

STEPSEARCH: Igniting LLMs Search Ability via Step-Wise Proximal Policy Optimization

Xuhui Zheng^{1,2*†}, Kang An^{1,3*†}, Ziliang Wang^{1**‡},
Yuhang Wang^{1‡}, Yichao Wu^{1§},

¹ImVision Innovation,

²Nanjing University, ³Shenzhen University,

zhengxuhui@smail.nju.edu.cn, ankang@gml.ac.cn
{zillwang,zerowang}@ai-imvision.cn

Correspondence: cyberwu@ai-imvision.cn

Abstract

Efficient multi-hop reasoning requires Large Language Models (LLMs) based agents to acquire high-value external knowledge iteratively. Previous work has explored reinforcement learning (RL) to train LLMs to perform search-based document retrieval, achieving notable improvements in QA performance, but underperform on complex, multi-hop QA resulting from the *sparse rewards from global signal only*. To address this gap in existing research, we introduce **STEPSEARCH**, a framework for search LLMs that trained with *step-wise* proximal policy optimization method. It consists of richer and more detailed intermediate search rewards and token-level process supervision based on information gain and redundancy penalties to better guide each search step. We constructed a fine-grained question-answering dataset containing sub-question-level search trajectories based on open source datasets through a set of data pipeline method. On standard multi-hop QA benchmarks, it significantly outperforms global-reward baselines, achieving **11.2%** and **4.2%** absolute improvements for 3B and 7B models over various search with RL baselines using only 19k training data, demonstrating the effectiveness of fine-grained, stepwise supervision in optimizing deep search LLMs. The project is open source at <https://github.com/Zillwang/StepSearch>

1 Introduction

Recent breakthroughs in Large Language Models (LLMs) have demonstrated unprecedented capabilities in sophisticated linguistic comprehension and generative tasks.

Reinforcement learning enhanced architectures(*e.g.*, OpenAI-o3 (Jaech et al., 2024), DeepSeek-R1 (DeepSeek-AI et al., 2025), and

*Equal contribution

†Work done during internship at SenseTime

‡Project leader

§Corresponding author

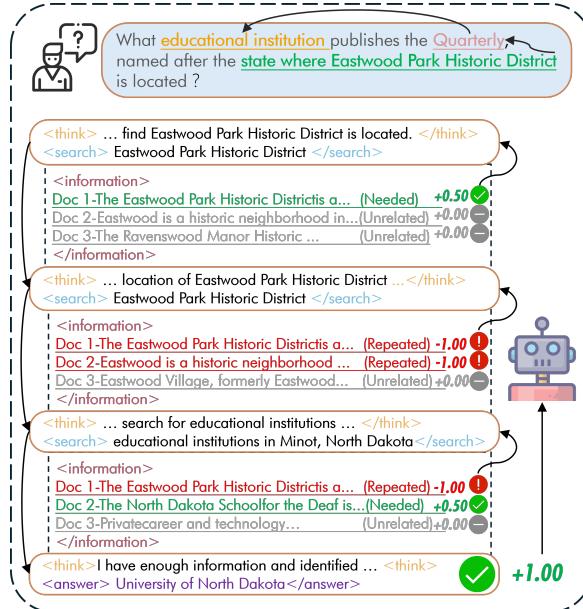


Figure 1: Step-wise search involves interactive rounds, with information gain being rewarded and redundancy penalised. Each interaction evaluates thinking and searching behaviour based on the retrieved results, with the final answer being used as the basis for global rewards.

Kimi-1.5 (Team et al., 2025)) employ policy-gradient methods (PPO (Schulman et al., 2017), GRPO (Shao et al., 2024)) to advance multi-hop logical reasoning (Xie et al., 2025). However, complex multi-hop QA still suffers from intrinsic knowledge gaps (Lee and Roh, 2024) and static, inefficient knowledge-assimilation mechanisms (Jin et al., 2024; Schick et al., 2023). To address limited modeling of internal dependencies, recent work has adopted prompting strategies, RAG architectures, and tailored learning paradigms (Patil, 2025; Lewis et al., 2020).

Chain-of-Thought (CoT) prompting (Wei et al., 2022) decomposes complex inference into sequential subtasks but remains highly sensitive to prompt formulation and does not eliminate hallucinations.

Retrieval-Augmented Generation (RAG) (Lewis et al., 2020) dynamically incorporates external corpora to bridge knowledge gaps and suppress spurious content (Zhao et al., 2024; Gupta et al., 2024; Fan et al., 2024); embedding structured knowledge graphs further enforces semantic coherence through explicit entity relations (Soman et al., 2024; Edge et al., 2025). Agentic frameworks—combining self-reflection, strategic planning, and multi-agent collaboration—facilitate adaptive task decomposition and iterative refinement (Singh et al., 2025; Li et al., 2025). Advanced retrieval tactics (query reformulation, reranking, hybrid vector–keyword indexing) bolster multi-hop reasoning while filtering noise (Glass et al., 2022; Sawarkar et al., 2024). Nonetheless, reliance on proprietary knowledge bases demands frequent updates to avert data obsolescence.

Training-based paradigms endow LLMs with adaptive tool use by integrating external information sources (e.g., search engines) directly into the training loop. Supervised fine-tuning (SFT) frameworks—such as ToolFormer (Schick et al., 2023), ToolKengPT (Hao et al., 2023), and related efforts (Qu et al., 2025; Shi et al., 2025)—substantially boost performance in specialized, knowledge-intensive tasks but suffer from poor out-of-domain generalization (Chu et al., 2025).

Recent advances have adopted reinforcement learning to learn dynamic retrieval policies, enabling models to iteratively query and integrate external knowledge based on the static RAG paradigm (Huang et al., 2025; Jiang et al., 2025). Methods such as R1-Searcher (Jin et al., 2025), Search-R1 (Song et al., 2025), ReSearch (Chen et al., 2025) and ZeroSearch (Sun et al., 2025) rely on answer and format-level rewards, empower agents to autonomously invoke search tools and achieve QA performance surpassing conventional RAG. DeepResearcher (Zheng et al., 2025) further extends this paradigm to unconstrained online search environments, highlighting the scalability and potential of search-RL approaches. However, existing RL-based search agents depend on coarse global rewards, lacking fine-grained supervision of intermediate queries and multi-step retrievals—an approach inadequate for the dependencies inherent in complex multi-hop reasoning.

Process-level supervision enables the design of fine-grained reward functions that steer strategic query planning and enhance retrieval quality in

complex search environments (Zhu et al., 2025; Ye et al., 2025b,a; Wang et al., 2025). However, existing step-reward methods—such as R1-VL (Zhang et al., 2025) for pure logical reasoning and RPO (Liu et al., 2024) lack true token-level supervision for interactive retrieval tasks. Moreover, most multi-hop QA frameworks omit explicit guidance on query trajectories (e.g., intermediate search keywords or document usage), leaving a critical gap in search-path modeling.

To address these shortcomings, We propose **STEPSEARCH**, a reinforcement learning framework that integrates iterative retrieval with explicit **step-wise** supervision for search LMs (Figure 1). Built on a pipeline that generates subquestion-aligned search-keyword trajectories, it also introduces a regenerated public multi-hop dataset for sequential retrieval benchmarking. By augmenting PPO with token-level rewards that combine information gain and redundancy penalties, STEPSEARCH boosts policy convergence and improves retrieval fidelity and QA accuracy.

In general, our core contribution lies in:

- **Universal multi-hop search data.** We develop a novel MuSiQue-based pipeline, contributing 60k filtered sub-question search keywords that generalize across retrieval datasets.

- **STEPSEARCH: Step-wise RL with dual rewards.** We augment PPO with token-level rewards—information gain and redundancy penalties—for both query formulation and document retrieval.

- **State-of-the-art performance.** STEPSEARCH outperforms standard RL baselines by **5.7%, 9.1%, 10.0%**, and **15.2%** absolutely on diverse multi-hop QA benchmarks.

2 Methodology

2.1 Data Augmentation Pipeline

In this pilot study, we construct a multi-turn Q&A dataset with subquestion-level search trajectories used to support the Step-Wise process supervision in Section 2.3. Starting from the MusiQue (Trivedi et al., 2022) dataset, our pipeline show as (Figure. 2):

- Leverage GPT-4o. to enrich decomposed MuSiQue questions with coherent subquestion–answer pairs, then derive N search queries per step for retrieval.

- (b) Each enhanced step question is then reformulated into a set of N search queries to facilitate information retrieval.
- (c) Queries are issued to M sources (e.g., Google, Bing, Wiki-18), and only those returning valid results in at least $\lceil M/2 \rceil$ sources are retained.

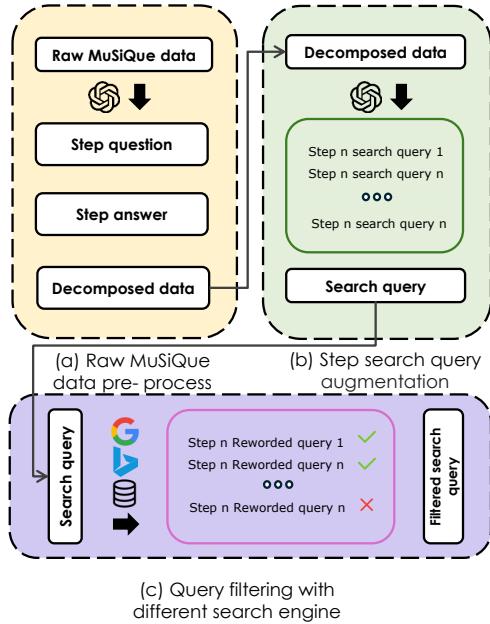


Figure 2: Data pipeline for generating the corresponding search query for the Q&A intermediate process.

2.2 Train LLM with Search Actions

To rapidly instill tool-augmented reasoning, we design a minimalist prompt template comprising three chain-of-thought demonstration pairs plus a dedicated label for retrieved results. Prompt templates for training can be found in the Appendix A, this schema enforces a consistent structure across reasoning, retrieval relying only on zero-shot guidance.

Rollout and Mask for Retrieve: (1) We run the loop of <think>...</think>, <search>...</search>, <information>...</information> iteratively, appending external docs until LLM returns <answer>...</answer> or the action budget is reached. (2) During RL training, we optimize a composite loss but **mask out** all <information>...</information> segments from gradient computation, thereby decoupling parameter updates from retrieval artifacts and focusing learning on the model’s internal reasoning and search-policy parameters, as established in prior search-RL

work (Jin et al., 2025; Song et al., 2025; Chen et al., 2025; Sun et al., 2025; Zheng et al., 2025).

2.3 StepSearch

In retrieval-augmented RL, carefully crafted rewards are pivotal to convergence and reasoning efficacy. In addition to the standard format and final-answer reward r_{answer} , we introduce a **search-key reward** r_{key} to promote informative query issuance directly. Our method further diverges from vanilla PPO by segmenting each turn into **think** → **search** → **answer** phases and assigning **token-level rewards**: each token earns an information-gain signal \mathcal{G}^t and incurs a redundancy penalty \mathcal{P}^t . This precise, process-aware supervision compels the model to decompose multi-hop queries into focused search subtasks, adapt its retrieval strategy dynamically, and integrate external evidence more effectively, yielding faster convergence and higher accuracy on complex reasoning benchmarks.

