CoCoA: Collaborative Chain-of-Agents for Parametric-Retrieved Knowledge Synergy in Retrieval-Augmented Generation

Anonymous EMNLP submission

Abstract

Retrieval-Augmented Generation (RAG) has emerged as a promising framework for enhancing the capabilities of Large Language Models (LLMs), especially in knowledge-intensive tasks. Despite its advantages, current RAG methods often struggle to *fully exploit knowledge during generation*. In particular, the synergy between the model's internal parametric knowledge and external retrieved knowledge remains limited. Retrieved contents may sometimes mislead generation, while certain generated content can guide the model toward more accurate outputs. In this work, we propose **Co**llaborative **C**hain-**o**f-**A**gents, a framework designed to enhance synergy over both parametric and retrieved knowledge. Specifically, we first introduce **CoCoA-zero**, a training-free multi-agent RAG framework that first performs knowledge induction and then generates answers. Further, we develop a long-chain training strategy for **CoCoA**, which synthesizes long trajectories from the CoCoA-zero framework to fine-tune LLMs, improving their ability to explicitly integrate and collaboratively leverage internal and external knowledge. Experimental results demonstrate the superiority of CoCoA in open-domain QA and multi-hop QA. Our code will be available on GitHub.

1 Introduction

Large Language Models (LLMs) (Achiam et al., 2023; Touvron et al., 2023) have demonstrated strong performance across a wide range of natural language tasks. However, the knowledge they rely on is embedded in their parameters and cannot be easily updated as new information emerges (Ji et al., 2023; He et al., 2022). To address this limitation, the Retrieval Augmented Generation (RAG) framework introduces an external retrieval component that brings in external knowledge and integrates it into the input context of the LLMs. This design has led to notable improvements in various natural language processing applications (Gao et al., 2023; Lewis et al., 2020). Existing research has primarily aimed to improve two aspects of RAG: *retrieving more relevant information* during retrieval and *better utilizing that information to guide generation* during generation. Despite these efforts, most retrieval-augmented language models (RALMs) still emphasize external retrieval, while paying insufficient attention to the rich internal knowledge already encoded in model parameters. This internal knowledge is especially valuable for open-domain question answering, where many queries are factual and often already covered during pretraining.

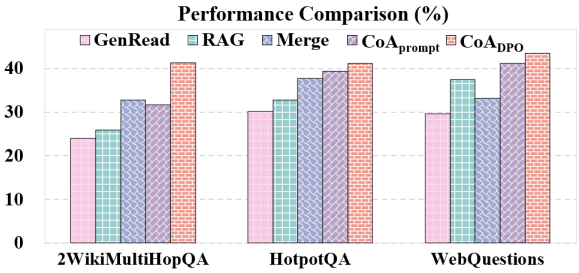

Figure 1: Evaluation on three datasets 2WikiMultiHopQA, HotpotQA, and WebQuestions. The Merge method is a simple strategy we use to verify the collaboration of internal and external knowledge. It directly generates a passage and merges it into the retrieved passages as the context of the LLM.

Specifically, as the knowledge in LLM's parameter becomes richer and the abilitiy of the LLM becomes stronger, sometimes answers with search information are not as good as direct answers. To validate the necessity of collaboratively synergizing internal (or parametric) and external (or retrieved) knowledge, we conduct experiments to compare performance. As shown in Fig. 1, across the three evaluation tasks, direct generation and GenRead (Yu et al., 2022) (use explicitly generated content) sometimes shows stronger performance. Also, we conduct a test experiment,"Merge", that explicitly integrates internal and external knowledge by generating a passage and retrieving the

