Process vs. Outcome Reward: Which is Better for Agentic RAG Reinforcement Learning

Wenlin Zhang[1], Xiangyang Li[2], Kuicai Dong[2], Yichao Wang[2], Pengyue Jia[1], Xiaopeng Li[1], Yingyi Zhang[1], Derong Xu[1], Zhaocheng Du[2], Huifeng Guo[2], Ruiming Tang[2], Xiangyu Zhao[1]

[1]Department of Data Science, City University of Hong Kong,

[2]Noah's Ark Lab, Huawei

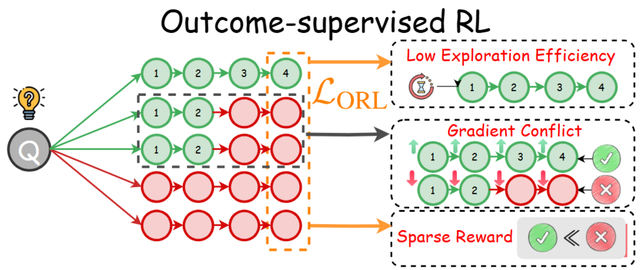
Background & Motivation

Background

Retrieval-Augmented Generation (RAG) enhances Large Language Models (LLMs) by integrating external knowledge, addressing issues of outdated information and hallucination. However, traditional RAG systems are constrained by static, linear workflows , lacking the adaptability required for multi-step reasoning and complex task management. To overcome these limitations, Agentic RAG systems have been proposed, enabling dynamic retrieval strategies, iterative context refinement, and adaptive workflows to handle complex queries beyond the capabilities of conventional RAG.

Motivation

Prevailing Agentic RAG methods, such as Search-R1, rely on **Outcome-Supervised Reinforcement Learning** (RL), using only the correctness of the final answer as the reward signal. This approach suffers from critical drawbacks: **sparse reward signals** , **low exploration efficiency** , and **gradient conflict**.



Contributions

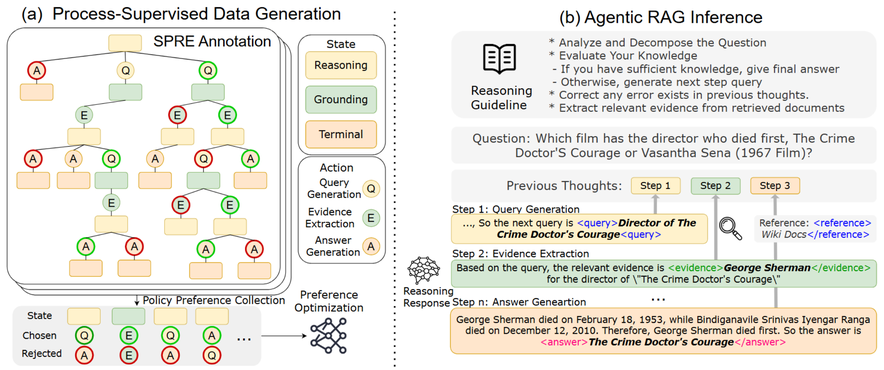
- We propose **ReasonRAG**, a novel Agentic RAG framework that leverages Process-Supervised Reinforcement Learning (PRL) for fine-grained policy optimization, moving beyond inefficient outcome-based rewards.
- We introduce **RAG-ProGuide**, a high-quality, automatically constructed dataset of 13,000 process-level preference pairs, designed to provide dense, step-by-step supervision for agentic RAG tasks.
- Our method achieves superior performance on five benchmark datasets using only 5k training instances, significantly outperforming the state-of-the-art Search-R1, which required 90k instances, demonstrating the high data efficiency and effectiveness of process-supervised RL.

AML Lab

HomePage

Code

Methodology



Process-Supervised Data Generation

- Shortest Path Reward Estimation (SPRE)

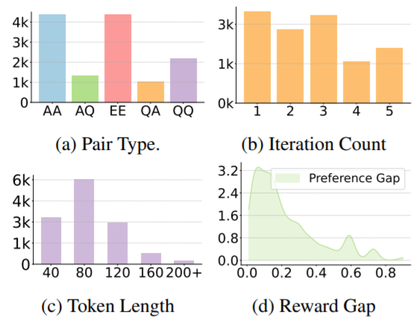
$$Q_t = \text{MonteCarlo}(x, y_{1:t}) = \frac{1}{k}\sum_{i=1}^{k} v(\text{rollout}_i) \cdot \alpha^{\text{step}(\text{rollout}_i)}$$

- Monte Carlo Tree Search (MCTS) for Process-level Exploration

$$\pi(a \mid s) = \text{LLM}(a \mid s) = \begin{cases} \pi_\theta(\cdot \mid x, y_{<i}, p_{\text{stage}}), & \text{if stage is Reasoning} \\ \pi_\theta(\cdot \mid x, y_{<i}, \text{docs}, p_{\text{stage}}), & \text{otherwise} \end{cases}$$

- RAG-ProGuide Dataset

Statistics	Number
Questions	4603
- PopQA	704 (15.3%)
- HotpotQA	2843 (61.8%)
- 2WikiMultihopQA	1056 (22.9%)
Actions	13289
- Query Generation	3295 (24.8%)
- Evidence Extraction	4305 (32.4%)
- Answer Generation	5689 (42.8%)
Avg./Min./Med./Max. Iteration	2.7/1/3/5
Avg./Min./Med./Max. Tokens	65.5/9/60/625