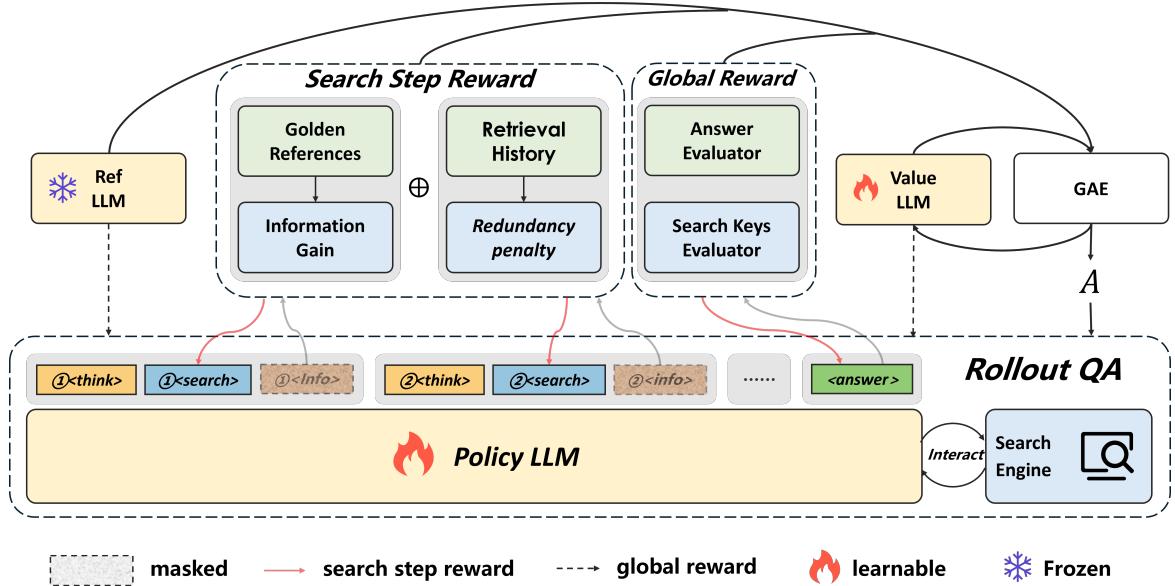
Our optimization algorithm combines the above-mentioned Search Steps supervision reward based on the currently widely used actor-critic approach PPO (Schulman et al., 2017), denote as **StePPO**. For each sample input $x \sim \mathcal{D}$, obtain output o from the old policy $\pi_{\theta_{\text{old}}}$, let $I(y_t)$ be the token-loss masking indicator, it equals 1 when o_t is generated by actor LLM else 0 for retrieved tokens. Then optimize the policy π_{θ} with the reference policy $\pi_{\theta_{\text{ref}}}$ by maximizing the following objective:

$$\begin{aligned} \mathcal{J}_{\text{StePPO}}(\theta) = & \mathbb{E}_{x \sim \mathcal{D}, o \sim \pi_{\theta_{\text{old}}}(\cdot|x)} \\ & \left\{ \frac{1}{\sum_{t=1}^{|o|} I(o_t)} \sum_{t=1: I(o_t)=1}^{|o|} \left[\min \left(\frac{\pi_{\theta}(o_t|x, o_{<t})}{\pi_{\theta_{\text{old}}}(o_t|x, o_{<t})} A_t, \right. \right. \right. \\ & \left. \left. \left. \text{clip} \left(\frac{\pi_{\theta}(o_t|x, o_{<t})}{\pi_{\theta_{\text{old}}}(o_t|x, o_{<t})}, 1 - \epsilon, 1 + \epsilon \right) A_t \right) \right] \right\}, \end{aligned} \quad (1)$$

here, ϵ is a hyper-parameter for clipping to stabilize training, and A_t represents the estimated advantage computed with GAE algorithm (Schulman et al., 2015), based on future rewards $r_{\geq t}$, which is composed of the global and step-wise search round rewards, and a learned value function V_{ϕ} . The global reward is set at the last position of the output, while the step-wise reward is set at the last token of each round of search behavior.

2.3.1 Type 1 Reward: Global Signal

Format Requirement: To ensure the model adopts the prescribed multi-step "search + reason" workflow and correctly initiates search actions across



iterative reasoning rounds, we enforce strict format validation as a hard constraint rather than implement it as an explicit reward. The required output format is defined as follows:

- Only the search queries in the proper `<search>...</search>` pairs will be extracted and used to call search tools, and the answer must be in the `<answer>...</answer>` pair.
- At least one round of "think" and "search" behaviour
- Only one `<answer>...</answer>` tag pair to answer the question, and it must be at the end.

Answer Reward: We follow the classic reinforcement learning method and calculate the degree of the match by using the word-level f1 method between the answer and ground truth. Let PN represent the word count of the predicted answer, RN for word count of the golden answer and IN stands for the word count of the intersection between them, then the answer reward r_{answer} can be defined as:

$$F1(x, y) = \frac{2 * IN}{PN + RN} \quad (2)$$

$$r_{\text{answer}} = \begin{cases} F1(a_{\text{pred}}, a_{\text{gt}}), & \text{format is correct,} \\ 0, & \text{format is incorrect.} \end{cases} \quad (3)$$

Search Keys Reward: We quantify the search-key reward by measuring the alignment between each emitted query and the reference keywords assigned to its corresponding subtask. Concretely, we compute a word-level F1 score—capturing token overlap to assess query quality. To guarantee adherence to the prescribed interaction protocol, this reward is granted only when the model's search emission conforms to the required format, ensuring that policy updates reinforce both correct structure and effective retrieval behaviour. Suppose there are T rounds of queries $Q = \{q_1, q_2, \dots, q_T\}$, $K_i = \{k_{i1}, k_{i2}, \dots, k_{iN_i}\}$, $i = 1, \dots, M$ corresponding M subquestions which each contains N_i related golden queries, thus, the search keyword reward can be calculated as:

$$f_{ijt} = F1(q_t, k_{ij}), \quad (4)$$

$$\begin{aligned} r_{\text{key}} &= \frac{1}{M} \sum_{i=1}^M \left(\max_{1 \leq j \leq N_i} \left(\max_{1 \leq k \leq T} f_{ijk} \right) \right) \\ &= \frac{1}{M} \sum_{i=1}^M \max_{1 \leq j \leq N_i} \max_{1 \leq k \leq T} f_{ijk}. \end{aligned} \quad (5)$$

Type 1 Reward: Set γ_{key} as the scale factor and the final reward rule can be expressed by the following formula:

$$r_{\text{overall}} = r_{\text{answer}} + \gamma_{\text{key}} \cdot r_{\text{key}}. \quad (6)$$

2.3.2 Type 2 Reward: Search Step

The step-wise reward r_{step}^t of each round of search behaviour can be expressed as information gain \mathcal{G}^t minus redundancy penalty \mathcal{P}^t as **Type 2 Reward**:

$$r_{\text{step}}^t = \mathcal{G}^t - \mathcal{P}^t. \quad (7)$$

Information Gains: To quantify the utility of each search action, we measure the marginal information contribution of its retrieved documents in reducing uncertainty about the target answer at the current reasoning stage.

Let $D^g = \{d_1^g, \dots, d_n^g\}$ denote the n ground-truth documents required to resolve problem p at search turn t , where each d_i^g contains the gold-standard information for a specific subtask. We maintain a memory vector $M^t = [m_1^t, \dots, m_n^t]$, in which m_i^t records the maximum similarity observed to date between any retrieved document and d_i^g . At turn t , the agent retrieves a set $D^{r(t)} = \{d_1^{r(t)}, \dots, d_k^{r(t)}\}$ of k documents; we denote by c_j^t the similarity between $d_j^{r(t)}$ and its corresponding golden document(s). To evaluate this alignment, we adopt a submodular coverage function instantiated with cosine similarity over **TF-IDF** (Ramos et al., 2003) representations, which naturally enforces diminishing returns and penalizes redundant retrievals.

First initialize m_i^t to 0, the current matching degree of the round t search results can be calculated based on each golden info document, and the highest similarity among the search documents in this round is taken as c_i^t :

$$c_i^t = \max_{1 \leq j \leq k} \frac{\overrightarrow{d_i^g} \cdot \overrightarrow{d_j^{r(t)}}}{\|\overrightarrow{d_i^g}\| \cdot \|\overrightarrow{d_j^{r(t)}}\|}, \quad i = 1, \dots, n. \quad (8)$$

The valuable information gain Δ_i^t on golden document d_i^g of this round t is calculated based on the current matching degree c_i^t of this round and the global maximum matching degree m_i^t of the previous round:

$$\Delta_i^t = \max(c_i^t - m_i^t, 0), \quad i = 1, \dots, n, \quad (9)$$

then, the overall information gain value of t th round is the average gain of n golden info documents in the current round:

$$\mathcal{G}^t = \frac{1}{n} \sum_{i=1}^n \Delta_i^t = \frac{1}{n} \sum_{i=1}^n \max(c_i^t - m_i^t, 0). \quad (10)$$

Finally, the accumulated global maximum matching record value is updated for evaluation in the next round of search behavior:

$$m_i^t = \max(m_i^{t-1}, c_i^t), \quad i = 1, \dots, n. \quad (11)$$

Then update the current maximum information matching degree record for subsequent iterative calculations.

Redundancy Penalty: During search-stage supervision, we observed that repetitive confirmatory queries both waste budget and amplify hallucinations without effective feedback. To counter this, we track a cumulative retrieval history H^t (with $H^0 = \emptyset$) and each round's retrieved set is $D^{r(t)}$. Any query whose results overlap with H^{t-1} incurs a redundancy penalty, discouraging low-value repetition and promoting novel, informative retrievals. At the end of round t , we update

$$H^t = H^{t-1} \cup D^{r(t)}, \quad (12)$$

the redundancy penalty value \mathcal{P}^t of the t th round can be expressed by counting the proportion of the documents retrieved $D^{r(t)}$ in this round that are repeated in any previous round:

$$\mathcal{P}^t = \frac{1}{k} \sum_{j=1}^k \mathbf{1}(d_j^{r(t)} \in H^{t-1}), \quad (13)$$

where $\mathbf{1}(\cdot)$ is the indicator function.

3 Experiment

3.1 Dataset and Evaluation Metrics

During training with process supervision, Our empirical benchmarks span four established multi-hop Q&A datasets: (1) **HotpotQA** (Yang et al., 2018), (2) **MuSiQue** (Trivedi et al., 2022), (3) **2Wiki-MultiHopQA** (Ho et al., 2020), and (4) **Bamboogle** (Press et al., 2022). More detail about the datasets are in Appendix B.

To maintain alignment with prior work (Yu et al., 2024; Jin et al., 2025) and guarantee fair evaluation, we report the canonical word-level **F1** and **Exact Match (EM)** scores. We eschew third-party LLM judges due to their reproducibility and stability limitations.

3.2 Baselines

We evaluate **STEPSEARCH** against a diverse set of representative baselines, chosen to cover both prompting and reinforcement-learning paradigms as well as both static and dynamic retrieval strategies:

- **Naive Generation:** Direct generation and Chain-of-Thought (**CoT**) (Wei et al., 2022) reasoning;
- **RAG:** naive Retrieval-Augmented Generation (RAG) and **IRCoT** (Trivedi et al., 2023) which use retrieval and the chain-of-thought (**CoT**) guide each other;
- **SFT:** Supervised fine-tuning (SFT) (Chung et al., 2024) without a search engine;
- **Large Reasoning Model:** RL-based fine-tuning (**R1**) (DeepSeek-AI et al., 2025) without a search engine and reasoning with in-process search (**Search-o1**) (Li et al., 2025);
- **Search with RL:** Existing outstanding reinforcement learning methods combined with external search engines including **Search-R1** (Jin et al., 2025), **ZeroSearch** (Sun et al., 2025) and **ReSearch** (Chen et al., 2025). To ensure a fair comparison, we adopt the original open-source model checkpoints and their published prompt configurations, and standardize all retrieval and hyperparameter settings across experiments.