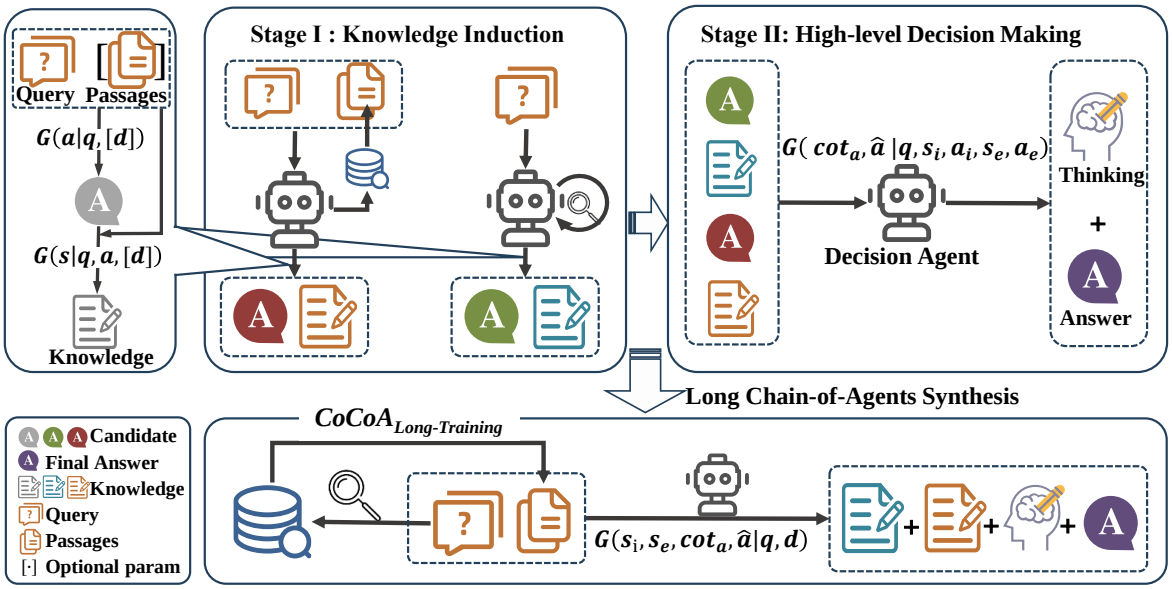

Figure 2: Illustration of the CoCoA framework. The top part is CoCoA-zero, a multi-agent collaboration framework. It integrates internal and external knowledge in a collaborative manner by first performing knowledge induction and then making decisions. The bottom part is the training strategy, which is based on CoCoA-zero and combines the trajectories of different agents into long chains to train and enhance the integration ability of the LLM.

passages as the final context simultaneously, as shown in Fig. 1. Its performance on multiple data sets is better than direct generation and generation with retrieval, which further verifies the potential of internal and external knowledge collaboration.

Existing methods solve the problem of knowledge collaboration through RAG pipeline optimization. Some approaches alleviate this through workflow or multi-module collaboration. For example, SURE (Kim et al., 2024a) generates multiple candidate answers and verifies them one by one to ensure reliability. CON (Yu et al., 2023) alleviates the harmful effects of external information by adding a processing chain. There are also some approaches that solve the problem of knowledge collaboration through enhanced training of the LLM. For instance, RAFT (Zhang et al., 2024) employs anti-noise training to enable the model to effectively utilize internal knowledge when external documents contain noise, while Self-RAG (Asai et al., 2023) learns to determine whether retrieval is needed in advance, thereby avoiding harmful content before retrieval. Despite these efforts, existing work still has notable limitations. On the one hand, methods like SURE tend to lose effectiveness as LLMs become more capable. On the other hand, Self-RAG and related methods cannot fully benefit from internally generated content.

To address the above challenges, we introduce **CoCoA**, which consists of a multi-agent reasoning framework and a training strategy that combines multi-agent trajectories into long chains to enhance LLM performance. Specifically, we first introduce **CoCoA-zero**, which features three agents: one for extracting pre-trained knowledge, one for retrieving external data, and one for making decisions by integrating both. This not only enables explicit construction of decoupled internal and external knowledge, but also provides collaborative reasoning traces for the training. Based on CoCoA-zero, we further introduce a train strategy for **CoCoA**, which significantly improves performance on knowledge-intensive tasks by integrating the collaborative capabilities of multi- agents into one model.

In general, our contributions can be summarized as follows:

- We introduce **CoCoA-zero**, a multi-agent framework that coordinates parametric and retrieved knowledge for improved generation.
- We develop a training paradigm for CoCoA, which distills multi-agent reasoning into long-chain, enabling LLMs to better exploit both internal and external knowledge.
- Extensive experiments demonstrate **CoCoA**'s effectiveness, offering insights for inference-time scaling on knowledge-intensive tasks.

2 Methodology

In this section, we present **CoCoA-zero** and **CoCoA**, as illustrated in Fig. 2. We first describe the multi-agent framework, CoCoA-zero, followed by the long-chain training strategy for CoCoA. The algorithm is presented in 1.

2.1 Preliminaries

We formalize the standard Retrieval-Augmented Generation framework. Given a query q and a corpus $\mathcal{D}$, the RAG system retrieves k relevant passages $C = \{c_1, c_2, \cdots, c_k\} \subset \mathcal{D}$ and generates an answer $\hat{a}$ based on the combined input. This process follows a retrieve-then-generate paradigm and can be formulated as:

$$\begin{aligned} C &= \mathcal{R}(q, \mathcal{D}, k), \\ \hat{a} &= \mathcal{G}(\mathcal{P}(q, C)), \end{aligned} \tag{1}$$

where $\mathcal{R}$ is the retriever, $\mathcal{P}$ is the prompt constructor that formats q and C, and $\mathcal{G}$ is the generator (e.g., a LLM) that predicts the final answer $\hat{a}$.

2.2 Two-stage RAG Framework: CoCoA-zero

In this section, we present our multi-agent RAG framework, CoCoA-zero, which also functions as the data synthesis pipeline for CoCoA. Stage 1 employs two specialized agents to induce knowledge from internal parameters and external retrieval, while Stage 2 introduces a third agent to synthesize their outputs for high-level decision-making.

2.2.1 Stage I: Knowledge Induction

It is challenging to extract implicit knowledge solely from the model's internal knowledge or retrieved passages. Inspired by GenRead (Yu et al., 2022) and SURE (Kim et al., 2024a), we design two dedicated agents for knowledge induction. Each agent first generates an answer to the question and then summarizes knowledge based on that answer.

