.97	43.10	38.88	38.25	49.08	46.15	11.18	20.91	22.57	28.46	37.69	35.86
	vanilla-RAG 70B	51.01	57.80	53.83	45.25	56.30	52.93	19.84	30.79	31.58	38.70	48.29	46.08
	SuRE 70B	25.20	41.34	41.20	30.60	48.23	41.00	11.60	22.00	19.40	22.46	37.19	33.86
<i>Multi</i>	ITER-RETGEN3 70B	44.60	50.92	46.20	48.20	60.12	53.40	24.20	33.17	30.00	39.00	48.07	43.20
	ITER-RETGEN4 70B	44.20	50.54	45.60	49.40	60.92	54.60	24.80	32.98	30.40	39.46	48.14	43.53
	ITER-RETGEN5 70B	44.00	50.35	45.60	49.40	60.51	54.80	24.00	31.92	29.60	39.13	47.59	43.33
	IRCOT 8B	41.80	49.94	44.80	43.40	53.82	50.80	17.20	27.57	28.40	34.13	43.77	41.33
	IRCOT 70B	60.16	67.06	62.37	49.60	61.31	57.23	24.30	35.29	34.74	44.68	54.55	51.45
	Self-Ask 8B	38.80	47.41	43.00	40.80	52.00	48.20	15.83	23.58	23.85	31.81	41.00	38.35
	Self-Ask 70B	57.80	66.44	61.00	50.60	62.60	59.40	25.20	36.68	33.80	44.53	55.24	51.40
	ReAct 8B	40.20	49.50	43.00	33.60	43.96	39.60	14.80	24.73	21.20	29.53	39.40	34.60
	ReAct 70B	59.40	68.58	61.60	46.00	59.89	53.40	28.20	39.46	35.60	44.53	55.98	50.20
<i>Knowledge Distillation</i>													
<i>Single</i>	SAIL	47.90	54.06	49.50	44.56	56.30	51.41	6.41	16.34	10.62	32.96	42.23	37.18
	KARD	47.80	54.48	51.40	43.80	54.59	53.00	14.60	25.54	24.60	35.40	44.87	43.00
	CoN	45.66	53.93	48.89	42.46	53.34	51.00	16.36	26.85	25.86	34.96	44.70	41.91
<i>Multi</i>	Self-RAG	41.15	46.99	42.82	36.85	44.88	41.26	9.16	17.19	12.80	29.05	36.35	32.29
	vanilla-KD	60.06	65.55	62.16	46.40	57.28	54.80	20.92	32.46	30.13	42.46	51.76	49.03
	STEPPER	63.60	69.45	66.00	51.00	62.80	61.00	23.59	36.13	34.07	46.06	56.12	53.69

Table 1: Overall experimental results with **Llama3.1-Instruct** as the base model. All listed models (SAIL, KARD, CoN, Self-RAG, and vanilla-KD) are trained with Llama3.1-Instruct 8B under the Knowledge Distillation criteria. Averages (Avg.) are computed across three datasets: 2Wiki, HotpotQA, and MuSiQue. The number for ITER-RETGEN represents the maximum number of retrieval steps.

over vanilla-KD and outperforming all baselines on 2Wiki and HotpotQA. These results underscore how STEPPER effectively inherits step-wise reasoning abilities from the teacher model, enabling a smaller student model to close the gap in reasoning performance.

6 Analysis

6.1 Effectiveness of Step Data in Enhancing Reasoning Abilities

We conduct an experiment to evaluate the effectiveness of step data in enhancing reasoning abilities required for multi-step retrieval-augmented LM. We categorize the necessary reasoning abilities into three types for evaluation: (1) *Reasoning Initialization*, (2) *Reasoning Expansion*, and (3) *Reasoning Aggregation*, as described in Section 4. To evaluate these abilities, we perform binary classification for each criterion using GPT-4, evaluated on the HotpotQA test dataset. The detailed prompt used for evaluation is provided in the Appendix C. We train the models using various step data configurations, specifically: Vanilla-KD ($S=5$), Vanilla-KD+First-step ($S=1,5$), Vanilla-KD+First-

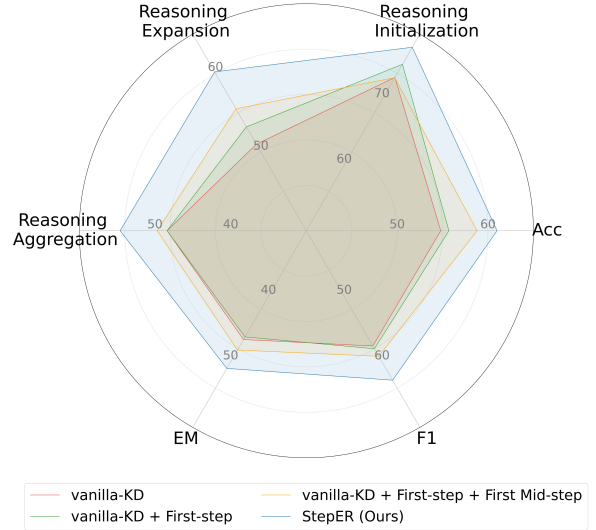


Figure 3: GPT evaluation results (Reasoning Initialization, Reasoning Expansion, and Reasoning Aggregation) on the HotpotQA for various step data configurations (vanilla-KD, +First-step, +First and First Mid-step, and STEPPER). STEPPER, which utilizes all available step data, achieves the highest performance across all evaluation criteria, demonstrating the effectiveness of step-wise training for multi-step retrieval.

step+First Mid-step ($S=1,2,5$), and STEPPER (all step data), with a maximum of $S = 5$ retrieval

steps.