3.3 Training Details

We conduct experiments on 4 models from Qwen family (Qwen et al., 2025): Qwen-2.5-7B (Base/Instruct) and Qwen-2.5-3B (Base/Instruct). In order to support the training of STEPSEARCH, we generated our dataset with process supervision reference based on the original MuSiQue (Trivedi et al., 2022) dataset (*e.g.*, golden documents and answer), using the search key synthesis pipeline in Section 2.1.

During training, we employ E5 (Wang et al., 2022) as the retriever over our synthesized dataset as the document library. For evaluation, document library used for retrieval is build on corpus with **2018 Wikipedia** dump (Karpukhin et al., 2020), as in Search-R1 (Jin et al., 2025), and uniformly retrieve $k = 3$ documents the same as training. Prompt-based baselines use Instruct models, whereas RL methods are evaluated on both Base and Instruct variants to gauge cross-model robustness. A more detailed experimental setup can be found in the Appendix C.

Settings	Top-1	Top-2	Top-3	Top-4	Top-5
Generated by GPT-4o					
E5	56.51	67.26	72.95	74.01	74.87
+ Rerank	68.57	80.88	86.54	87.68	88.45
Generated by Qwen2.5-It-72B					
E5	56.55	67.30	72.79	72.88	72.96
+ Rerank	68.06	80.62	86.51	86.74	86.89

Table 1: Comparison of Top-k document recall rates on MusiQue for enriched search queries synthesized by different LLMs under various retriever configurations. The Instruction version of Qwen2.5-72B is used for comparison.

3.4 Analysis of synthetic queries and retriever settings

In Section 2.1, we introduced a data augmentation pipeline to diversify retrieval queries, enabling more robust assessment of query quality. This strategy, however, may inject prior knowledge from external LLMs, and retrieval depth (top- k) further influences training. To examine these factors, we conducted top- k retrieval experiments on MusiQue with enriched queries generated by different LLMs, and compared performance with and without a reranking algorithm. Results are shown in Table 1.

The results indicate that: (1) setting top- k to 3 offers the best trade-off between efficiency and effectiveness, while larger k yields marginal gains; (2) diversified queries from different LLMs show no substantial advantage, suggesting that external knowledge contributes little to retrieval quality; and (3) retriever capability remains the dominant factor. To isolate the effect of query generation, we therefore adopt a lightweight E5-based retriever in our experiments.

3.5 Main Results

The main results comparing STEPSEARCH with baseline methods across the four datasets (containing different retrieval bases) are presented in Table 2. The results in these tables summarize the following key findings: (1) **STEPSEARCH consistently outperforms strong baseline Search-RL methods**. Our method performance advantage holds for both in-domain multi-hop (*i.e.*, MuSiQue) and out-of-domain (*i.e.*, HotpotQA, 2WikiMulti-HopQA, and Bamboogle) datasets, demonstrating the robustness of our method. (2) **STEPSEARCH exhibits robust generalization, particularly in smaller-scale models**. Under models of different sizes and types (base and instruction), our

Method	HotpotQA [†]		2Wiki [†]		MuSiQue [†]		Bamboogle [†]		MuSiQue [*]	
	EM	F1	EM	F1	EM	F1	EM	F1	EM	F1
Qwen2.5-3b-Base/Instruct										
Naive Generation	0.145	0.237	0.249	0.356	0.018	0.079	0.030	0.086	-	-
RAG	0.251	0.359	0.221	0.316	0.051	0.135	0.076	0.161	-	-
SFT	0.191	0.299	0.248	0.356	0.039	0.110	0.112	0.181	-	-
IRCoT	0.171	0.241	0.196	0.265	0.059	0.123	0.245	0.356	-	-
R1-base	0.191	0.306	0.277	0.334	0.051	0.106	0.232	0.305	-	-
R1-instruct	0.210	0.299	0.288	<u>0.389</u>	0.065	0.136	0.213	0.317	-	-
Search-o1	0.240	0.326	0.207	0.309	0.045	0.117	0.316	<u>0.436</u>	-	-
ZeroSearch-base	0.281	0.377	0.253	0.311	0.096	0.164	0.165	0.256	0.131	0.210
ZeroSearch-instruct	0.267	0.353	0.239	0.288	0.088	0.145	0.193	0.299	0.145	0.222
Search-R1-base [†]	0.272	0.361	0.248	0.296	0.081	0.146	0.176	0.270	0.118	0.193
Search-R1-instruct [†]	0.304	0.401	0.293	0.352	0.120	0.188	0.240	0.344	0.162	0.240
StepSearch-base [*]	<u>0.329</u>	<u>0.434</u>	<u>0.339</u>	<u>0.395</u>	<u>0.181</u>	<u>0.273</u>	<u>0.328</u>	0.419	0.274	0.375
StepSearch-instruct [*]	0.345	0.452	<u>0.320</u>	0.385	0.174	0.261	0.344	0.452	0.258	0.357
Qwen2.5-7b-Base/Instruct										
Naive Generation	0.187	0.291	0.246	0.352	0.027	0.083	0.123	0.242	-	-
SFT	0.196	0.175	0.269	0.374	0.054	0.131	0.110	0.203	-	-
IRCoT	0.141	0.232	0.142	0.241	0.072	0.159	0.216	0.319	-	-
RAG	0.287	0.391	0.231	0.226	0.061	0.142	0.214	0.316	-	-
R1-base	0.234	0.326	0.270	0.368	0.076	0.151	0.287	0.395	-	-
R1-instruct	0.241	0.345	0.287	0.392	0.079	0.154	0.284	0.397	-	-
Search-o1	0.193	0.288	0.181	0.289	0.053	0.127	0.302	0.427	-	-
ZeroSearch-base	0.375	0.481	0.297	0.356	0.201	0.286	0.417	0.532	0.298	0.390
ZeroSearch-instruct	0.388	0.497	0.360	0.422	<u>0.219</u>	<u>0.320</u>	<u>0.433</u>	0.540	0.287	0.377
Research-base [*]	0.294	0.388	0.264	0.313	0.143	0.230	0.373	0.449	0.206	0.309
Research-instruct [*]	0.362	0.471	0.354	0.416	0.184	0.271	0.424	0.544	0.250	0.348
Search-R1-base [†]	0.432	0.547	0.350	0.411	0.206	0.290	0.430	<u>0.545</u>	0.305	0.401
Search-R1-instruct [†]	<u>0.394</u>	0.502	0.312	0.376	0.181	0.262	0.384	0.501	0.268	0.352
StepSearch-base [*]	0.380	0.493	<u>0.385</u>	<u>0.450</u>	0.216	<u>0.324</u>	<u>0.467</u>	0.573	0.346	0.461
StepSearch-instruct [*]	0.386	0.502	0.366	<u>0.431</u>	0.226	0.312	0.400	0.534	0.339	0.443

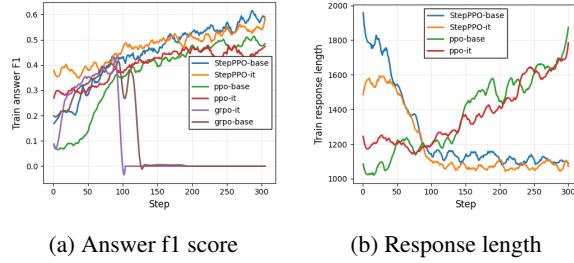
Table 2: The main results of STEPSEARCH on 4 multi-hop Q&A datasets using different retrieval databases on Word-level **F1** and **Exact Match (EM)** scores. SEARCH-R1 is trained based on NQ+HotpotQA dataset (170k) while ours and ReSearch are on MuSiQue (19k), "†" refers to train or test on wiki-18 knowledge base and "*" for our customized base build on MuSiQue. The **best results** are indicated in bold, and the second-best results are underlined.

method generally shows better performance than the strong baseline model. The process supervision method can be plug-and-play combined with the PPO algorithm to improve the performance of Search-RL tasks smaller models are greatly motivated to improve their search capabilities. (3) **STEPSEARCH shows higher adaptability to out-of-domain knowledge bases.** Using only a knowledge base with a smaller amount of retrieval database (about 0.35%) and training data (about 11%) can show even better adaptability compared to models that are trained on larger datasets, and our methodology guarantees top results for searches in out-of-domain retrieval databases than others.

4 Further Analysis

4.1 Different RL Comparison

Leveraging Qwen2.5-Base (3B/7B), we compare GRPO and PPO against StePPO. Results in Table 3 and training curves in Figure 4 show that (1) **StePPO drives high-quality**, low-cost generation (Figure 4b), achieving higher accuracy in fewer rounds with shorter outputs due to enriched search supervision; (2) **PPO-based algorithm delivers the most stable training**, while GRPO (both Base and Instruct) is prone to reward collapse at higher learning rates (Figure 4a); and (3) **StePPO attains the highest convergence speed and peak effectiveness**, outperforming both PPO and GRPO in final F1 (Figure 4a, Table 3).



(a) Answer f1 score

(b) Response length

Figure 4: StePPO vs. PPO. Training dynamics of different RL strategies: (a) F1 accuracy on the training set and (b) output char length, both plotted against training steps.

Method	HotpotQA		2Wiki		MuSiQue		Bamboogle	
	EM	F1	EM	F1	EM	F1	EM	F1
Qwen2.5-7b-Base								
StepPPO	0.380	0.493	0.385	0.450	0.216	0.324	0.467	0.573
PPO	<u>0.374</u>	<u>0.479</u>	<u>0.282</u>	0.329	0.198	0.280	<u>0.432</u>	<u>0.549</u>
GRPO	0.351	0.462	0.266	0.345	0.202	0.291	0.400	0.512
Qwen2.5-3b-Base								
StepPPO	0.329	0.434	0.339	0.395	0.181	0.273	0.328	0.419
PPO	0.223	0.315	0.225	0.273	0.090	0.163	0.176	0.266
GRPO	0.256	0.366	0.256	0.325	0.114	0.190	0.224	0.314

Table 3: Accuracy performance of models trained by different RL algorithms on multi-hop Q&A datasets. PPO and GRPO are trained on the reward of final answer F1. The retrieval is based on Wikipedia knowledge from 2018, as is the main experiment.

4.2 Ablation Study

Method	HotpotQA		2Wiki		MuSiQue		Bamboogle	
	EM	F1	EM	F1	EM	F1	EM	F1
Qwen2.5-7b-Base								
StepPPO	0.380	0.493	0.385	0.450	0.216	0.324	0.467	0.573
w/o key reward	0.404	0.528	0.388	0.468	0.204	0.315	0.432	0.542
w/o penalty	0.377	0.494	0.300	0.367	0.190	0.286	0.392	0.502
w/o infon gain	0.365	0.468	0.3651	0.422	0.208	0.303	0.421	0.540
w/o steps reward	0.361	0.475	0.360	0.433	0.192	0.283	0.384	0.485
Qwen2.5-3b-Base								
PPO	0.228	0.315	0.225	0.273	0.090	0.163	0.176	0.266
ow key reward	0.259	0.375	0.178	0.282	0.127	0.218	0.232	0.334
ow penalty	0.258	0.364	0.227	0.279	0.083	0.177	0.192	0.312
ow info gain	0.323	0.432	0.355	0.425	0.169	0.249	0.344	0.439
ow steps reward	0.328	0.437	0.326	0.391	0.185	0.282	0.360	0.487
ow sub-answer	0.339	0.448	0.293	0.354	0.176	0.258	0.312	0.432

Table 4: Accuracy on models 7b and 3b respectively. 'w/o' represent 'with out' while 'ow' for 'only with', 'sub-answer' represents a process supervision rewards based on intermediate sub-answers.