Induction of Internal Knowledge. Directly allowing the model to explicitly generate its own internal knowledge is difficult to control and will inevitably result in sparse or inconsistent knowledge being generated. Following SURE (Kim et al., 2024a), we introduce conditional induction. Specifically, the Internal Knowledge Agent samples a candidate a_{in} from the LLM based on the question:

$$a_{\text{in}} = G(\mathcal{P}(q)) \tag{2}$$

We then prompt the model to generate a knowledge passage s_{in}, conditioned on q and a_{in}, which reflects the model's internal understanding:

$$s_{\text{in}} = G(\mathcal{P}(q, a_{\text{in}})). \tag{3}$$

Induction of External Knowledge. For retrieved passages, the External Knowledge Agent follows a similar procedure. Specially, it first retrieve some passages $C = \{c_1, c_2, \cdots, c_k\}$ from the corpus $\mathcal{D}$. Conditioned on both q and C, it produces a second candidate a_{ext}:

$$a_{\text{ex}} = G(\mathcal{P}(q, C)) \tag{4}$$

Then, conditioned on q, a_{ex} and C, the agent induces the external knowledge passage s_{ex}, :

$$s_{\text{ex}} = G(\mathcal{P}(q, a_{\text{ex}}, C)). \tag{5}$$

This conditional knowledge induction framework enhances the model's ability to articulate relevant knowledge, providing a strong foundation for the high-level decision-making in the next stage.

2.2.2 Stage II: High-level Decision Making

Building on the candidate answers and inductive knowledge obtained in Stage I, the second stage leverages the LLM's reasoning ability to perform high-level decision making.

Specifically, the Decision-Making Agent adopts COT (Wei et al., 2022) reasoning over the internal and external candidate answers and their corresponding knowledge. It will be prompted with all five components (questions, internal and external candidate answers and their corresponding inductive knowledge) and generate the final answer $\hat{a}$ through COT.

$$cot_{\text{a}}, \hat{a} = G(\mathcal{P}_{\text{cot}}(q, s_{\text{in}}, a_{\text{in}}, s_{\text{ex}}, a_{\text{ex}})) \tag{6}$$

Here, cot_{a} denotes the reasoning path that guides the final answer generation.

The model thereby functions as a high-level aggregator, reinforcing potentially consistent beliefs and resolving potential conflicts between internal beliefs and retrieved evidence. By explicitly modeling and comparing knowledge before committing to an answer, our framework improves the transparency and robustness of the decision process.

2.3 Training Strategy for CoCoA

Although multi-agent collaboration for internal and external knowledge coordination is simple and effective, how to achieve global optimization across multiple agents remains non-trivial.

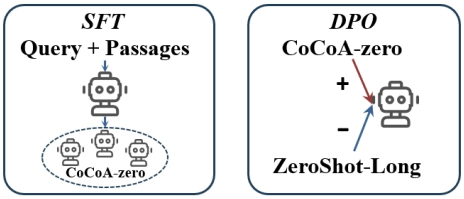

Figure 3: Illustration of the training for CoCoA.

To this end, we propose the collaborative Chain-of-Agents training strategy, which aims to optimize multi-agent collaboration end to end by supervising the LLM on long-form reasoning trajectories. These trajectories are synthesized from the multi-agent pipeline described in Section 2.2 and reflect the full reasoning process that integrates both parametric and retrieved knowledge.

2.3.1 Supervised Fine-Tuning

The CoCoA-zero framework is designed to (1) control the direction of knowledge generation via conditional induction, (2) decouple internal and external knowledge through parallel reasoning paths, and (3) integrate both sources through Chain-of-Thought decision making.

To supervise the model to achieve explicit and collaborative knowledge integration, we synthesize training samples by concatenating the intermediate results produced by CoCoA-zero into a single long-form response. Specifically, given a question q and a set of retrieved documents C, we integrate the intermediate results from the CoCoA-zero pipeline (i.e., internal induction s_{in}, external induction s_{ex}, the CoT reasoning trace cot_{a} during integration and the final answer $\hat{a}$) into a long response y and promote the evolution of model capabilities through the following supervision objectives:

$$\mathcal{L}_{\text{SFT}} = -\mathbb{E}_{(x,y)\sim\mathcal{D}}\big[\log P_\theta(s_{\text{in}}, s_{\text{ex}}, cot_{\text{a}}, \hat{a} \mid q, d)\big]. \tag{7}$$

This stage explicitly exposes the model to collaborative long samples, where the target outputs are synthesized based on the outputs of CoCoA-zero.

2.3.2 Direct Preference Optimization

To further enhance the model's ability to integrate internal and external knowledge, we perform Direct Preference Optimization (DPO) (Rafailov et al., 2023) following the SFT stage.

Specifically, we first prompt the LLM to generate structured long-form responses in a zero-shot setting and observe that the results are significantly inferior to those from the CoCo-zero pipeline. Motivated by this, we construct training instances where CoCoA-zero outputs serve as preferred responses y^+, and zero-shot outputs serve as rejected responses y^-. Each training instance includes a context $x = (q, d)$, a preferred response $y^+ = (s_{\text{int}} \oplus s_{\text{ext}} \oplus t \oplus \hat{a})$ from the CoCo-zero, and a rejected response y^- from the ZeroShort-Long baseline.

The DPO objective encourages the model to prefer y^+ over y^- by optimizing:

$$\begin{aligned}\mathcal{L}_{\text{DPO}}(\pi_\theta) = -\,\mathbb{E}_{(x,y^+,y^-)\sim\mathcal{D}}\big[\log\sigma\big(&\\ \beta\cdot\log\pi_\theta(y^+|x) - \beta\cdot\log\pi_\theta(y^-|x)\big)&\\ +\,\alpha\cdot\big(-\log\pi_\theta(y^+|x)\big)\big]&\end{aligned} \tag{8}$$

where $\pi_\theta(y|x)$ denotes the unnormalized log-probability of response y under the model θ.