Vanilla-KD relies solely on Final-step data and struggles to capture detailed intermediate reasoning. In contrast, adding First-step data strengthens the ability to initiate reasoning (*Reasoning Initialization*). By offering a clear starting point for multi-step reasoning, the model can more effectively identify and focus on relevant information at the beginning of the reasoning process. Furthermore, incorporating the First-step data and First Mid-step data improves the expansion process (*Reasoning Expansion*), enabling the model to elaborate on its initial line of reasoning before arriving at the final conclusion. Finally, STEP-ER, which jointly leverages all step data, outperforms all other models, confirming that step-wise data enhances the reasoning abilities required for multi-step retrieval settings.

6.2 Effectiveness of Difficulty-Aware Adaptive Weighting Strategy

Strategy	HotpotQA			MuSiQue		
	EM	F1	Acc	EM	F1	Acc
Uniform ($\lambda = 1, 1, 1$)	50.40	61.57	58.40	21.67	33.28	33.58
Weight First ($\lambda = 1.5, 1, 0.5$)	49.10	61.63	57.70	21.04	31.24	32.46
Weight Last ($\lambda = 0.5, 1, 1.5$)	48.80	60.78	58.00	21.91	33.85	33.37
Difficulty-Aware (Ours)	51.00	62.80	61.00	23.59	36.13	34.07

Table 2: Comparison of our Difficulty-Aware adaptive weighting strategy against several fixed-weight baselines. The table illustrates how adaptively learning σ_j with $\lambda_j = \frac{1}{2\sigma_j^2}$ to control the relative difficulty of each task leads to consistent improvements on both HotpotQA and MuSiQue, thereby enabling a more balanced and effective multi-step reasoning process.

As introduced in Equation (4), our overall loss consists of three partial losses $\{L_{init}, L_{exp}, L_{agg}\}$, each scaled by $\frac{1}{2\sigma_j^2}$. Specifically, We set $\lambda_j = \frac{1}{2\sigma_j^2}$ adaptively control the relative difficulty of each task. Table 2 compares this Difficulty-Aware strategy against several fixed-weight baselines. Notably, we observe consistent improvements on both HotpotQA and MuSiQue. This indicates that adaptively learning σ_j based on task difficulty leads to a more balanced and effective multi-step reasoning process.

6.3 Applicability to Another Multi-time Retrieval Approach

We further investigate the generality of our step-wise knowledge distillation by integrating STEP-ER with Self-Ask, another multi-time retrieval framework where each step search query is generated from a decomposition of the original question. As shown in Table 3, STEP-ER consistently shows a

Model	HotpotQA			MuSiQue		
	EM	F1	Acc	EM	F1	Acc
Self-Ask 8B	40.80	52.50	48.20	15.83	23.58	23.85
vanilla-KD	47.60	60.11	56.40	26.90	38.92	37.00
STEP-ER	49.80	62.33	57.80	28.20	40.52	38.80
Self-Ask 70B	50.60	62.60	59.40	25.20	36.68	33.80

Table 3: Evaluation results for Self-Ask on HotpotQA and MushiQue on Llama3.1-Instruct. We compare the teacher model (Self-Ask 70B) with student models (8B) distilled through either vanilla-KD or STEP-ER.

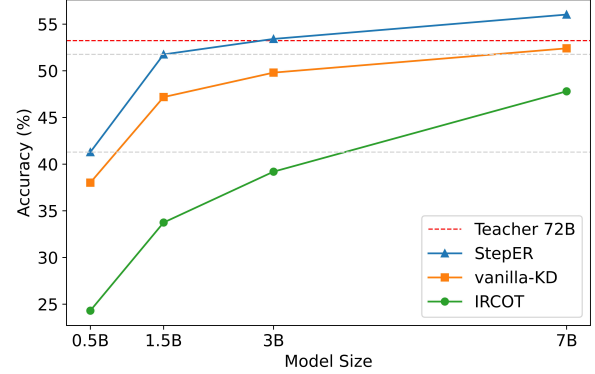


Figure 4: Model scalability of STEP-ER on Qwen2.5-Instruct. We compare models of varying sizes (0.5B, 1.5B, 3B, 7B) and show how STEP-ER scales effectively while maintaining strong multi-time reasoning performance.