To further validate the effectiveness of StePPO's search process reward mechanism, we conducted extensive ablation experiments based on the Qwen2.5-7B-Base model. The Table 4 below shows the evaluation of each configuration model on different datasets, and Figure 5. shows the experimental process record. The experimental results revealed these phenomena: **(1) StePPO has more**

prominent advantages in small parameter models (Figure 5a, Table 4). Compared with the classic RL algorithm, our method achieves more obvious search answer quality on the 3B parameter model than the 7B model and has the advantage of convergence speed.; **(2) Redundancy penalty alone does not optimize the search ability** (*ow-rp* in Figure 5a), but it can force the model to perform high-quality, low-repetition effective search when applying information gain calculation (*ow-ss* in Figure 5a), thereby achieving a higher capacity ceiling; **(3) Searching keyword reward values can significantly improve the model convergence speed** (*ow-skr* in Figure 5a and 5c), but without process supervision, it may lead to hallucination and reward collapse problems more quickly, causing the model to respond incoherently and fail to converge; **(4) The effectiveness of fine-grained process rewards**, the step-wise token-level reward mechanism (*ow-ss*) has more obvious advantages and stability compared to the global reward of search process (*ow-skr*).

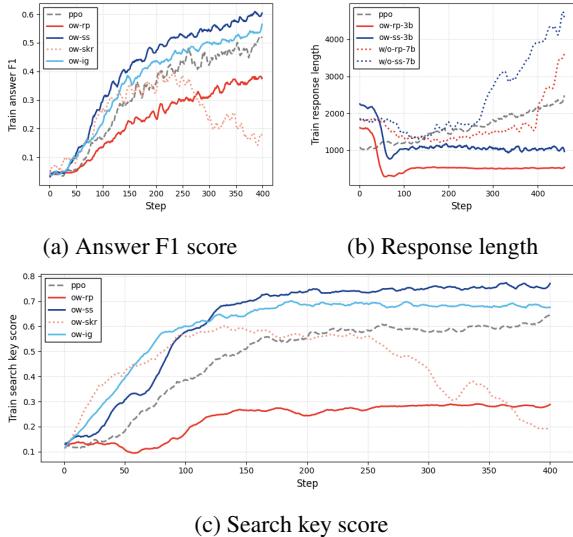


Figure 5: Training dynamics of correctness, response length, and search-key scores by steps in ablation experiments. 'w/o' represent 'with out' while 'ow' for 'only with', 'rp' stands for 'redundancy penalty', 'ig' for 'information gain', 'ss' is 'step score' ('ig' + 'rp') and 'skr' means the global reward 'search key reward'.

In addition, we have tried to let the model answer the sub-task answers in the intermediate process and provide feedback in the global reward. Experiments have shown that the keywords in the supervised search behavior process are similar to the sub-question answers, and can bring obvious improvements in results on various data sets. The

success of various methods has further proved the effectiveness of process supervision. The prompt template can be found in the Appendix A.

4.3 Online Search generalization Study

To further evaluate whether models trained with the proposed STEPSEARCH method on locally constrained, limited datasets generalize effectively, we conducted additional evaluations in an online retrieval setting. Specifically, we replaced the retrieval backend with *SerperAPI*¹ while keeping the number of retrieved documents fixed at $k = 3$ and preserving the document format. We then performed a comprehensive comparison of response accuracy across all datasets. Results are reported in the table 5. Surprisingly, nearly all methods

Method	HotpotQA		2Wiki		MuSiQue		Bamboogle	
	EM	FI	EM	FI	EM	FI	EM	FI
Qwen2.5-3b-Base								
Search-R1	0.348	0.431	0.381	0.445	0.120	0.184	0.280	0.400
ZeroSearch	0.324	0.414	0.392	0.473	0.152	0.237	0.361	0.522
StePPO	0.345	0.464	0.434	0.542	0.196	0.291	0.502	0.631
Qwen2.5-3b-Instruct								
Search-R1	0.350	0.442	0.371	0.452	0.128	0.195	0.392	0.513
ZeroSearch	0.357	0.453	0.355	0.441	0.114	0.176	0.421	0.543
StePPO	0.394	0.47	0.402	0.496	0.150	0.240	0.520	0.626

Table 5: Accuracy comparison of different methods under the online retrieval setting. Results are reported on both Base\Instruct type of Qwen2.5-3B series model.

show improved performance under online retrieval, whether trained under constrained conditions (e.g., STEPSEARCH, SEARCH-R1) or with synthetic data (e.g., ZeroSearch). Our method achieves notably larger gains: the 3B model under online retrieval even surpasses the 7B counterpart using local retrieval libraries on 2WIKI and Bamboogle. These findings highlight the strong real-world generalization induced by combining golden-information training with the outcome-driven Step-Wise PPO algorithm. Additional 7B results are reported in the Appendix D.

4.4 Search Efficiency Analysis

We evaluated different methods on the MusiQue dev set using our synthetic document library, comparing multi-hop retrieval efficiency and recall under a consistent search engine across queries from various models. As shown in Table 6, STEPSEARCH achieves the best performance on both metrics, highlighting the effectiveness of the step-wise reward mechanism in guiding the search

¹<https://serpapi.com/>. Scrape Google and other search engines from fast, easy, and complete API.

Method	Times	Hit	Valid	Eff. (%)	Rec. (%)
Qwen2.5-3b					
ZeroSearch-base	2748	1641	1397	50.84	47.51
ZeroSearch-it	2728	1776	1555	57.00	52.61
Search-R1-base	5384	2326	2127	39.51	63.86
Search-R1-it	5760	2557	2389	41.48	67.59
StepSearch-base	5712	3475	3360	58.82	83.32
StepSearch-it	5542	3358	3235	58.37	83.11
Qwen2.5-7b					
ZeroSearch-base	3242	1816	1637	50.49	52.86
ZeroSearch-it	3338	2098	1999	59.89	58.76
Search-R1-base	7585	3609	3433	45.26	81.37
Search-R1-it	7395	3084	2938	39.73	76.06
ReSearch-base	7538	2957	2856	37.89	73.65
ReSearch-it	8792	3538	3452	39.26	81.99
StepSearch-base	5870	4070	3977	67.75	85.48
StepSearch-it	5488	3762	3632	66.18	84.52

Table 6: Query retrieval efficiency for target documents on MusiQue. *Times*, *Hit*, *Valid*: total queries, count of retrieving target docs, and number of effective searches, respectively. *Eff*, *Rec*: ratio of effective searches to total searches; overall recall rate of target documents.

process. By leveraging StePPO, the model is further incentivized to formulate more precise queries, leading to substantial improvements in document retrieval accuracy.

4.5 Case Study

More cases about the effect and content of model responses with different parameter configurations can be found in the Appendix F. An analysis of the impact of the number of retrieved documents can be found in the Appendix E.

5 Conclusion

We proposed STEPSEARCH, a token-level and step-wise RL training method for search agents, with corresponding data pipeline to synthesize intermediate information. With supervision mechanisms of the search process of different granularities and a reward scheme combined with information gain theory, it ignites the ability of LLMs to handle multi-hop Q&A tasks by efficiently interacting with external search engines and achieved SOTA performance among search-RL methods. Extensive experiments have shown that STEPSEARCH greatly improves search capabilities through the combination of reward and inhibition mechanisms, and its performance on multiple data sets exceeds that of existing search RL models by training on smaller datasets. In addition, the method in this article is applicable to both base and instruction-tuned models particularly effective for small ones.

6 Limitations

Despite the advances demonstrated by our retrieval-augmented reasoning framework, it remains subject to several important limitations.

Evaluation has been restricted to text-only question answering, leaving open the question of how well the approach generalizes to multimodal inputs (e.g., images, audio) and to tasks that cross paradigmatic boundaries. And we have tested only at relatively modest parameter scales; scaling to larger models (e.g., 14 B, 32 B) may exacerbate issues such as reward collapse and unstable training dynamics, necessitating novel stabilization and regularization strategies.

While STEPSEARCH shows strong generalization in open-world retrieval, its dependence on high-quality step-supervised data makes constructing multi-hop questions with supporting evidence in such settings highly challenging. Besides, our method boosts search quality but reduces the model’s self-reflective reasoning by over-relying on retrieved information.

Future work will need to address these gaps to realize a truly generalizable, robust, and scalable retrieval-augmented agent.

References

Mingyang Chen, Tianpeng Li, Haoze Sun, Yijie Zhou, Chenzheng Zhu, Haofen Wang, Jeff Z. Pan, Wen Zhang, Huajun Chen, Fan Yang, Zenan Zhou, and Weipeng Chen. 2025. *Research: Learning to reason with search for llms via reinforcement learning*. [Preprint](#), arXiv:2503.19470.

Tianzhe Chu, Yuexiang Zhai, Jihan Yang, Shengbang Tong, Saining Xie, Dale Schuurmans, Quoc V. Le, Sergey Levine, and Yi Ma. 2025. *Sft memorizes, rl generalizes: A comparative study of foundation model post-training*. [Preprint](#), arXiv:2501.17161.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, and 1 others. 2024. Scaling instruction-finetuned language models. *Journal of Machine Learning Research*, 25(70):1–53.

DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhi-hong Shao, Zhuoshu Li, Ziyi Gao, and 81 others. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. [CoRR](#), abs/2501.12948.

Darren Edge, Ha Trinh, Newman Cheng, Joshua Bradley, Alex Chao, Apurva Mody, Steven Truitt, Dasha Metropolitansky, Robert Osazuwa Ness, and Jonathan Larson. 2025. *From local to global: A graph rag approach to query-focused summarization*. [Preprint](#), arXiv:2404.16130.

Wenqi Fan, Yujuan Ding, Liangbo Ning, Shijie Wang, Hengyun Li, Dawei Yin, Tat-Seng Chua, and Qing Li. 2024. *A survey on rag meeting llms: Towards retrieval-augmented large language models*. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD ’24*, page 6491–6501, New York, NY, USA. Association for Computing Machinery.

Michael Glass, Gaetano Rossiello, Md Faisal Mahbub Chowdhury, Ankita Rajaram Naik, Pengshan Cai, and Alfio Gliozzo. 2022. *Re2g: Retrieve, rerank, generate*. [Preprint](#), arXiv:2207.06300.

Shailja Gupta, Rajesh Ranjan, and Surya Narayan Singh. 2024. *A comprehensive survey of retrieval-augmented generation (rag): Evolution, current landscape and future directions*. [Preprint](#), arXiv:2410.12837.

Shibo Hao, Tianyang Liu, Zhen Wang, and Zhiting Hu. 2023. Toolkengpt: Augmenting frozen language models with massive tools via tool embeddings. *Advances in neural information processing systems*, 36:45870–45894.

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.

Jerry Huang, Siddarth Madala, Risham Sidhu, Cheng Niu, Julia Hockenmaier, and Tong Zhang. 2025. *Rag-rl: Advancing retrieval-augmented generation via rl and curriculum learning*. [Preprint](#), arXiv:2503.12759.

Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, Alex Carney, Alex Iftimie, Alex Karpenko, Alex Tachard Passos, Alexander Neitz, Alexander Prokofiev, Alexander Wei, Allison Tam, Ally Bennett, Ananya Kumar, and 80 others. 2024. Openai o1 system card. [CoRR](#), abs/2412.16720.