Intuitively, the zero-shot prompting is a simple compliance with instructions, which will produce entangled reasoning. In contrast, CoCo-zero will produce a path of internal and external knowledge collaboration, resulting in a more reliable answer.

DPO thus bridges symbolic multi-agent collaboration and end-to-end generation, enabling the model to internalize structured reasoning through preference-based supervision.

Algorithm 1 CoCoA: Example of one sample

Input: Query q, corpus $\mathcal{D}$, hyperparameters k
Output: Final answer $\hat{a}$ or training sample y

CoCoA-zero:
- $a_{\text{in}} \leftarrow G_{\text{in}}(\mathcal{P}(q))$ ▷ Candidate
- $s_{\text{in}} \leftarrow G_{\text{in}}(\mathcal{P}(q, a_{\text{in}}))$ ▷ Internal knowledge induction
- $C \leftarrow \mathcal{R}(q, \mathcal{D}, K)$ ▷ Top-K retrieval
- $a_{\text{ex}} \leftarrow G_{\text{ex}}(\mathcal{P}(q, C))$ ▷ Candidate
- $s_{\text{ex}} \leftarrow G_{\text{ex}}(\mathcal{P}(q, a_{\text{ex}}, C))$ ▷ External knowledge induction
- $(cot_{\text{a}}, \hat{a}) \leftarrow G_{\text{dm}}(\mathcal{P}(q, s_{\text{in}}, s_{\text{ex}}, a_{\text{in}}, a_{\text{ex}}))$ ▷ Decision making

if Supervised Fine-tuning **then**
- $y \leftarrow (s_{\text{in}} \oplus s_{\text{ex}} \oplus cot_{\text{a}} \oplus \hat{a})$ ▷ Target
- Update model with $\mathcal{L}_{\text{SFT}}$ in Eq. 7.

end if
if DPO Training **then**
- $y^- \leftarrow G(\mathcal{P}_{\text{ZS}}(q, C))$
- $y^+ \leftarrow (s_{\text{in}} \oplus s_{\text{ex}} \oplus cot_{\text{a}} \oplus \hat{a})$
- Update model with $\mathcal{L}_{\text{DPO}}$ in Eq. 8,

end if
return $\hat{a}$ or the trained model CoCoA

Method	2WikiMQA		HotpotQA		WebQuestions		NaturalQA‡		PopQA_longtail‡		TriviaQA‡	
	EM	F1	EM	F1	EM	F1	EM	F1	EM	F1	EM	F1
Llama-3.1-Instruct Train-free & w/o retrieval												
Llama-3.1-70B	*33.80*	*33.43*	*37.00*	*37.89*	*44.83*	*43.92*	*47.29*	*47.14*	*29.95*	*30.79*	*77.89*	*78.93*
Llama-3.1-8B	27.60	28.35	24.00	27.09	40.11	**39.98**	33.91	35.69	22.59	23.48	62.87	64.17
Llama-3.1-8B+COT	23.80	26.55	26.20	32.26	38.04	39.43	36.51	38.78	23.23	24.14	64.90	66.98
Llama-3.1-8B+GenRead	24.00	23.92	29.20	31.15	29.53	29.67	30.39	31.22	26.45	26.42	54.12	54.29
Llama-3.1-Instruct Train-free & w/ retrieval												
Llama-3.1-70B	*22.00*	*23.12*	*35.20*	*38.03*	*39.76*	*39.05*	*47.87*	*46.97*	*40.60*	*38.75*	*70.97*	*71.44*
Llama-3.1-8B+StrandardRAG	26.80	25.07	31.40	34.16	37.65	37.32	45.01	44.37	**40.89**	38.51	66.83	67.16
Llama-3.1-8B+COT	22.40	25.25	32.40	38.71	35.73	36.17	42.85	43.28	39.60	37.93	65.85	67.54
Llama-3.1-8B+CON	19.00	21.32	32.80	38.67	34.40	38.05	43.19	**45.43**	39.17	38.71	65.64	66.82
Llama-3.1-8B+SURE	18.40	21.32	32.00	37.26	32.48	39.01	41.00	44.90	40.31	**39.62**	63.14	62.91
CoCoA-zero-8B	**31.40**	**31.92**	**37.40**	**41.20**	**43.11**	39.13	**45.21**	43.27	38.81	38.60	**70.73**	**69.99**
CoCoA-zero-70B	*40.40*	*39.86*	*43.20*	*45.74*	*43.46*	*41.64*	*52.19*	*51.24*	*44.82*	*43.83*	*78.35*	*77.57*
RALM w/ retrieval & w/ Training												
Self-RAG 7B	37.40	17.93	33.40	20.57	44.64	25.75	40.47	44.46	44.25	15.64	66.30	37.27
Self-RAG 13B	38.80	22.61	35.40	21.64	**45.87**	25.31	43.99	**48.60**	**44.39**	16.14	68.74	38.22
DeepSeek-R1-Distill-8B	36.80	25.79	35.00	32.66	44.34	31.87	45.21	36.78	42.75	37.87	65.62	58.07
CoCoA-SFT-8B	41.00	36.87	**39.40**	**46.31**	42.96	41.32	48.28	48.25	43.25	42.21	70.72	70.39
CoCoA-DPO-8B	**42.00**	**40.58**	39.00	43.39	44.83	**42.21**	**48.28**	46.26	43.60	**42.35**	**71.52**	**70.42**

Table 1: EM/F1 of different methods experimented on six datasets. The best and second best scores are highlighted in **bold** and underlined, respectively. ‡ represents the OOD (Out-of-Distribution) evaluation dataset.