1–2% performance gain over vanilla-KD on both HotpotQA and MuSiQue, highlighting the effectiveness of explicitly distilling intermediate rationales at each retrieval step rather than relying solely on supervision from the Final-step data. In addition, STEP-ER achieves substantial improvements over the Self-Ask 8B baseline, boosting its accuracy by 9.6% on HotpotQA and by 14.95% on MuSiQue. Consequently, these results demonstrate that our approach can integrate seamlessly with various multi-step retrieval-augmented LMs.

6.4 Model Scalability

Figure 4 shows that STEP-ER consistently achieves the highest accuracy across all Qwen2.5-Instruct model sizes (0.5B, 1.5B, 3B, and 7B) (Yang et al., 2024) on HotpotQA. Notably, the 3B STEP-ER model nearly matches the performance of the Qwen2.5-Instruct 72B teacher, while the 7B STEP-ER even surpasses it. Furthermore, 3B STEP-ER outperforms the 7B vanilla-KD, and 1.5B STEP-ER surpasses the 3B vanilla-KD, indicating that STEP-ER can effectively bridge model-scale gaps by distilling step-wise reasoning abilities. These findings underscore the practicality of STEP-ER in resource-constrained scenarios, where smaller models can

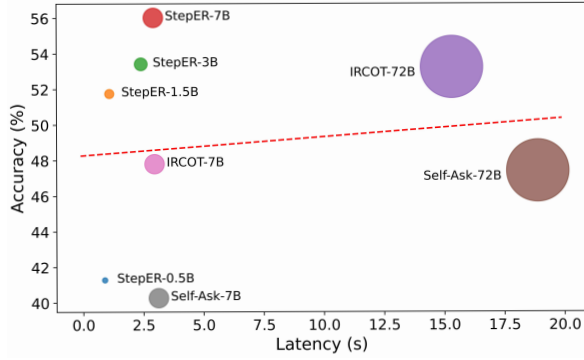


Figure 5: Accuracy (%) versus Latency (s) of StepER on Qwen2.5-Instruct. Marker size indicates model parameter count. StepER models achieve superior performance with lower latency than larger models, offering the best trade-off between efficiency and effectiveness.

achieve performance levels comparable to much larger counterparts (Sanh, 2019; Liu et al., 2024).

6.5 Trade-off Between Latency and Accuracy

We measure the latency as the average inference time per sample on HotpotQA with Qwen2.5-Instruct models. Figure 5 illustrates the trade-off between inference latency and accuracy for different models, where the marker size indicates the model’s parameter count. StepER-7B surpasses 70B-scale models in terms of accuracy, yet requires only a fraction of their latency. Thus, our evaluation confirms that StepER-7B stands out as the most efficient and effective model, delivering the best trade-off between latency and accuracy.

6.6 Out-of-Domain Adaptation

To evaluate the transferability of our approach, we conducted out-of-domain experiments by training the model on one dataset and testing it on another. We use the 2Wiki (2W), HotpotQA (HQ), and MuSiQue (MQ) datasets. Figure 6 shows the performance accuracy of the two methods, StepER and vanilla-KD, across four domain adaptation scenarios: HQ→2W, HQ→MQ, MQ→2W, and MQ→HQ.

StepER consistently outperforms vanilla-KD across all four domain adaptation scenarios. In each case, StepER exhibits higher accuracy than vanilla-KD, with gains ranging from 1% to 4%. This indicates that distilling the teacher’s step-wise reasoning ability offers better transferability to out-of-domain tasks than relying solely on the aggregation step as in vanilla-KD. Overall, these results suggest that learning step-specific reasoning abilities not only enhances in-domain reasoning but also leads to stronger generalization to unseen datasets.

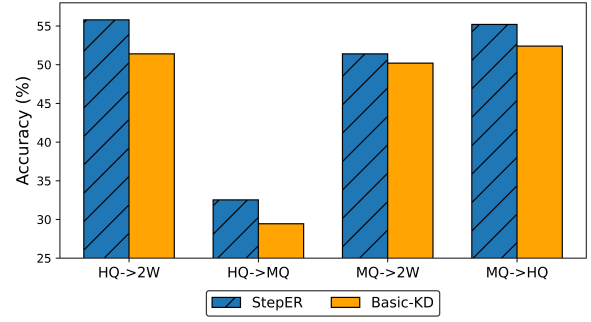


Figure 6: Out-of-domain adaptation results for StepER versus vanilla-KD across four domain transfer scenarios: HQ→2W, HQ→MQ, MQ→2W, and MQ→HQ. StepER consistently outperforms vanilla-KD, demonstrating stronger cross-domain generalization.

6.7 Qualitatively Analysis

Table 4 shows a HotpotQA example illustrating the difference in answers between vanilla-KD and StepER. While vanilla-KD correctly identifies ‘Comic Book Girl 19’ and the university, it incorrectly answers with ‘Savannah, Georgia,’ failing to align with the question about the university’s European location. In contrast, StepER identifies all relevant information about ‘Comic Book Girl 19’, the university, and its European location, providing the correct answer. Vanilla-KD learns without considering each reasoning ability, leading to the failure to extract key information relevant to the question. On the other hand, StepER learns reasoning abilities at each step, allowing the model to generate the correct answer progressively.