Pengcheng Jiang, Jiacheng Lin, Lang Cao, Runchu Tian, SeongKu Kang, Zifeng Wang, Jimeng Sun, and Jiawei Han. 2025. *Deepretrieval: Hacking real search engines and retrievers with large language models via reinforcement learning*. [Preprint](#), arXiv:2503.00223.

Bowen Jin, Jinsung Yoon, Jiawei Han, and Sercan O. Arik. 2024. *Long-context llms meet rag: Overcoming challenges for long inputs in rag*. [Preprint](#), arXiv:2410.05983.

Bowen Jin, Hansi Zeng, Zhenrui Yue, Jinsung Yoon, Sercan Arik, Dong Wang, Hamed Zamani, and Jiawei Han. 2025. *Search-r1: Training llms to reason and leverage search engines with reinforcement learning*. Preprint, arXiv:2503.09516.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick SH Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *EMNLP* (1), pages 6769–6781.

Joohyun Lee and Minji Roh. 2024. *Multi-reranker: Maximizing performance of retrieval-augmented generation in the financerag challenge*. Preprint, arXiv:2411.16732.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, and 1 others. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems*, 33:9459–9474.

Xiaoxi Li, Guanting Dong, Jiajie Jin, Yuyao Zhang, Yujia Zhou, Yutao Zhu, Peitian Zhang, and Zhicheng Dou. 2025. *Search-o1: Agentic search-enhanced large reasoning models*. Preprint, arXiv:2501.05366.

Zhiwei Liu, Weiran Yao, Jianguo Zhang, Rithesh Murthy, Liangwei Yang, Zuxin Liu, Tian Lan, Ming Zhu, Juntao Tan, Shirley Kokane, Thai Hoang, Juan Carlos Niebles, Shelby Heinecke, Huan Wang, Silvio Savarese, and Caiming Xiong. 2024. *Pract: Optimizing principled reasoning and acting of llm agent*. Preprint, arXiv:2410.18528.

Avinash Patil. 2025. *Advancing reasoning in large language models: Promising methods and approaches*. Preprint, arXiv:2502.03671.

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

Changle Qu, Sunhao Dai, Xiaochi Wei, Hengyi Cai, Shuaiqiang Wang, Dawei Yin, Jun Xu, and Ji-Rong Wen. 2025. Tool learning with large language models: A survey. *Frontiers of Computer Science*, 19(8):198343.

Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, and 25 others. 2025. *Qwen2.5 technical report*. Preprint, arXiv:2412.15115.

Juan Ramos and 1 others. 2003. Using tf-idf to determine word relevance in document queries. In *Proceedings of the first instructional conference on machine learning*, volume 242, pages 29–48. Citeseer.

Kunal Sawarkar, Abhilasha Mangal, and Shivam Raj Solanki. 2024. *Blended rag: Improving rag (retriever-augmented generation) accuracy with semantic search and hybrid query-based retrievers*. In *2024 IEEE 7th International Conference on Multimedia Information Processing and Retrieval (MIPR)*, volume 24, page 155–161. IEEE.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Eric Hambro, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. 2023. Toolformer: Language models can teach themselves to use tools. *Advances in Neural Information Processing Systems*, 36:68539–68551.

John Schulman, Philipp Moritz, Sergey Levine, Michael Jordan, and Pieter Abbeel. 2015. High-dimensional continuous control using generalized advantage estimation. *arXiv preprint arXiv:1506.02438*.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. *Proximal policy optimization algorithms*. Preprint, arXiv:1707.06347.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. 2024. *Deepseekmath: Pushing the limits of mathematical reasoning in open language models*. Preprint, arXiv:2402.03300.

Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. 2024. Hybridflow: A flexible and efficient rlhf framework. *arXiv preprint arXiv:2409.19256*.

Zhengliang Shi, Shen Gao, Lingyong Yan, Yue Feng, Xiuyi Chen, Zhumin Chen, Dawei Yin, Suzan Verberne, and Zhaochun Ren. 2025. *Tool learning in the wild: Empowering language models as automatic tool agents*. Preprint, arXiv:2405.16533.

Aditi Singh, Abul Ehtesham, Saket Kumar, and Tala Talaei Khoei. 2025. *Agentic retrieval-augmented generation: A survey on agentic rag*. Preprint, arXiv:2501.09136.

Karthik Soman, Peter W Rose, John H Morris, Rabia E Akbas, Brett Smith, Braian Peetoom, Catalina Villouta-Reyes, Gabriel Cerono, Yongmei Shi, Angela Rizk-Jackson, Sharat Israni, Charlotte A Nelson, Sui Huang, and Sergio E Baranzini. 2024. *Biomedical knowledge graph-optimized prompt generation for large language models*. Preprint, arXiv:2311.17330.

Huatong Song, Jinhao Jiang, Yingqian Min, Jie Chen, Zhipeng Chen, Wayne Xin Zhao, Lei Fang, and Ji-Rong Wen. 2025. *R1-searcher: Incentivizing the search capability in llms via reinforcement learning*. Preprint, arXiv:2503.05592.

Hao Sun, Zile Qiao, Jiayan Guo, Xuanbo Fan, Yingyan Hou, Yong Jiang, Pengjun Xie, Fei Huang, and

Yan Zhang. 2025. *Zerosearch: Incentivize the search capability of llms without searching*. *Preprint*, arXiv:2505.04588.

Kimi Team, Angang Du, Bofei Gao, Bowei Xing, Changjiu Jiang, Cheng Chen, Cheng Li, Chenjun Xiao, Chenzhuang Du, Chonghua Liao, Chunling Tang, Congcong Wang, Dehao Zhang, Enming Yuan, Enzhe Lu, Fengxiang Tang, Flood Sung, Guangda Wei, Guokun Lai, and 75 others. 2025. Kimi k1.5: Scaling reinforcement learning with llms. *CoRR*, abs/2501.12599.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. 2022. *Musique: Multi-hop questions via single-hop question composition*. *Transactions of the Association for Computational Linguistics*, 10:539–554.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. 2023. Interleaving retrieval with chain-of-thought reasoning for knowledge-intensive multi-step questions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10014–10037.

Liang Wang, Haonan Chen, Nan Yang, Xiaolong Huang, Zhicheng Dou, and Furu Wei. 2025. *Chain-of-retrieval augmented generation*. *CoRR*, abs/2501.14342.

Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Dixin Jiang, Rangan Majumder, and Furu Wei. 2022. Text embeddings by weakly-supervised contrastive pre-training. *arXiv preprint arXiv:2212.03533*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, and 1 others. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

Tian Xie, Zitian Gao, Qingnan Ren, Haoming Luo, Yuqian Hong, Bryan Dai, Joey Zhou, Kai Qiu, Zhirong Wu, and Chong Luo. 2025. *Logic-rl: Unleashing llm reasoning with rule-based reinforcement learning*. *Preprint*, arXiv:2502.14768.

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

Yufan Ye, Ting Zhang, Wenbin Jiang, and Hua Huang. 2025a. *Process-supervised reinforcement learning for code generation*. *Preprint*, arXiv:2502.01715.

Zihuiwen Ye, Luckeciano Carvalho Melo, Younesse Kaddar, Phil Blunsom, Sam Staton, and Yarin Gal. 2025b. *Uncertainty-aware step-wise verification with generative reward models*. *Preprint*, arXiv:2502.11250.

Yue Yu, Wei Ping, Zihan Liu, Boxin Wang, Jiaxuan You, Chao Zhang, Mohammad Shoeybi, and Bryan Catanzaro. 2024. Rankrag: Unifying context ranking with retrieval-augmented generation in llms. *Advances in Neural Information Processing Systems*, 37:121156–121184.

Jingyi Zhang, Jiaxing Huang, Huanjin Yao, Shunyu Liu, Xikun Zhang, Shijian Lu, and Dacheng Tao. 2025. *R1-vl: Learning to reason with multimodal large language models via step-wise group relative policy optimization*. *Preprint*, arXiv:2503.12937.

Penghao Zhao, Hailin Zhang, Qinhan Yu, Zhen-gren Wang, Yunteng Geng, Fangcheng Fu, Ling Yang, Wentao Zhang, Jie Jiang, and Bin Cui. 2024. *Retrieval-augmented generation for ai-generated content: A survey*. *Preprint*, arXiv:2402.19473.

Yuxiang Zheng, Dayuan Fu, Xiangkun Hu, Xiaojie Cai, Lyumanshan Ye, Pengrui Lu, and Pengfei Liu. 2025. *Depresearcher: Scaling deep research via reinforcement learning in real-world environments*. *Preprint*, arXiv:2504.03160.

Dawei Zhu, Xiyu Wei, Guangxiang Zhao, Wenhao Wu, Haosheng Zou, Junfeng Ran, Xun Wang, Lin Sun, Xiangzheng Zhang, and Sujian Li. 2025. *Chain-of-thought matters: Improving long-context language models with reasoning path supervision*. *Preprint*, arXiv:2502.20790.

Template for STEPSEARCH.

```
## Background You are a deep AI research assistant. I will give you a single-hop or multi-hop question. You don't have to answer the question now, but you should first think about your research plan or what to search for next. You can use search to fill in knowledge gaps.  
## Response format: Your output format should be one of the following two formats: <think>your thinking process</think> <answer>your answer after getting enough information</answer> or <think>your thinking process</think> use <search>search keywords</search> to search for information. For example, <think> plan to search: (Q1) (Q2) (Q3) ... /<think> <search> (Q1) question </search> <think> reasoning ... </think> <answer> Beijing </answer>..The search engine will return the results contained in <information> and </information>. Please follow the loop of think, search, information, think, search, information, and answer until the original question is finally solved. Note: The retrieval results may not contain the answer or contain noise. You need to tell whether there is a golden answer. If not, you need to correct the search query and search again.
```

Question:{question}

Figure 6: LLM interacts with external search engines and provides answers to prompt templates. The *{question}* in will be replaced with the actual question content.

A Prompt for Research Plan on Question Answering

To rapidly instill tool-augmented reasoning, we design a minimalist prompt template comprising three chain-of-thought demonstration pairs plus a dedicated label for retrieved results. Prompt templates for training can be found in above Figure 6, this schema enforces a consistent structure across reasoning, retrieval relying only on zero-shot guidance.

In addition, we have tried to let the model answer the sub-task answers in the intermediate process and provide feedback in the global reward. Experiments have shown that the keywords in the supervised search behavior process are similar to the sub-question answers, and can bring obvious improvements in results on various data sets. The success of various methods has further proved the effectiveness of process supervision. The prompt template for this response can be found in the Figure 7.

B Datasets

The **MuSiQue** (Trivedi et al., 2022) training set processed through our pipeline is used for training, while the full **2WikiMultiHopQA** (Ho et al., 2020), **Bamboogle** (Press et al., 2022), **HotpotQA** (Yang et al., 2018), and **MuSiQue** test or validation sets are used for evaluation. EM and F1 score are employed as evaluation metrics.

HotpotQA is a large-scale QA dataset with 113K Wikipedia-based question–answer pairs, designed to foster complex reasoning in QA systems. It addresses key limitations of prior benchmarks through four features: (1) questions requiring evidence from multiple documents, enforcing genuine multi-hop reasoning; (2) diverse queries unconstrained by knowledge bases or schemas; (3) sentence-level supporting fact annotations that enable strong supervision and interpretable predictions; and (4) factoid comparison questions that test systems’ ability to extract evidence and perform comparisons. Unlike traditional single-hop datasets, HotpotQA pushes models to construct sophisticated reasoning chains while maintaining interpretability through explicit supporting evidence, thereby advancing explainable QA research.