2.4 Optimization Analysis

We compare independent training and chain-of-agents training under a simplified two-step setting involving pre-generation processing followed by answer generation.

$$\mathcal{L}\text{indep} = -\log P_\theta(s|x,d) - \log P_\phi(\hat{a}|s) \quad (9)$$

$$\mathcal{L}\text{chain} = -\log P_\theta(s|x,d) - \log P_\theta(\hat{a}|x,d,s) \quad (10)$$

Gradient comparison:

$$\frac{\partial \mathcal{L}\text{chain}}{\partial \theta} = \frac{\partial \mathcal{L}\text{indep}}{\partial \theta} + \Delta_g \quad (11)$$

where $\Delta_g := \frac{\partial}{\partial \theta}\big[-\log P_\theta(\hat{a}|x,s,d)\big]$. Δ_g captures feedback from the answer to the preprocessing, which is absent in independent training. Chain training is a special type of multi-task learning that helps to break out of local optimization. The experimental results are in Section 3.6, and detailed derivations are in Appendix C.

3 Experiments

In this section, we report our experiments results, and provide a analysis of them.

3.1 Implementation Details

Training Data We sample subsets from the training sets of HotpotQA (Ho et al., 2020a), 2WikiMultiHopQA (Ho et al., 2020b) and WebQuestions (Berant et al., 2013), then synthesize data using the CoCoA-zero and filter them based on gold answers. This results in 6.8k filtered samples for SFT. For DPO, we select 1151 samples, which are the ones that are answered incorrectly by zero-shot but correctly by the CoCoA-zero framework. For each sample, we gather 5 relevant passages using CONTRIEVER (Izacard et al., 2021).

Training Details We fine-tune LLaMA3.1-8B with LoRA (r=16, α=16, dropout=0.05). During SFT, we train for 5 epochs with a learning rate of 3e-5. For DPO, we used β=0.2 and α=0.2 (RPO), with a learning rate of 5e-6. All experiments are conducted on a single A100 GPU.

Inference Details During inference, we use Contriever (Izacard et al., 2021) as the retriever and set k to 5. For all datasets, we use 21M English Wikipedia (Karpukhin et al., 2020) dump as the source passages for the retrieval. Prompts for the experiments can be found in Appendix E.

3.2 Datasets and Evaluation Metrics

Eval Datasets To evaluate the effectiveness and generalization of CoCoA, we conduct experiments on three open-domain question answering datasets: WebQuestions (Berant et al., 2013), NaturalQuestions (Kwiatkowski et al., 2019), and TriviaQA (Joshi et al., 2017), as well as three multi-hop question answering benchmarks: HotpotQA (Ho et al., 2020a), 2WikiMultiHopQA (Ho et al., 2020b), and PopQA_longtail (Asai et al.,

2023). Dataset statistics are summarized in Table 2, and further details are provided in Appendix A.

Evaluation Metrics We report both exact match (EM) and F1 scores. Following Asai et al. (2023); Mallen et al. (2022), we adopt a non-strict **EM** metric that deems a prediction correct if it contains the gold answer, rather than requiring an exact string match. F1 measures token-level overlap between the predicted and gold answers. In our setting, longer responses often yield higher **EM** scores due to increased coverage, but may reduce **F1** scores by introducing irrelevant content. Thus, considering both metrics provides a more balanced evaluation.

Task Type	Datasets	# Samples
Multi-HopQA	2WikiMultiHopQA	500
	HotpotQA	500
	PopQA_longtail	1399
OpenQA	WebQuestions	2032
	NaturalQA	3610
	TriviaQA	11313

Table 2: Description of tasks and evaluation datasets.

3.3 Baselines

We selected several of the most representative methods for comparison. 1) **StandardRAG**, which is the most classic "retrieve-then-read" paradigm. 2) **Chain-Of-Thought** (Wei et al., 2022): Uses CoT prompting to generate intermediate reasoning steps before producing the final answer. 3) **Chain-Of-Note** (Yu et al., 2023): Refines and summarizes retrieved passages prior to answering. 4) **GenRead** (Yu et al., 2022): Generates self-contained intermediate context to answer questions, effectively replacing retrieval with generation. 5) **Self-RAG** (Asai et al., 2023): Employs adaptive retrieval and self-reflection to decide when and how to use external knowledge. 6) **DeepSeek-R1-Distill-8B** (Guo et al., 2025): A distilled LLaMA-8B model released by DeepSeek-R1, trained on curated reasoning data. All retrieval-based methods use top-5 passages. Other experimental settings follow those reported in the original papers.

3.4 Main Results

Experimental results are presented in Table 1, and we summarize the key findings as follows:

Superiority and Generalization of CoCoA: Both our train-free framework CoCoA-zero and CoCoA methods achieve state-of-the-art performance across almost all datasets. In particular, CoCoA improves the EM and F1 of 2WikiMultiHopQA tasks by **15.2%** and **15.51%** respectively. CoCoA-zero improves the average EM and F1 of all tasks by 3.01% and 2.93% respectively, while other Train-free methods are ineffective. Moreover, despite being trained with limited data, CoCoA also performed well on other out-of-distribution datasets, demonstrating its robustness.