7 Conclusion

We propose StepER, a framework for learning the reasoning abilities required for multi-retrieval augmented LMs. By categorizing reasoning abilities into initialization, expansion, and aggregation, we treat each as a separate task and use step data for multi-task learning. We further introduce reasoning difficulty-aware training, dynamically adjusting task importance as learning progresses. Extensive experiments show that StepER improves reasoning abilities and outperforms existing methods across various model sizes and multi-step retrieval settings. Given its adaptability, StepER can enhance performance in diverse domains. Future work will explore its generalizability across various tasks.

Limitations

We propose STEP-ER, a method for effectively learning the reasoning abilities required for multi-step retrieval-augmented LMs, which demonstrates strong performance across several multi-hop datasets. Given the nature of knowledge distillation (KD), where the student model learns from the teacher model’s rationale, it is crucial to filter the training dataset to prevent propagating errors from the teacher model to the student. In this study, we use a filtering method based solely on whether the final answer is correct. However, since errors can occur in the reasoning path even when the final answer is correct, future work may focus on filtering based on the correctness of the reasoning path at each step to further enhance performance. Additionally, incorporating parameter-efficient fine-tuning methods could lead to more efficient learning.

Ethical Considerations

We used publicly available datasets, including 2WikiMultiHotpotQA, HotpotQA, and MuSiQue. For models, we employed publicly released LLaMA-3.1-Instruct, Qwen-2.5-Instruct, GPT-4o, and GPT-4o-mini. Therefore, we do not anticipate significant ethical concerns from our work.

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*.

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

Shijie Chen, Yu Zhang, and Qiang Yang. 2021. [Multi-task learning in natural language processing: An overview](#). *ACM Computing Surveys*.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. 2023. Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 24(240):1–113.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al.

2024. Scaling instruction-finetuned language models. *Journal of Machine Learning Research*, 25(70):1–53.

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

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

Michelle Guo, Albert Haque, De-An Huang, Serena Yeung, and Li Fei-Fei. 2018. Dynamic task prioritization for multitask learning. In *Proceedings of the European Conference on Computer Vision (ECCV)*.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Mingwei Chang. 2020. Retrieval augmented language model pre-training. In *International conference on machine learning*, pages 3929–3938. PMLR.

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

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, et al. 2022. An empirical analysis of compute-optimal large language model training. *Advances in Neural Information Processing Systems*, 35:30016–30030.

Cheng-Yu Hsieh, Chun-Liang Li, Chih-Kuan Yeh, Hootan Nakhost, Yasuhisa Fujii, Alexander Ratner, Ranjay Krishna, Chen-Yu Lee, and Tomas Pfister. 2023. Distilling step-by-step! outperforming larger language models with less training data and smaller model sizes. *arXiv preprint arXiv:2305.02301*.

Gautier Izacard, Patrick Lewis, Maria Lomeli, Lucas Hosseini, Fabio Petroni, Timo Schick, Jane Dwivedi-Yu, Armand Joulin, Sebastian Riedel, and Edouard Grave. 2023. Atlas: Few-shot learning with retrieval augmented language models. *Journal of Machine Learning Research*, 24(251):1–43.

Soyeong Jeong, Jinheon Baek, Sukmin Cho, Sung Ju Hwang, and Jong C Park. 2024. Adaptive-rag: Learning to adapt retrieval-augmented large language models through question complexity. *arXiv preprint arXiv:2403.14403*.

Zhengbao Jiang, Frank Xu, Luyu Gao, Zhiqing Sun, Qian Liu, Jane Dwivedi-Yu, Yiming Yang, Jamie Callan, and Graham Neubig. 2023. [Active retrieval augmented generation](#). In *Proceedings of the 2023*