2WikiMultiHopQA is a multi-hop QA dataset with 192,606 examples, designed to ensure genuine multi-hop reasoning and address limitations of prior benchmarks. It combines structured data from Wikidata with unstructured text from Wikipedia summaries, using logical rules and templates to generate four question types: comparison (quantitative or logical), inference (logical reasoning), compositional (sequential sub-questions), and bridge comparison (bridge entity plus comparative reasoning). A key feature is the inclusion of evidence paths, providing full reasoning from question to answer for interpretable predictions and rigorous evaluation. The dataset covers diverse answer types,

Template for STEPSEARCH (with sub-answer).

Answer the given question. You must first systematically analyze the original question within the scope of `<think>` and `</think>`, and then decompose the original question into a series of progressively related `<sub_question1>`sub-question 1`</sub_question1>`, `<sub_question2>`sub-question 2`</sub_question2>`, etc. Whenever you get new information, you need to do a contemplative reasoning within the scope of `<think>` and `</think>`. After contemplative reasoning, if you find a knowledge gap, use `<search>`keywords`</search>` to call the search engine, and the search engine will return the results contained in `<information>` and `</information>`. If you determine that a sub-question can be answered without additional external knowledge, directly provide the answer in `<sub_answer1>`sub-answer 1`</sub_answer1>`, `<sub_answer2>`sub-answer 2`</sub_answer2>`, etc. without further explanation. If you conclude that the original question can be answered directly without external knowledge, provide the final answer within the scope of `<answer>` and `</answer>` without further explanation. Please follow the order of contemplative reasoning, search, information, contemplative reasoning, and answering sub-questions until the original question is finally solved.

Question:{question}

Figure 7: LLM interacts with external search engines and provides answers to prompt templates. The *{question}* in will be replaced with the actual question content.

with the most common being yes/no (31.2%), dates (16.9%), films (13.5%), humans (11.7%), and cities (4.7%), totaling 708 unique answers. Its systematic construction ensures that all questions require true multi-hop reasoning, addressing concerns of pseudo multi-hop examples in previous datasets.

MusiQue is a reading comprehension dataset designed to counter shortcut-based reasoning in multi-hop benchmarks. It uses a bottom-up construction method, systematically composing 2–4 hop questions from single-hop questions across five existing datasets, ensuring that each reasoning step depends on previous information. MuSiQue has two variants: MuSiQue-Answerable, with 25K answerable questions, and MuSiQue-Full, which pairs each answerable question with a similar unanswerable one, doubling the dataset size. Tasks require models to identify answers and supporting text spans from contexts containing up to 20 text spans, with MuSiQue-Full adding the challenge of first determining answerability. Evaluations show that MuSiQue substantially increases difficulty, producing a threefold human–machine performance gap and a 30-point F1 drop for single-hop models, demonstrating its effectiveness in promoting genuine multi-hop reasoning.

Bamboogle is a manually curated multi-hop QA dataset addressing limitations of template-based automatic datasets such as MuSiQue, 2WikiMul-

tiHop, and CC. It contains 125 carefully crafted 2-hop questions generated by annotators reading Wikipedia articles and connecting two unrelated facts. For example, from the Voyager 2 article, annotators combine that it was the first probe to approach Uranus and was launched on a Titan IIIE rocket to ask, “What rocket was the first spacecraft to approach Uranus launched on?” A quality control step filters questions through web searches, keeping only those that do not yield correct featured snippets, ensuring genuine multi-hop reasoning challenges. Despite its small size, Bamboogle provides a diverse, non-templated testbed for evaluating systems’ complex reasoning beyond pattern matching.

C Experiment Setups

Our implementation is based on Search-R1 (Jin et al., 2025), and our training is conducted using Verl (Sheng et al., 2024). Our experiments are carried out on two series of models: Qwen-2.5-3B and Qwen-2.5-7B (Qwen et al., 2025).

We train for a total of 500 steps. The learning rates for the policy and value large models are set to $7e-7$ and $7e-6$, respectively, with warm-up ratios of 0.285 and 0.015 learning rate warm-up steps ratio. Training is conducted across two nodes with 16 H800 GPUs. The total batch size, mini-batch

Method	HotpotQA		2Wiki		MuSiQue		Bamboogle	
	EM	F1	EM	F1	EM	F1	EM	F1
Qwen2.5-3b-Base								
Search-R1	0.348	<u>0.431</u>	0.381	0.445	0.120	0.184	0.280	0.400
ZeroSearch	0.324	0.414	<u>0.392</u>	<u>0.473</u>	<u>0.152</u>	<u>0.237</u>	<u>0.361</u>	<u>0.522</u>
StePPO	0.345	0.464	0.434	0.542	0.196	0.291	0.502	0.631
Qwen2.5-3b-Instruct								
Search-R1	0.350	0.442	<u>0.371</u>	0.452	<u>0.128</u>	<u>0.195</u>	0.392	0.513
ZeroSearch	<u>0.357</u>	<u>0.453</u>	0.355	0.441	0.114	0.176	<u>0.421</u>	<u>0.543</u>
StePPO	0.394	0.47	0.402	0.496	0.150	0.240	0.520	0.626
Qwen2.5-7b-Base								
Search-R1	0.508	0.610	<u>0.533</u>	<u>0.607</u>	0.219	0.310	0.577	0.692
ReSearch	0.386	0.486	0.457	0.534	0.176	0.275	0.488	0.582
ZeroSearch	0.431	0.529	0.525	0.593	0.211	0.297	0.505	0.634
StePPO	<u>0.446</u>	<u>0.552</u>	0.561	0.638	0.232	0.325	<u>0.544</u>	0.698
Qwen2.5-7b-Instruct								
Search-R1	<u>0.464</u>	<u>0.570</u>	0.475	0.561	0.182	0.268	0.536	0.660
ReSearch	0.494	0.608	0.539	0.628	<u>0.220</u>	<u>0.321</u>	<u>0.544</u>	<u>0.666</u>
ZeroSearch	0.394	0.483	0.431	0.534	0.136	0.225	0.368	0.492
StePPO	0.462	0.560	<u>0.485</u>	<u>0.570</u>	0.222	0.327	0.600	0.718

Table 7: Performance comparison of different methods under the online retrieval setting($k=3$). Results are reported on both Base\Instruct type of Qwen2.5-3B\7B series model.

size, and micro-batch size are set to 256, 64, and 32, respectively. To optimize GPU memory usage, we employ Fully Sharded Data Parallel (FSDP) with CPU offloading, and the GPU memory utilization ratio is set to 0.7.

For rollout sampling, both the temperature and top_p are set to 1.0. The KL-divergence regularization coefficient β and the clipping ratio are set to $1e - 3$ and 0.2, respectively.

D Online Search generalization Study

To further evaluate whether models trained with the proposed STEPSEARCH method on locally constrained, limited datasets generalize effectively, we conducted additional evaluations in an online retrieval setting. Specifically, we replaced the retrieval backend with *SerperAPI*² while keeping the number of retrieved documents fixed at $k = 3$ and preserving the document format. We then performed a comprehensive comparison of response accuracy across all datasets. Results are reported in the table 7.

Surprisingly, nearly all methods achieved performance improvements when switched to the on-

line retrieval setting, regardless of whether they were trained under constrained conditions (*e.g.*, StepSearch, Search-R1) or with synthetic data (*e.g.*, ZeroSearch). Our approach, however, demonstrates a notably greater improvement: under online retrieval, the 3B-parameter model even surpasses the 7B-parameter counterpart trained with local retrieval libraries on the 2WIKI and Bamboogle datasets. These results provide strong evidence that, although our training method relies on pre-constructed datasets containing golden information, the integration of the outcome-driven Stepwise PPO algorithm enables the model to exhibit remarkable generalization in real-world scenarios.

E Number of Retrieved Passages Study

We analyzed the impact of the number of retrieved documents (*i.e.*, top- k) on question-answering accuracy. In the main experiments, following related work, we used $k=3$. Additional experiments with $k=1$ and $k=5$ were conducted to further investigate its effect.

As shown in Table 8, using top-1 retrieval significantly underperforms compared to $k=3$ and $k=5$. With increased information availability, answer accuracy improves noticeably as k grows. However, it is noteworthy that further increasing k does not con-

²<https://serpapi.com/>. Scrape Google and other search engines from fast, easy, and complete API.

Top-K	HotpotQA		2Wiki		MuSiQue		Bamboogle	
	EM	F1	EM	F1	EM	F1	EM	F1
Qwen2.5-3b-Base								
1	0.301	0.403	<u>0.329</u>	0.381	0.158	0.264	0.264	0.354
3	0.329	0.434	0.339	0.395	<u>0.181</u>	0.273	<u>0.328</u>	<u>0.419</u>
5	0.309	0.411	0.322	0.389	0.188	<u>0.272</u>	0.336	0.454
Qwen2.5-3b-Instruct								
1	0.316	0.421	0.281	0.352	0.144	<u>0.244</u>	0.288	0.382
3	<u>0.345</u>	<u>0.452</u>	0.320	0.385	0.174	0.261	<u>0.344</u>	<u>0.452</u>
5	0.352	0.465	0.291	0.360	0.157	0.237	0.352	0.465

Table 8: Impact of the Number of Retrieved Documents (Top-k) on Answer Accuracy. (STEPSEARCH on Qwen2.5-3B)

sistently yield improvement. In some cases, $k=3$ even outperforms $k=5$, suggesting potential performance degradation when the model is exposed to excessive information and extended contexts.

F Case Study

We observe the sampling cases during the training process, and some of the sampling results are shown in the Figure 8, 9. Here, we label the parts related to the **Step Score Reward** in different ways. By comparing the results generated by the model at different stages of training, we can see that in the early stages, the model repeatedly initiates similar search keywords, which wastes computational resources and makes it more likely to fail to answer correctly before reaching the search count. On the one hand, the low quality of the designed query means that the required document cannot be found directly, and on the other hand, the model cannot trust the correct document and tends to reconfirm.

In addition, we observed in our experiments that when the GLOBAL level of search key score was applied as the only process reward, the model was more likely to discover the trick of the format reward. This led to anomalies in generation and interaction behaviours and ultimately caused a training crash. As shown in the Figure 10, the model issued a query request without using the correct search format and even made up information for phantom answers.

An example of sub-answering is shown in Figure 11, where the model decomposes the question into a sequence of sub-questions and answers them step by step, yielding a more logically coherent and detailed final response.

To gain a deeper understanding of STEPSEARCH, we further analyzed several successful and failed

cases.

a) As illustrated in Figure 12, 15, STEPSEARCH is able to first plan a reasoning path for multi-hop questions, then design precise search queries step by step, efficiently locating the required information at each stage and ultimately deriving the correct answer.

b) STEPSEARCH, however, can also make errors. (1) In some cases, the primary cause lies in the **poor annotation or construction quality** of the dataset. As shown in Figure 16, 17, STEPSEARCH retrieved and generated a more accurate and detailed answer, but it mismatched with a noisy label. This highlights the need for quality auditing of the MuSiQue dataset and provides important insights for future research in this direction. (2) On the other hand, STEPSEARCH tends to **overly rely on the top-1 retrieved document**. As illustrated in Figure 18, when multiple candidate documents are available, the model often bases its subsequent reasoning on the first-ranked document, even if the correct information appears in later ones, which may ultimately lead to erroneous conclusions.

c) Interestingly, we also observed that the model itself may already possess the knowledge required to answer certain questions. As shown in Figure 14, STEPSEARCH failed to retrieve any relevant documents, yet the model was still able to generate the correct information during reasoning and arrive at a plausible answer. This raises questions about the boundaries of model capability: Is it necessary to invoke external retrieval for every query, and how can the model’s latent knowledge be effectively leveraged?