Advantage of CoCoA-zero Framework: CoCoA-zero surpasses other train-free methods by a clear margin and matches the performance of StandardRAG with a 70B model while using only an 8B LLM. Moreover, CoCoA-zero improves the average EM of all tasks by **3.01%** under the 8B setting, and by **7.67%** under the 70B setting. This proves that larger models are more beneficial to our collaboration, and also illustrates the importance of internal knowledge of stronger LLMs.

CoCoA Training vs. Reasoning Distillation: For the fine-tuning method, we used long-chain training to achieve strong performance, which is better than the 8B distilled version of DeepSeek-R1. This suggests that in knowledge-intensive tasks, expanding with chain-of-thought reasoning may be less effective, while explicitly outputting internal and external key knowledge proves more superior.

Benefit of Direct Preference Optimization: Comparing our supervised and preference-aligned variants, DPO training brings consistent improvements across all datasets. This suggests that contrastive preference learning helps the model better align collaborative responses with high-quality multi-agent outputs.

Method	2WikiMQA	HotpotQA	WebQuestions
CoCoA-zero	**31.66**	**39.30**	**41.12**
w/o Internal	23.26 (↓ 8.40)	36.56 (↓ 2.74)	39.10 (↓ 2.02)
w/o External	28.97 (↓ 2.69)	30.96 (↓ 8.34)	38.97 (↓ 2.15)
w/o Thinking	30.38 (↓ 1.28)	37.17 (↓ 2.13)	39.75 (↓ 1.37)
Zero-Shot	18.55 (↓ 13.11)	35.01 (↓ 4.29)	35.38 (↓ 5.74)
StandardRAG	25.94 (↓ 5.72)	32.78 (↓ 6.52)	37.49 (↓ 3.63)

Table 3: Ablation study on knowledge induction and decision-making. The zero-shot method for knowledge integration in Section 2.3.2 is also included. The average of EM and F1 is used for fair evaluation.

3.5 Ablation Study I: Different Modules

To better understand the contribution of each module in CoCoA-zero, we conduct an ablation study

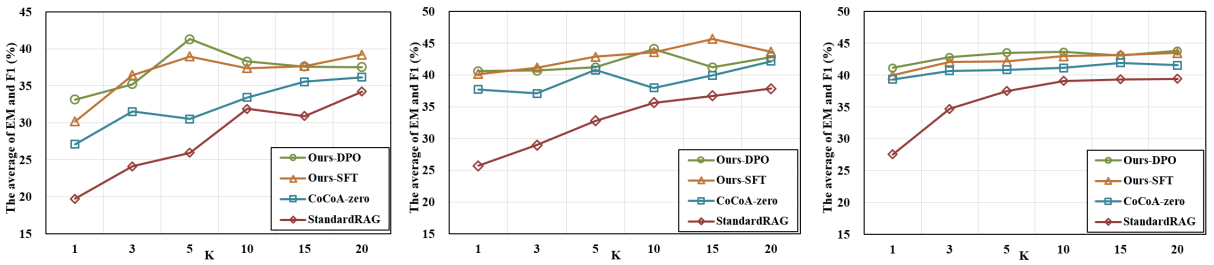

Figure 4: Performance changes as the number of documents in the context changes.

by selectively removing internal/external induction and the reasoning mechanism.

As shown in Table 3, removing internal induction significantly degrades performance, especially by 8.4% on 2WikiMultiHopQA, indicating the importance of leveraging parametric knowledge. Similarly, excluding external induction also leads to a noticeable performance drop across all datasets, highlighting the complementary role of retrieved knowledge. Moreover, disabling the reasoning mechanism in decision making results in a moderate but consistent decrease, suggesting that explicit reasoning over both knowledge contributes to deeper understanding and more accurate responses.

To further verify the effectiveness of multi-agent collaboration and the rationale of negative sample selection in DPO training, we include a zero-shot variant using only prompt-based alignment without fine-tuning. As expected, it shows the lowest performance, confirming the need for learned coordination between internal and external knowledge.

Overall, these results confirm the effectiveness of our multi-agent collaboration design, where each component plays a non-trivial role in achieving optimal performance.

3.6 Ablation Study II: Training Strategies

To evaluate the effectiveness of our training strategy for CoCoA, we conduct an ablation study comparing different training configurations on the LLaMA3.1-8B model. As shown in Table 4, "Long-DPO$_{8B}$" achieves the best overall performance, confirming the benefit of aligning long-form outputs via Direct Preference Optimization.

The "Short-SFT$_{8B\times3}$" variant, where each task segment is trained on a separate model, shows clear degradation in performance, especially on 2WikiMultiHopQA. This indicates that separating induction and reasoning capabilities into isolated modules weakens the model's ability to holistically integrate information across steps. The "Short-SFT$_{8B}$" variant, which combines three instruction capabilities into a single model but retains short-form generation, performs better than "Short-SFT$_{8B\times3}$" but still falls behind our approaches. This shows that simply merging instructions is slightly less performant than our long chain consolidation.

Our training strategy for CoCoA, represented by "Long-DPO$_{8B}$" and "Long-SFT$_{8B}$" variants, explicitly models multi-agent collaboration as a unified long-form output. The superior performance of these models underscores the advantage of training models to generate cohesive and contextually rich responses rather than fragmented predictions. This, to a certain extent, provides new perspectives for the expansion of knowledge-intensive long chains.