641	<i>Conference on Empirical Methods in Natural Language Processing</i> , pages 7969–7992, Singapore. Association for Computational Linguistics.	694
642		695
643		696
644	Zhouyu Jiang, Mengshu Sun, Lei Liang, and Zhiqiang Zhang. 2024. Retrieve, summarize, plan: Advancing multi-hop question answering with an iterative approach. <i>arXiv preprint arXiv:2407.13101</i> .	697
645		698
646		699
647		
648	Minki Kang, Seanie Lee, Jinheon Baek, Kenji Kawaguchi, and Sung Ju Hwang. 2023. Knowledge-augmented reasoning distillation for small language models in knowledge-intensive tasks. <i>Advances in Neural Information Processing Systems</i> , 36:48573–48602.	700
649		701
650		702
651		703
652		
653		
654	Alex Kendall, Yarin Gal, and Roberto Cipolla. 2017. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics . <i>2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition</i> , pages 7482–7491.	704
655		705
656		706
657		707
658		
659	Jaehyung Kim, Jaehyun Nam, Sangwoo Mo, Jongjin Park, Sang-Woo Lee, Minjoon Seo, Jung-Woo Ha, and Jinwoo Shin. 2024. Sure: Summarizing retrievals using answer candidates for open-domain qa of llms. <i>arXiv preprint arXiv:2404.13081</i> .	708
660		709
661		710
662		711
663		712
664		713
665	Hojae Lee, Junho Kim, and SangKeun Lee. 2024. Mentor-kd: Making small language models better multi-step reasoners. <i>arXiv preprint arXiv:2410.09037</i> .	714
666		715
667		716
668		717
669	Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. <i>Advances in Neural Information Processing Systems</i> , 33:9459–9474.	718
670		719
671		720
672		
673		
674	Xiang Li, Shizhu He, Fangyu Lei, JunYang JunYang, Tianhuang Su, Kang Liu, and Jun Zhao. 2024. Teaching small language models to reason for knowledge-intensive multi-hop question answering. In <i>Findings of the Association for Computational Linguistics: ACL 2024</i> , pages 7804–7816.	721
675		722
676		723
677		724
678		725
679		
680	Sicong Liang and Yu Zhang. 2020. A simple general approach to balance task difficulty in multi-task learning . <i>Preprint</i> , arXiv:2002.04792.	726
681		727
682		728
683		729
684	Zechun Liu, Changsheng Zhao, Forrest Iandola, Chen Lai, Yuandong Tian, Igor Fedorov, Yunsang Xiong, Ernie Chang, Yangyang Shi, Raghuraman Krishnamoorthi, et al. 2024. Mobilellm: Optimizing sub-billion parameter language models for on-device use cases. <i>arXiv preprint arXiv:2402.14905</i> .	730
685		731
686		732
687		733
688		734
689		735
690	Hongyin Luo, Yung-Sung Chuang, Yuan Gong, Tianhua Zhang, Yoon Kim, Xixin Wu, Danny Fox, Helen Meng, and James Glass. 2023. Sail: Search-augmented instruction learning. <i>arXiv preprint arXiv:2305.15225</i> .	736
691		737
692		738
693		739
		740
	Arindam Mitra, Luciano Del Corro, Shweti Mahajan, Andres Coda, Clarisse Simoes, Sahaj Agarwal, Xuxi Chen, Anastasia Razdaibiedina, Erik Jones, Kriti Aggarwal, et al. 2023. Orca 2: Teaching small language models how to reason. <i>arXiv preprint arXiv:2311.11045</i> .	741
		742
		743
		744
		745
	Keerthiram Murugesan and Jaime G. Carbonell. 2017. Self-paced multitask learning with shared knowledge . In <i>International Joint Conference on Artificial Intelligence</i> .	746
		747
		748
	Ofir Press, Muru Zhang, Sewon Min, Ludwig Schmidt, Noah A Smith, and Mike Lewis. 2022. Measuring and narrowing the compositionality gap in language models. <i>arXiv preprint arXiv:2210.03350</i> .	
	Jack W Rae, Sebastian Borgeaud, Trevor Cai, Katie Millican, Jordan Hoffmann, Francis Song, John Aslanides, Sarah Henderson, Roman Ring, Susannah Young, et al. 2021. Scaling language models: Methods, analysis & insights from training gopher. <i>arXiv preprint arXiv:2112.11446</i> .	
	Stephen Robertson and Hugo Zaragoza. 2009. The probabilistic relevance framework: Bm25 and beyond . <i>Foundations and Trends® in Information Retrieval</i> , 3(4):333–389.	
	V Sanh. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. <i>arXiv preprint arXiv:1910.01108</i> .	
	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. <i>arXiv preprint arXiv:2305.15294</i> .	
	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. <i>arXiv preprint arXiv:2301.12652</i> .	
	Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. 2022a. Interleaving retrieval with chain-of-thought reasoning for knowledge-intensive multi-step questions. <i>arXiv preprint arXiv:2212.10509</i> .	
	Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. 2022b. MuSiQue: Multi-hop questions via single-hop question composition . <i>Transactions of the Association for Computational Linguistics</i> , 10:539–554.	
	Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, et al. 2022. Emergent abilities of large language models. <i>arXiv preprint arXiv:2206.07682</i> .	
	Johannes Welbl, Pontus Stenetorp, and Sebastian Riedel. 2018. Constructing datasets for multi-hop reading comprehension across documents . <i>Transactions of</i>	

749 *the Association for Computational Linguistics*, 6:287–
750 302.

751 Fangyuan Xu, Weijia Shi, and Eunsol Choi. 2023. Re-
752 comp: Improving retrieval-augmented lms with com-
753 pression and selective augmentation. *arXiv preprint*
754 *arXiv:2310.04408*.

755 Xiaohan Xu, Ming Li, Chongyang Tao, Tao Shen,
756 Reynold Cheng, Jinyang Li, Can Xu, Dacheng Tao,
757 and Tianyi Zhou. 2024. A survey on knowledge dis-
758 tillation of large language models. *arXiv preprint*
759 *arXiv:2402.13116*.

760 An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui,
761 Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu,
762 Fei Huang, Haoran Wei, et al. 2024. Qwen2. 5 tech-
763 nical report. *arXiv preprint arXiv:2412.15115*.