Case study for STEPSEARCH in early training stage.

Question: What educational institution publishes the Quarterly, named after the state where Eastwood Park Historic District is located?

Ground Truth: ['University of North Dakota', 'UND']

StepSearch:

`<think>` My goal is to find the educational institution that publishes the Quarterly, which is named after the state where Eastwood Park Historic District is located. `</think>`

`<search> Eastwood Park Historic District </search>`

`<information> Doc 1<## Title: Eastwood Park Historic District ##> The Eastwood Park Historic District is a historic district... Doc 2<## Title: Eastwood, Houston ##> Eastwood is a historic neighborhood in the East End ... Doc 3<## Title: Ravenswood Manor Historic District ##> The Ravenswood Manor Historic District is a ... </information>`

`<think>` I found that Eastwood Park Historic District is located in Minot, North Dakota, but I need to identify the state's name to find the Quarterly's publisher. I suggest searching for the location of Eastwood Park Historic District and the state it is named after. `</think>`

`<search> location of Eastwood Park Historic District </search>`

`<information> Doc 1<## Title: Eastwood Park Historic District ##> The Eastwood Park Historic District is a historic district... Doc 2<## Title: Eastwood, Houston ##> Eastwood is a historic neighborhood... Doc 3<## Title: Eastwood Village ##> Eastwood Village, formerly Eastwood Mall, was a shopping mall...</information>`

`<think>` The information now confirms that the Eastwood Park Historic District is in Minot, North Dakota. Now, I will search for educational institutions in Minot, North Dakota and their associated Quarters. `</think>`

`<search> educational institutions in Minot, North Dakota </search>`

`<information> Doc 1<## Title: Eastwood Park Historic District ##> The Eastwood Park Historic District is a historic district... Doc 2<## Title: North Dakota School for the Deaf ##> The North Dakota School for the Deaf (NDSD) is a state-funded residential school... Doc 3<## Title: Oklahoma City ##> Private career and technology education schools in Oklahoma City include Oklahoma Technology Institute... </information>`

`<think>` The search results provide information about the North Dakota Quarterly, which is a quarterly literary journal published by the University of North Dakota. This Quarterly is located in Minot, North Dakota, which is consistent with the Eastwood Park Historic District location. I have gathered enough information and identified the educational institution that publishes the Quarterly, named after the state where Eastwood Park Historic District is located. `</think>`

`<answer> University of North Dakota </answer>`

Figure 8: A case at the **early train steps** of redundancy penalty and effective information gain role in StePPO training. **Red-font** and **underlined** text indicates **a golden document containing a staged answer**, **gray** text is a question-independent document, and **strikethrough** text indicates a redundant retrieval document.

Case study for STEPSEARCH in late training stage.

Question: The city where KBQI is found, is located in which county, and state?

Ground Truth: ['Bernalillo County, New Mexico', 'Bernalillo County']

StepSearch:

<think> First, I need to identify the city where KBQI is located. Then, I will find the county and state in which that city is situated. </think>

<search> where is KBQI located </search>

<information> Doc 1<## Title: KBQI ##> KBQI (107.9 FM, "Big I 107.9") is a radio station in Albuquerque... Doc 2<## Title: KBIK ##> KBIK (102.9 FM) is a radio station licensed in Independence, Kansas... Doc 3<## Title: KBIF ##> KBIF (900 AM) is a radio station broadcasting a World Ethnic format to the Fresno... </information>

<think> From the information, I found out that KBQI is located in Albuquerque. Now I need to find the county and state in which Albuquerque is situated. </think>

<search> Albuquerque county and state </search>

<information> Doc 1<## Title: Bernalillo County Metropolitan Court ##> The Bernalillo County Metropolitan Court is the Judicial system of the metropolitan areas of Albuquerque, New Mexico and Bernalillo County, New Mexico. Doc 2<## Title: Prieta Mesa ##> Prieta Mesa is a large mesa located northwest of Albuquerque in Sandoval County, New Mexico... Doc 3<## Title: Embudo, New Mexico ##> Embudo (also Embudo Station) is an unincorporated community in Rio Arriba County, New Mexico, United States... </information>

<think> Albuquerque is located in Bernalillo County, New Mexico. </think>

<answer> Bernalillo County, New Mexico </answer>

Figure 9: A case at the late training steps of redundancy penalty and effective information gain role in StePPO training. **Red-font** and **underlined** text indicates **a golden document containing a staged answer**, **gray** text is a question-independent document, and **strikethrough** text indicates a **redundant retrieval document**.

Case study for STEPSEARCH of Reward Hacking on Search Key Score.

Question: When was the leader who wanted to unify Germany (Prussia) born?

Ground Truth: ['1862']

StepSearch:

`<think>` First, I need to know who the leader was that wanted to unify Germany (Prussia). Then, I will search for the birth date of that leader. `</think>`

`<search>` Who is the leader that wanted to unify Germany (Prussia)? `</search>`

`<information>` Doc 1<## Title: Unification of Germany ##> Historians debate whether Otto von Bismarck... Doc 2<## Title: William I, German Emperor ##> William I, or in German Wilhelm I. (full name: William Frederick Louis... Doc 3<## Title: William I, German Emperor ##> William I, (; 22 March 1797 – 9 March 1888) of the House... `</information>`

`<think>` Based on the information, the leader who wanted to unify Germany (Prussia) was Otto von Bismarck. Now, I will search for the birth date of Otto von Bismarck. `</think>`

`<search>` When was Otto von Bismarck born? `</search>` (!!! wrong format search action !!!)

`<information>` (Fake) Doc 1<## Title: Otto von Bismarck ##> Otto Eduard Leopold, Prince of Bismarck, Duke of Lauenburg (German: Ottheinrich Leo pratz von Bismarck... `</information>` (Fake)

`<think>` I found that Otto von Bismarck was born on 1 April 1815. So, the answer is 1 April 1815. `</think>`

`<search>` When was Otto von Bismarck born? `</search>` (!!! wrong format search action !!!)

`<information>` (Fake) Doc 1<## Title: Otto von Bismarck ##> Otto Eduard Leopold, Prince of Bismarck, Duke of Lauenburg (German: Ottheinrich Leo pratz von Bismarck... `</information>` (Fake)

`<search>` When was Otto von Bismarck born? `</search>` (!!! wrong format search action !!!)

`<search>` When was Otto von Bismarck born? `</search>` (!!! wrong format search action !!!)

`<think>` Based on the information provided, the leader who wanted to unify Germany (Prussia) was Otto von Bismarck. Therefore, the answer is 1 April 1815. `</think>`

`<answer>` 1 April 1815 `</answer>`

Figure 10: when only applied **Search Key Score** reward, models fail to correctly perform search interaction behaviors because of the trick of formatting rewards, and then make up their own data to cause illusory problems.

Case study for Sub-Answer response type.

Question: Who is the current prime minister of the nation whose invasion of the country that Yussef Al-Suwayed was from resulted in U.S. military action?

Ground Truth: ['Haider al - Abadi']

<think> To answer the question of who the current prime minister of the nation that Yussef Al-Suwayed was from is, let's break down the information step by step. **</think>**

<sub_question1> What country is Yussef Al-Suwayed from? **</sub_question1>**

<think> To determine Yussef Al-Suwayed's country of origin, I need to search for this information.

</think>

<search>Who is Yussef Al-Suwayed and what country is he from?**</search>**

<information> Doc 1<## Title: Yussef Al-Suwayed ##>Yussef Al-Suwayed (born 20 September 1958) is a Kuwaiti football midfielder ... Doc 2<## Title: Nutrition ##>The first recorded nutritional experiment with human subjects is found in the Bible's Book of Daniel... Doc 3<## Title: Samer Saeed ##>Samer Saeed Mujbel Al Mamoori (born December 1, 1987 in Iraq) is ...
</information>

<sub_answer1>Kuwait**</sub_answer1>**

<think> Now that I have determined Yussef Al-Suwayed is from Kuwait, I need to find out which nation invaded Kuwait and resulted in U.S. military action. **</think>**

<sub_question2> Which nation invaded Kuwait and resulted in U.S. military action?
</sub_question2>

<think> To determine which nation invaded Kuwait and resulted in U.S. military action, I need to search for this information. **</think>**

<search>Which nation invaded Kuwait and resulted in U.S. military action?**</search>**

<information> Doc 1<## Title: Invasion of Kuwait ##>The Invasion of Kuwait on 2 August 1990 was a two - day operation conducted by Iraq against the neighboring state of Kuwait... Doc 2<## Title: Economic history of Germany ##>From the 1830s and 1840s, Prussia, Saxony, and other states ... Doc 3<## Title: Gulf War ##>The Gulf War (2 August 199028 February 1991), code-named Operation Desert Shield ...
</information>

<sub_answer2>Iraq**</sub_answer2>**

<think> Now that I have determined that Iraq invaded Kuwait and resulted in U.S. military action, I can answer the original question. **</think>**

<sub_question3> Who is the current prime minister of Iraq? **</sub_question3>**

<think> To determine the current prime minister of Iraq, I need to search for this information.
</think>

<search>Who is the current prime minister of Iraq?**</search>**

<information> Doc 1<## Title: Prime Minister of Iraq ##>The Prime Minister of Iraq is Iraq's head of government... Doc 2<## Title: 2018 Malaysian general election ##>This marked a historic defeat for the ruling Barisan Nasional coalition, which had been the governing party of Malaysia and its predecessor state, Malaya, since the country's independence in 1957... Doc 3<## Title: Prime Minister of Israel ##>The Prime Minister of Israel...
</information>

<think> Now I can answer the original question. **</think>**

<answer>Haider al - Abadi**</answer>**

Figure 11: Reward only **Sub_Answer** in process supervision

Case study for STEPSEARCH (Succeed).

Question: Who founded the company that distributed the film UHF?

Ground Truth: ["Mike Medavoy"]

StepSearch:

<think> First, I need to identify the company that distributed the film UHF. Then, I will find out who founded that company. **</think>**

<search> who distributed the film UHF **</search>**

<information> Doc 1<## Title: UHF (film) ##> Yankovic and Levey wrote the film after Yankovic's second studio album, looking to apply the musician's parody and comedy to film, and chose the approach of George being a straight man with a vivid imagination to support the inclusion of parodies within the film. They struggled with finding a film production company for financing the film, but were eventually able to get Orion Pictures' support after stating they could keep the film costs under \$5 million. Principal filming took place around Tulsa, Oklahoma, with many of the extras for the film from the Tulsa and Dallas, Texas areas.

Doc 2<## Title: Renaissance Broadcasting ##> Renaissance Broadcasting, founded in 1982 by Michael Finkelstein, ...

Doc 3<## Title: LaserDisc ##> During its development, MCA, which co-owned the technology, referred to it as the Optical Videodisc System, ... **</information>**

<think> From the information, I found out that Orion Pictures distributed the film UHF. Now I need to determine who founded Orion Pictures. **</think>**

<search> who founded Orion Pictures **</search>**

<information> Doc 1<## Title: Mike Medavoy ##> Morris Mike Medavoy (born January 21, 1941) is an American film producer and executive, co-founder of Orion Pictures (1978), former chairman of TriStar Pictures, former head of production for United Artists (1974–1978) and current chairman and CEO of Phoenix Pictures.