Process-Supervised Preference Optimization

$$\mathcal{L}(\theta) = -\mathbb{E}_{(x, y_{<t}, y_t^w, y_t^l)\sim\mathcal{D}}\left[\log\sigma\left(\beta\log\frac{p_\theta(y_t^w|x, y_{<t})}{p_\theta(y_t^l|x, y_{<t})}\right)\right]$$

Agentic RAG Inference

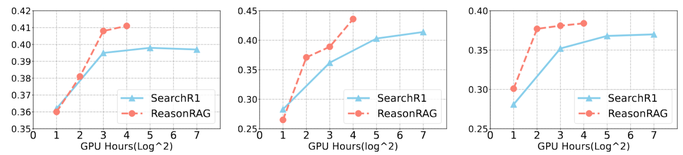
- Agent autonomously reasons by iteratively cycling through "Reasoning," "Grounding," and "Terminal" states to dynamically decide whether to answer, generate a new query, or extract evidence

Experiments

Main Results

Type	Method	PopQA		HotpotQA		2WikiMulti		Bamboogle		MuSiQue		Avg.	
		EM	F_1	EM	F_1	EM	F_1	EM	F_1	EM	F_1	EM	F_1
Zero-shot	Naïve Generation	12.7	16.5	15.7	24.8	20.2	28.0	6.4	17.4	2.7	10.2	11.5	19.4
	Standard RAG	38.4	44.7	29.3	39.9	29.4	36.3	17.6	24.1	6.7	15.1	24.3	32.0
Active	FLARE	14.3	17.6	18.1	25.7	27.9	32.8	12.0	20.8	4.3	12.6	15.3	21.9
	Self-RAG(146k)	22.7	33.9	21.0	29.7	12.0	25.2	1.6	10.9	4.6	13.3	12.4	22.6
Adaptive	AdaptiveRAG(3k)	36.6	41.5	29.1	40.7	24.2	33.4	18.4	26.1	6.9	14.3	23.0	31.2
RAG-CoT	Iter-Retgen	38.7	44.9	30.3	42.1	31.2	38.7	19.2	26.4	7.7	14.2	25.4	33.3
	IRCoT	36.2	43.6	27.7	41.5	23.5	32.5	17.2	22.5	8.6	13.2	22.6	30.7
Summary	RECOMP	40.5	45.8	29.7	41.2	33.2	39.4	21.7	28.6	9.2	15.8	26.9	34.2
	LongLLMLingua	39.2	45.1	31.4	43.2	34.5	40.2	20.3	27.4	8.7	14.9	26.8	34.2
	Selective-Context	34.9	41.5	19.3	27.3	20.3	29.7	15.3	22.6	6.1	13.7	19.2	27.0
Reasoning	Search-o1	33.2	40.3	24.8	38.1	16.4	27.1	30.4	40.6	6.3	13.7	22.2	31.96
	AutoRAG(10k)	38.6	44.1	33.3	43.7	39.5	46.1	24.8	32.2	11.3	18.3	29.5	36.9
	Search-R1(90k)	39.7	44.8	37.0	47.0	41.4	48.0	32.0	43.8	**14.6**	19.9	32.8	40.7
	ReasonRAG(5k)	**41.5***	**46.2***	**38.4***	**48.9***	**43.6***	**50.4***	**36.0***	**45.5***	12.8	**20.6***	**34.4***	**42.3***

Training cost and convergence speed comparison (EM%)

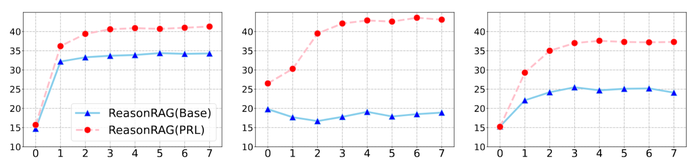


(a) PopQA (b) 2WikiMultiHopQA (c) HotpotQA

Effectiveness of Different Optimization Strategies

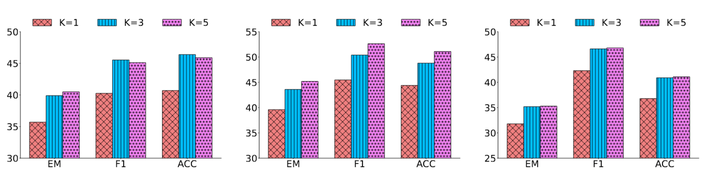
Method	PopQA		HotpotQA		2WikiMulti		Bamboogle		MuSiQue		Avg.	
	EM	F_1	EM	F_1	EM	F_1	EM	F_1	EM	F_1	EM	F_1
ReasonRAG (Base)	35.6	42.7	23.7	38.2	15.2	28.9	28.0	38.7	7.7	15.4	22.0	32.8
ReasonRAG (SFT)	31.6	37.4	26.8	38.7	35.1	40.9	17.6	27.3	8.6	15.5	23.9	32.0
ReasonRAG (RL-ORL): 5k queries	23.0	30.9	28.1	32.6	32.0	43.8	17.5	24.1	5.9	13.1	21.3	28.9
ReasonRAG (RL-ORL): 10k queries	39.5	45.7	36.7	46.7	40.5	47.2	30.7	40.6	12.6	19.5	32.0	39.9
ReasonRAG (RL-PRL)	**41.5**	**46.2**	**38.4**	**48.9**	**43.6**	**50.4**	**36.0**	**45.5**	**12.8**	**20.6**	**34.5**	**42.3**

Impact of Search Iteration on Performance



(a) PopQA (b) 2WikiMultiHopQA (c) HotpotQA

Effect of top-k retrieved documents



(a) PopQA (b) 2WikiMultiHopQA (c) HotpotQA