764 Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio,
765 William Cohen, Ruslan Salakhutdinov, and Christo-
766 pher D. Manning. 2018. [HotpotQA: A dataset for](#)
767 [diverse, explainable multi-hop question answering](#).
768 In *Proceedings of the 2018 Conference on Empiri-
769 cal Methods in Natural Language Processing*, pages
770 2369–2380, Brussels, Belgium. Association for Com-
771 putational Linguistics.

772 Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak
773 Shafran, Karthik Narasimhan, and Yuan Cao. 2023.
774 ReAct: Synergizing reasoning and acting in language
775 models. In *International Conference on Learning*
776 *Representations (ICLR)*.

777 Wenhao Yu, Hongming Zhang, Xiaoman Pan, Kaixin
778 Ma, Hongwei Wang, and Dong Yu. 2023. Chain-of-
779 note: Enhancing robustness in retrieval-augmented
780 language models. *arXiv preprint arXiv:2311.09210*.

781 Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang,
782 Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen
783 Zhang, Junjie Zhang, Zican Dong, Yifan Du, Chen
784 Yang, Yushuo Chen, Zhipeng Chen, Jinhao Jiang,
785 Ruiyang Ren, Yifan Li, Xinyu Tang, Zikang Liu,
786 Peiyu Liu, Jian-Yun Nie, and Ji-Rong Wen. 2023.
787 [A survey of large language models](#). *arXiv preprint*
788 *arXiv:2303.18223*.

A Additional Experimental Setups

A.1 Datasets

We use publicly available multi-hop datasets mentioned in IRCOT. The characteristics of each dataset are as follows:

- **2WikiMultiHopQA:** A dataset constructed using Wikipedia documents and a knowledge graph, requiring a two-hop reasoning process to answer questions.
- **HotpotQA:** A dataset where annotators created questions and answers based on multiple Wikipedia articles.
- **MuSiQue:** A dataset formed by combining multiple single-hop questions into multi-hop questions requiring 2 to 4 hops.

Following the experimental setup of IRCOT, we construct a corpus by merging the labeled documents in each dataset. We randomly sample 50,000 instances from the training data of each dataset. Since MuSiQue contains fewer than 50,000 training instances, we use its entire training set. After filtering, the final number of training samples used is 33,584 for 2WikiMultiHopQA, 30,572 for HotpotQA, and 5,515 for MuSiQue. For validation and testing, we randomly sample 500 instances from the original validation set of each dataset to construct the validation and test datasets.

A.2 Baselines

We employ the following models for our experiments. Detailed prompts for each model are provided in Section C

- **Few-shot In-Context Learning**

1. **No Retrieval:** The LLM generates answers by directly using the question as a prompt.
2. **Single-Step Retrieval:** The question is used as a query to search the corpus once. The top- k retrieved documents are prepended to the question as input. In the case of SuRE, ...
3. **Multi-Step Retrieval:** Multiple retrieval steps are performed according to each model’s methodology to generate the final answer. For Self-Ask and ReAct, we follow the prompts provided in ITER-RETGEN.

- **Knowledge Distillation**

1. **Single-Step Retrieval:** In the case of SAIL, we follow the original approach using a RoBERTa entailment classification model to assess the relevance between retrieved documents and the question. Based on this relevance score, the retrieved results are formatted according to SAIL’s specifications and combined with the question as input. KARD fine-tunes smaller language models to produce rationales by leveraging augmented knowledge from an external knowledge base, assisted by a neural reranker. CoN employs a dedicated prompt for document selection to filter out irrelevant passages and thereby enhance reasoning performance. The student models are then distilled with the teacher’s noise-filtering signal.
2. **Multi-Step Retrieval:** For Self-RAG, we build the critic model and generator using the Llama3.1-Instruct model, which we then use during inference. For vanilla-KD, only the Final-step data from the constructed step-wise training dataset is used for training.

B Additional Experiments

B.1 Qualitative Analysis

Table 4 presents a HotpotQA example that highlights the difference in responses between vanilla-KD and STEP-ER. In the case of vanilla-KD, the model successfully answers questions about the identity of ‘Comic Book Girl 19’ and the university from which she graduated. However, despite being asked about

Question: In what european city is a location of the college from which the woman known as Comic Book Girl 19 received her degree? **Answer:** Lacoste, France

Retrieved document #1:

Wikipedia Title: Comic Book Girl 19

Danika Lee Massey, also known as Comic Book Girl 19 or CBG19, is a YouTube personality known for her commentaries on comics, films, books, and television shows. She has a degree in sequential art from the Savannah College of Art and Design.

Retrieved document #2:

Wikipedia Title: Savannah College of Art and Design

Savannah College of Art and Design (SCAD), is a private, nonprofit, accredited university with locations in Savannah, Georgia; Atlanta, Georgia; Hong Kong; and Lacoste, France.