Method	2Wiki	HotpotQA	WebQ	Average
Long-DPO$_{8B}$	**41.29**	41.20	**43.52**	**42.00**
Long-SFT$_{8B}$	38.94	**42.86**	42.14	41.31
Short-SFT$_{8B}$	33.91	40.04	40.13	38.03
Short-SFT$_{8B\times3}$	28.31	40.58	39.84	36.24

Table 4: Ablation study of the training strategy for CoCoA. For simplicity and fairness, the average of EM and F1 is used as the metric.

3.7 When the Number of K Changes

In order to better explore the robustness of our CoCoA with respect to the number of documents, we set K to vary in the interval [1, 3, 5, 10, 15, 20]. The results are shown in Fig. 4. Overall, our method outperforms StandardRAG across different values of K. Moreover, our method achieves stronger performance than StandardRAG when given less context. We speculate that this is because our model can better utilize internal knowledge, especially when given less information. However, our advantage decreases when the number of documents is too large. We speculate that this is due to the long context bottleneck of the model.

In summary, our method demonstrates strong robustness across different context sizes and provides

a practical solution in settings with limited external information or constrained retrieval capacity.

3.8 Generalization to Non-QA Tasks

To further evaluate the generalization ability of CoCoA, we test its performance on fact verification and multiple-choice tasks. As shown in Figure 5, our training did not reduce the performance of these tasks compared to standard RAG. In fact, in some cases, we even observed a slight improvement. One explanation is that our training strategy encourages collaborative output that leverages the capabilities of the LLM, rather than injecting knowledge directly, and thus possesses a certain degree of universality.

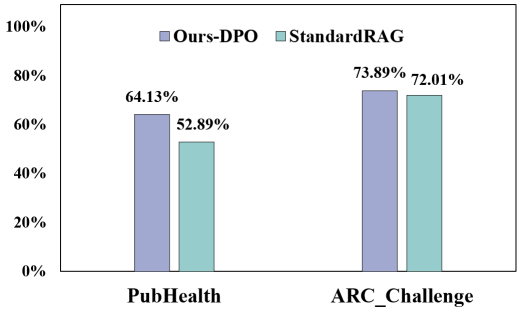

Figure 5: Illustration of accuracy changes when transferring to non-QA tasks, with accuracy as the metric.

4 Related Works

4.1 Retrieval-augmented Generation

In recent years, in order to solve the problems of outdated knowledge in the model and hallucination of large language models, retrieval-augmented generation has been introduced (Fan et al., 2024; Gao et al., 2023), and many efforts have been made in two aspects: "*how to retrieve more relevant information*" including retriever fine-tuning (Nian et al., 2024) and query optimization (Ma et al., 2023; Wang et al., 2023a, 2024a) and "*how to better use the retrieved information to generate answers*" including domain fine-tuning (Wang et al., 2024b; Zhang et al., 2024; Yue et al., 2025) and controlled decoding strategies (Shi et al., 2023). Our CoCoA falls into the second category: better utilization of knowledge.

4.2 RAG Pipeline Optimization

Pipeline optimization usually adds pre-generation processing, retrieval intent identification, or optimizes the pipeline as a whole. For example, Glass et al. (2022); Kim and Lee (2024) and Yu et al. (2023) introduce reranking and refinement steps before generation, mitigating the impact of noisy retrieved passages. SKR (Wang et al., 2023b) and UAR (Cheng et al., 2024) avoid unnecessary retrieval by adding retrieval intent identification processes before generation. SURE(Kim et al., 2024a) first generates multiple candidate answers and performs conditional summary verification based on the candidate answers, allowing LLMs to focus on specific contexts. However, these methods either overly emphasize external context and become dependent on retrieval content, or overlook the synergistic integration of the model's internal knowledge with retrieved external information, potentially limiting answering performance.

4.3 RALM Enhancement

Retrieved-Augmented Language Model(RALM) enhancement is usually achieved by adjusting the language model to achieve effective use of the information. One common approach is to train the language model itself. For example, RAFT (Zhang et al., 2024) improves the model's ability to resist noise in external context by introducing noise resistance training. REAR (Wang et al., 2024b) achieves the model's trade-off between external context and internal knowledge by training the model's relevance-guided generation capabilities. Self-RAG (Asai et al., 2023) trains LLMs to decide whether to perform retrieval and to improve their self-reflection ability. Another approach involves guiding the decoding (Shi et al., 2023; Kim et al., 2024b). For instance, CAD (Shi et al., 2023) enforces absolute trust in retrieved information by using contrastive decoding under the assumption that external information is fully correct. However, both approaches tend to underutilize the model's internal knowledge, which may constrain the quality and informativeness of its responses.

5 Conclusion

We present **CoCoA**, a retrieval-augmented generation framework that enhances LLM performance by enabling effective collaboration between parametric and retrieved knowledge. Through a two-stage multi-agent pipeline and the long-chain training strategy, our method achieves strong performance on QA tasks, highlighting CoCoA's effectiveness and providing a new insight into the long-chain expansion of knowledge-intensive tasks.