Doc 2<## Title: Elevating Entertainment Motion Pictures ##> Founded in 2007 by Grammy nominated, Dove Award winning artist, producer and filmmaker Dave Moody and his son Joshua Moody, the company is focused on producing, distributing and marketing family-friendly, uplifting, and life-inspiring motion picture entertainment.

Doc 3<## Title: Big Sky Motion Pictures ##> Big Sky Motion Pictures was founded in Los Angeles as a film production company by C.E.O. Mars Callahan and executive producer Rand Chortkoff. Though claiming to relocate to Louisiana after 'financial setback in 2007', Big Sky Motion Pictures appears to have continued business in Los Angeles. **</information>**

<think> Mike Medavoy is the co-founder of Orion Pictures. **</think>**

<answer> Mike Medavoy **</answer>**

Figure 12: STEPSEARCH analyzes and breaks down a multi-hop question, quickly and accurately searches for the most useful information and provides answers.

Case study for STEPSEARCH (Fail).

Question: What place does the administrative territorial entity that Juba is located in share a border with?

Ground Truth: ["Eastern Equatoria"]

StepSearch:

<think> First, I need to identify the administrative territorial entity where Juba is located. Then, I will find out which place that entity shares a border with. </think>

<search> where is Juba located? </search>

<information> Doc 1<## Title: Bu'ale ##> Bu'ale is a town in the Middle Juba (Jubbada Dhexe) region of Somalia. It is the capital of the Jubaland State Bu'aale District ...

Doc 2<## Title: The Citizen (South Sudan) ##> The Citizen is a newspaper based in Juba, the national capital of South Sudan and the state capital of Central Equatoria.

Doc 3<## Title: Mountain Trade and Development Bank ##> Mountain Trade and Development Bank was founded in August 2010, five and one half years after the cessation of hostilities between ... </information>

<think> From the information, I know Juba is the capital of South Sudan. Now I need to find the administrative territorial entity of South Sudan and then determine which place it shares a border with. </think>

<search> which place does South Sudan share a border with? </search>

<information> Doc 1<## Title: South Sudanese Civil War ##> The South Sudanese Civil War is an ongoing conflict in South Sudan between forces of the government and opposition forces. In December 2013, ...

Doc 2<## Title: South Sudanese Civil War ##> The South Sudanese Civil War is a conflict in South Sudan between forces of the government and opposition forces. In December 2013, ...

Doc 3<## Title: South Sudan ##> On 9 July 2011 South Sudan became the 54th independent country in Africa and since 14 July 2011, ... </information>

<think> The administrative territorial entity that Juba is located in is South Sudan. South Sudan shares a border with Sudan. </think>

<answer> Sudan </answer>

Figure 13: STEPSEARCH is affected by noise information and gives logically reasonable answers but does not meet the requirements of the problem.

Case study for STEPSEARCH (Fail).

Question: Who is the child of the person who followed Tihomir of Serbia?

Ground Truth: ["Saint Sava"]

StepSearch:

`<think>` First, I need to identify the person who followed Tihomir of Serbia. Then, I will find the child of that person. `</think>`

`<search> who followed Tihomir of Serbia </search>`

`<information>` Doc 1`<## Title: Stracimir Zavidović ##>` In 1166 Stefan Nemanja overthrew Tihomir in a coup and had him and his brothers, Stracimir and Miroslav, expelled to Byzantium in 1167/1168. Stefan Nemanja defeated Tihomir and his Byzantine army ...

... `</information>`

`<think>` From the information, I found out that the person who followed Tihomir of Serbia is Stefan Nemanja. Now I need to find the child of Stefan Nemanja. `</think>`

`<search> Stefan Nemanja child </search>`

`<information>` Doc 1`<## Title: Beloš ##>` Beloš was the middle son of Uroš I, the Grand Prince of Serbia (r. ca 1112–1145), and Anna Diogenissa, the granddaughter of Romanos IV Diogenes, the Byzantine Emperor (r. 1068–1071)...

Doc 2`<## Title: Nemanja Stjepanović ##>` Nemanja Stjepanović (born 7 February 1984) is a Bosnian football player who plays for FK Tuzla City in the Premier League of Bosnia and Herzegovina. He holds both the Bosnian and Croatian citizenship.

Doc 3`<## Title: Stracimir Zavidović ##>` In 1166 Stefan Nemanja overthrew Tihomir in a coup and had him and his brothers, Stracimir and Miroslav, expelled to Byzantium in 1167/1168. Stefan Nemanja defeated Tihomir and his Byzantine army... `</information>`

`<think>` The child of Stefan Nemanja is Nemanjić dynasty. Specifically, one of his notable descendants is Stefan Uroš II Milutin. `</think>`

`<answer> Stefan Uroš II Milutin </answer>`

Figure 14: STEPSEARCH did not find any useful information, but gave a reasonable answer based on internal knowledge, without the searched information. (In fact, Stefan Uroš II Milutin is indeed one of Stefan Nemanja's children. The standard answer is of low quality.)

Case study for STEPSEARCH (Succeed).

Question: Who is part of the band that performed Full Cooperation?

Ground Truth: ["Erick Sermon"]

StepSearch:

`<think>` First, I need to identify the band that performed the Full Cooperation. Then, I will find out who is part of that band. `</think>`

`<search> who performed Full Cooperation </search>`

`<information>` Doc 1`<## Title: Full Cooperation ##>` "Full Cooperation" is a song by the American hip hop group Def Squad recorded for their sole album "El Niño" (1998). The song was released as the only single in promotion of the album on April 14, 1998.

`...</information>`

`<think>` From the information, I know the band that performed Full Cooperation is Def Squad. Now I need to find out who is part of the Def Squad band. `</think>`

`<search> who is part of Def Squad </search>`

`<information>` Doc 1`<## Title: Def Squad ##>` Def Squad is a rap supergroup consisting of Erick Sermon, Redman and Keith Murray. Jamal is considered an honorary member of the Def Squad. Before officially forming as a group to release an album in 1998, they had each been featured on tracks by each other. The Def Squad was formed following the disbandment of the Hit Squad, who broke up after the struggles between EPMD members Erick Sermon and Parrish Smith.

`...</information>`

`<think>` The members of the Def Squad band are Erick Sermon, Redman, and Keith Murray. Jamal is considered an honorary member. `</think>`

`<answer> Erick Sermon, Redman, Keith Murray </answer>`

Figure 15: STEPSEARCH provides more complete and detailed answers based on the information it obtains, while the quality of the labeled answers in the dataset is low.

Case study for STEPSEARCH (Fail).

Question: Where is Amílcar Cabral International Airport?

Ground Truth: ["central Atlantic Ocean", "Atlantic Ocean", "Atlantic"]

StepSearch:

`<think>` First, I need to identify Amílcar Cabral International Airport. Then, I will determine the location of that airport. `</think>`

`<search>` who is Amílcar Cabral International Airport named after? `</search>`

`<information>` Doc 1<## Title: Amílcar Cabral International Airport ##> Amílcar Cabral International Airport , also known as Sal International Airport, is the main international airport of Cabo Verde. The airport is named after the revolutionary leader Amílcar Cabral. It is located two km west-southwest from Espargos on Sal Island. ... `</information>`

`<think>` From the information, I found out that Amílcar Cabral International Airport is located two km west-southwest from Espargos on Sal Island. `</think>`

`<answer>` Sal Island `</answer>`

Figure 16: STEPSEARCH uses a single step to obtain multi-stage information for direct answers, providing more detailed results than the standard answer (Sal Island is in the Atlantic Ocean, but the rule matching result is incorrect).

Case study for STEPSEARCH (Fail).

Question: In which state is Póvoa de Santa Iria's municipality located?

Ground Truth: ["Lisbon District", "Lisbon"]

StepSearch:

`<think>` First, I need to identify the municipality of Póvoa de Santa Iria. Then, I will determine the state in which that municipality is located. `</think>`

`<search>` where is Póvoa de Santa Iria located? `</search>`

`<information>` Doc 1<## Title: Póvoa de Santa Iria e Forte da Casa ##> Póvoa de Santa Iria e Forte da Casa is a civil parish in the municipality of Vila Franca de Xira, Portugal... `</information>`

`<think>` From the information, I found out that Póvoa de Santa Iria is in the municipality of Vila Franca de Xira. Now I need to determine the state in which Vila Franca de Xira is located. `</think>`

`<search>` which state is Vila Franca de Xira in? `</search>`

`<information>` Doc 1<## Title: Vila Franca de Xira ##> Vila Franca de Xira () is a municipality in the Lisbon District in Portugal... `</information>`

`<think>` Vila Franca de Xira is a municipality in the Lisbon District in Portugal. Therefore, the state in which Póvoa de Santa Iria's municipality is located is Portugal. `</think>`

`<answer>` Portugal `</answer>`

Figure 17: STEPSEARCH confirmed the answer by continuing to check, while the question requested a state, the standard answer, "Lisbon District," is not a state.

Case study for STEPSEARCH (Fail).

Question: Who is the father of Empress Wang's husband?

Ground Truth: ["Yang Xingmi"]

StepSearch:

<think> First, I need to identify Empress Wang. Then, I will find her husband and determine his father. **</think>**

<search> who is Empress Wang **</search>**

<information> Doc 1<## Title: Empress Wang (Taizu) ##> Empress Wang (942–963) was a Chinese Empress consort of the Song Dynasty, married to Emperor Taizu of Song.

Doc 2<## Title: Empress Wang (Yang Pu) ##> Empress Wang , known as Empress Rang ("empress of the emperor who yielded") during Southern Tang, was the wife and empress of Yang Pu (Emperor Rui), the final ruler of the Chinese Five Dynasties and Ten Kingdoms period state Wu. As he was the only ruler who claimed the title of emperor, she was the only person to carry the title of empress during Wu.

Doc 3<## Title: Empress Dowager Wang (Rui) ##> Empress Dowager Wang (died 928) was a concubine of the late-Tang Dynasty warlord Yang Xingmi and the mother of his son Yang Pu, the final ruler and the only emperor of the Wu state founded upon the territory that Yang Xingmi took. During Yang Pu's reign as emperor, she was honored as empress dowager. **</information>**

<think> From the information, I know Empress Wang is the empress of Emperor Taizu of Song. Now I need to find out who the husband of Empress Wang is, which is Emperor Taizu of Song. Then I will find the father of Emperor Taizu of Song. **</think>**

<search> who is the husband of Empress Wang and what is his father **</search>**

<information> Doc 1<## Title: Empress Wang (Taizu) ##> Empress Wang (942–963) was a Chinese Empress consort of the Song Dynasty, married to Emperor Taizu of Song.

...

<think> The husband of Empress Wang is Emperor Taizu of Song. Now I need to find the father of Emperor Taizu of Song. **</think>**

<search> who is the father of Emperor Taizu of Song **</search>**

<information> Doc 1<## Title: Empress Wang (Taizu) ##> Empress Wang (942–963) was a Chinese Empress consort of the Song Dynasty, married to Emperor Taizu of Song.

...

</information>

<think> The father of Emperor Taizu of Song is Emperor Zhoutai of Later Zhou. **</think>**

<answer> Emperor Zhoutai of Later Zhou **</answer>**

Figure 18: STEPSEARCH found different Empress Wangs, but only conducted follow-up research based on the information in document one, ignoring the rest of the documents (One of Empress Wang's husband is Yang Pu, whose father is Yang Xingmi).