(Other retrieved documents are omitted.)

vanilla-KD: The woman known as Comic Book Girl 19 received her degree from the Savannah College of Art and Design. The Savannah College of Art and Design is located in Savannah, Georgia. Savannah, Georgia is in the United States. So the answer is: Savannah, Georgia

STEPER(Ours): The woman known as Comic Book Girl 19 received her degree from the Savannah College of Art and Design. The Savannah College of Art and Design is located in Savannah, Georgia, United States. However, the Savannah College of Art and Design also has locations in Hong Kong and Lacoste, France. So the answer is: Lacoste, France

Table 4: Qualitative Analysis. In comparison to vanilla-KD, our StepER demonstrates the ability to expand reasoning and extract more relevant information from the question, resulting in a more accurate answer, as shown in the HotpotQA example.

{knowledge}
Q: Answer the following question by reasoning step-by-step.
{Question}
A:

Table 5: QA prompt for IRCOT.

the university’s European location, the model incorrectly provides the location of a university in another country. This highlights that vanilla-KD fails to incorporate the reasoning abilities required for each step, which is crucial in a multi-retrieval setting. In contrast, STEP-ER successfully identifies all the relevant details about ‘Comic Book Girl 19,’ the university, and its European location, resulting in the correct answer. Vanilla-KD’s failure to consider reasoning abilities at each step prevents it from extracting key information necessary to answer the question. On the other hand, STEP-ER’s multi-step approach enables it to progressively extract and combine reasoning abilities, ultimately generating the correct response.

C Prompts

{knowledge}

Q: Answer the following question by reasoning step-by-step.

Are both Kurram Garhi and Trojkrsti located in the same country?

A: Kurram Garhi is located in the country of Pakistan. Trojkrsti is located in the country of Republic of Macedonia. Thus, they are not in the same country. So the answer is: no.

{knowledge}

Q: Answer the following question by reasoning step-by-step.

When did the director of film Laughter In Hell die?

A: The film Laughter In Hell was directed by Edward L. Cahn. Edward L. Cahn died on August 25, 1963. So the answer is: August 25, 1963.

{knowledge}

Q: Answer the following question by reasoning step-by-step.

What is the cause of death of Grand Duke Alexei Alexandrovich Of Russia's mother?

A: The mother of Grand Duke Alexei Alexandrovich of Russia is Maria Alexandrovna. Maria Alexandrovna died from tuberculosis. So the answer is: tuberculosis.

{knowledge}

Q: Answer the following question by reasoning step-by-step.

Are the directors of films The Sun of the Sleepless and Nevada (1927 film) both from the same country?

A: The director of Sun of the Sleepless is Temur Babluani. The director of Nevada (1927 film) is John Waters. John Waters is from the country of America. Temur Babluani is from the country of Georgia. Thus, John Walters and Temur Babluani are not from the same country. So the answer is: no.

{knowledge}

Q: Answer the following question by reasoning step-by-step.

When was the director of film P.S. Jerusalem born?

A: P.S. Jerusalem was directed by Danae Elon. Danae Elon was born on December 23, 1970. So the answer is: December 23, 1970.

{knowledge}

Q: Answer the following question by reasoning step-by-step.

When did the director of film Hypocrite (Film) die?

A: The film Hypocrite was directed by Miguel Morayta. Miguel Morayta died on 19 June 2013. So the answer is: 19 June 2013.

{knowledge}

Q: Answer the following question by reasoning step-by-step.

Where did the director of film Maddalena (1954 Film) die?

A: The film Maddalena is directed by Augusto Genina. Augusto Genina died in Rome. So the answer is: Rome.

Table 6: 7-Shot Demonstrations for IRCOT on 2WikiMultiHopQA.

{knowledge}

Q: Answer the following question by reasoning step-by-step. Nobody Loves You was written by John Lennon and released on what album that was issued by Apple Records, and was written, recorded, and released during his 18 month separation from Yoko Ono?

A: The album issued by Apple Records, and written, recorded, and released during John Lennon's 18 month separation from Yoko Ono is Walls and Bridges. Nobody Loves You was written by John Lennon on Walls and Bridges album. So the answer is: Walls and Bridge

{knowledge}

Q: Answer the following question by reasoning step-by-step.
When did the director of film Laughter In Hell die?

A: The film Laughter In Hell was directed by Edward L. Cahn. Edward L. Cahn died on August 25, 1963. So the answer is: August 25, 1963.

{knowledge}

Q: Answer the following question by reasoning step-by-step. Who was born first, James D Grant, who uses the pen name of Lee Child, or Bernhard Schlink?

A: James D Grant, who uses the pen name of Lee Child, was born in 1954. Bernhard Schlink was born in 1944. Thus, Bernhard Schlink was born first. So the answer is: Bernhard Schlink.

{knowledge}

Q: Answer the following question by reasoning step-by-step. Which band formed first, Sponge Cola or Hurricane No. 1?

A: Sponge Cola band was formed in 1998. Hurricane No. 1 was formed in 1996. Thus, Hurricane No. 1 band formed the first. So the answer is: Hurricane No. 1.

{knowledge}

Q: Answer the following question by reasoning step-by-step. In which state of Australia will you find the themed lands Ocean parade and DreamWorks Experience both within the Dreamworld theme park complex on the Gold Coast?

A: The themed land of Ocean parade is in the state of Queensland in Australia. The themed land of The DreamWorks Experience is in the state of Queensland in Australia. Thus, both Ocean parade and The DreamWorks Experience are in the state of Queensland. So the answer is: Queensland.

{knowledge}

Q: Answer the following question by reasoning step-by-step. Mister Magoo's Christmas Carol was produced by the same studio that produced a film that featured the only animated-film role by who?

A: Mister Magoo's Christmas Carol was produced by United Productions of America studio. United Productions of America studio produced a film Gay Purr-we, which features the voice of Judy Garland in her only animated-film role. So the answer is: Judy Garland.

{knowledge}

Q: Answer the following question by reasoning step-by-step. How many awards did the "A Girl Like Me" singer win at the American Music Awards of 2012?

A: The singer of "A Girl Like Me" singer is Rihanna. In the American Music Awards of 2012, Rihanna won one award. So the answer is: one.

Table 7: 7-Shot Demonstrations for IRCOT on HotpotQA.

{knowledge}

Q: Answer the following question by reasoning step-by-step. What is the headquarters for the organization who sets the standards for ISO 21500?

A: The standards for ISO 21500 were set by International Organization for Standardization. The International Organization for Standardization has headquarters in Geneva. So the answer is: Geneva.

{knowledge}

Q: Answer the following question by reasoning step-by-step. When did Britain withdraw from the country containing Hoora?

A: Hoora is in the country of Bahrain. Britain withdrew from Bahrain in 1971. So the answer is: 1971.

{knowledge}

Q: Answer the following question by reasoning step-by-step. When did Britain withdraw from the country where the village of Wadyan is found?

A: Wadyan is in the country of Bahrain. Britain withdraw from Bahrain in 1971. So the answer is: 1971.

{knowledge}

Q: Answer the following question by reasoning step-by-step. What shares a border with Rivière-Verte in the province WRSU-FM broadcasts in?

A: WRSU-FM was licensed to broadcast to New Brunswick. Rivière-Verte, New Brunswick shares border with Edmundston. So the answer is: Edmundston.

{knowledge}

Q: Answer the following question by reasoning step-by-step. What genre is the record label of the performer of So Long, See You Tomorrow associated with?

A: The performer of So Long, See You Tomorrow is Bombay Bicycle Club. The record label of Bombay Bicycle Club is Island Records. The genre of Island Records is jazz. So the answer is: jazz.

{knowledge}

Q: Answer the following question by reasoning step-by-step. What is the genre of the record label of the band that performed on the Crush Tour?

A: The Crush Tour is performed by the band Bon Jovi. The record label of Bon Jovi is Island Records. The genre of Island Records is jazz. So the answer is: jazz.

{knowledge}

Q: Answer the following question by reasoning step-by-step. How many countries in Pacific National University's continent are recognized by the organization that mediated the truce ending the Iran-Iraq war?

A: Pacific National University is located in Khabarovsk, Russia Khabarovsk, Russian is in the continent of Asia. The entity that mediated the truce which ended the Iran-Iraq War is the UN. The number of member states that UN recognises in Asia is 53. So the answer is: 53.

Table 8: 7-Shot Demonstrations for IRCOT on MuSiQue.

You will be given a reasoning task with passage(s), a question, gold answer(s), and generated answer from model.

Your task is to evaluate the generated answer as either 0 or 1 based on the following criteria.

Consider the passages when making your evaluation.

You must answer the evaluation form using json format.

Evaluation Criteria:

1. Reasoning Initialization: Evaluate how well the generated answer starts the reasoning path based on the given passages and question. Does the first sentence provide a logical and relevant foundation for the rest of the reasoning? Consider the following:

- If the first reasoning step provides a necessary foundation for expanding the reasoning, evaluate it positively.

- If the first reasoning path is irrelevant or diverges from addressing the question directly, evaluate it negatively regardless of whether the answer is correct or incorrect.

2. Reasoning Expansion: Assess how well the generated answer extracts and applies relevant information from the passages to address the question. Does each subsequent sentence logically expand upon the first sentence to develop the reasoning effectively? Consider the following:

- If the model correctly extracts key information and logically expands upon it to support the reasoning, evaluate positively.

- If relevant information exists in the passages but is ignored or misused, evaluate negatively.

3. Reasoning Aggregation: Assess the alignment between the reasoning path and the final answer. Does the reasoning path logically lead to the final answer and ensure its correctness based on the provided reasoning? Consider the following:

- If both the reasoning path and the final answer are logically consistent, correct, and directly address the question, evaluate it positively.

- If the reasoning path contains correct intermediate steps but the final answer is logically inconsistent or incorrect, evaluate it negatively.

- If the reasoning path is incorrect but the final answer happens to be correct, also evaluate it negatively.

Evaluation Form:

- Reasoning Initialization: {{0 / 1}}

- Reasoning Expansion: {{0 / 1}}

- Reasoning Aggregation: {{0 / 1}}

Question:

{question}

Gold Answer List:

{gold_answer_list}

Passages:

{passage}

Generated Answer:

{generated_answer}

Table 9: GPT evaluation prompt for assessing reasoning abilities