Limitations

While CoCoA has demonstrated excellent performance and provided valuable insights into collaboration with parametric and retrieved knowledge, there are still some limitations:

- The current design focuses on a specific agent collaboration pattern via long-chain training. Its applicability to broader or alternative multi-agent architectures remains to be examined.
- Our approach focuses on inducing knowledge, but does not explicitly capture non-knowledge cues (such as logical clues) that can be equally important for complex reasoning tasks.
- Although the approach performs robustly under limited supervision, its scaling behavior with respect to larger models and datasets has not been systematically explored.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

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

Jonathan Berant, Andrew Chou, Roy Frostig, and Percy Liang. 2013. Semantic parsing on freebase from question-answer pairs. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1533–1544.

Qinyuan Cheng, Xiaonan Li, Shimin Li, Qin Zhu, Zhangyue Yin, Yunfan Shao, Linyang Li, Tianxiang Sun, Hang Yan, and Xipeng Qiu. 2024. Unified active retrieval for retrieval augmented generation. *arXiv preprint arXiv:2406.12534*.

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*, pages 6491–6501.

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

Michael Glass, Gaetano Rossiello, Md Faisal Mahbub Chowdhury, Ankita Naik, Pengshan Cai, and Alfio Gliozzo. 2022. Re2g: Retrieve, rerank, generate. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2701–2715.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.

Hangfeng He, Hongming Zhang, and Dan Roth. 2022. Rethinking with retrieval: Faithful large language model inference. *arXiv preprint arXiv:2301.00303*.

Xanh Ho, Anh-Khoa Duong Nguyen, Saku Sugawara, and Akiko Aizawa. 2020a. Constructing a multi-hop qa dataset for comprehensive evaluation of reasoning steps. *arXiv preprint arXiv:2011.01060*.

Xanh Ho, Anh-Khoa Duong Nguyen, Saku Sugawara, and Akiko Aizawa. 2020b. Constructing a multi-hop qa dataset for comprehensive evaluation of reasoning steps. *arXiv preprint arXiv:2011.01060*.

Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. 2021. Unsupervised dense information retrieval with contrastive learning. *arXiv preprint arXiv:2112.09118*.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. 2023. Survey of hallucination in natural language generation. *ACM Computing Surveys*, 55(12):1–38.

Mandar Joshi, Eunsol Choi, Daniel S Weld, and Luke Zettlemoyer. 2017. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1601–1611.

Vladimir Karpukhin, Barlas Oğuz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. *arXiv preprint arXiv:2004.04906*.

Jaehyung Kim, Jaehyun Nam, Sangwoo Mo, Jongjin Park, Sang-Woo Lee, Minjoon Seo, Jung-Woo Ha, and Jinwoo Shin. 2024a. Sure: Summarizing retrievals using answer candidates for open-domain qa of llms. *arXiv preprint arXiv:2404.13081*.

Kiseung Kim and Jay-Yoon Lee. 2024. Re-rag: Improving open-domain qa performance and interpretability with relevance estimator in retrieval-augmented generation. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 22149–22161.

Youna Kim, Hyuhng Joon Kim, Cheonbok Park, Choonghyun Park, Hyunsoo Cho, Junyeob Kim, Kang Min Yoo, Sang-goo Lee, and Taeuk Kim. 2024b. Adaptive contrastive decoding in retrieval-augmented generation for handling noisy contexts. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 2421–2431.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, et al. 2019. Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:453–466.

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

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474.

Xinbei Ma, Yeyun Gong, Pengcheng He, Hai Zhao, and Nan Duan. 2023. Query rewriting in retrieval-augmented large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5303–5315.

Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Hannaneh Hajishirzi, and Daniel Khashabi. 2022. When not to trust language models: Investigating effectiveness and limitations of parametric and non-parametric memories. *arXiv preprint arXiv:2212.10511*, 7.

Jinming Nian, Zhiyuan Peng, Qifan Wang, and Yi Fang. 2024. W-rag: Weakly supervised dense retrieval in rag for open-domain question answering. *arXiv preprint arXiv:2408.08444*.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36:53728–53741.

Weijia Shi, Xiaochuang Han, Mike Lewis, Yulia Tsvetkov, Luke Zettlemoyer, and Scott Wen-tau Yih. 2023. Trusting your evidence: Hallucinate less with context-aware decoding. *arXiv preprint arXiv:2305.14739*.

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

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

Haoyu Wang, Ruirui Li, Haoming Jiang, Jinjin Tian, Zhengyang Wang, Chen Luo, Xianfeng Tang, Monica Cheng, Tuo Zhao, and Jing Gao. 2024a. Blendfilter: Advancing retrieval-augmented large language models via query generation blending and knowledge filtering. *arXiv preprint arXiv:2402.11129*.

Liang Wang, Nan Yang, and Furu Wei. 2023a. Query2doc: Query expansion with large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9414–9423.

Yile Wang, Peng Li, Maosong Sun, and Yang Liu. 2023b. Self-knowledge guided retrieval augmentation for large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 10303–10315.

Yuhao Wang, Ruiyang Ren, Junyi Li, Wayne Xin Zhao, Jing Liu, and Ji-Rong Wen. 2024b. Rear: A relevance-aware retrieval-augmented framework for open-domain question answering. *arXiv preprint arXiv:2402.17497*.

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

Wenhao Yu, Dan Iter, Shuohang Wang, Yichong Xu, Mingxuan Ju, Soumya Sanyal, Chenguang Zhu, Michael Zeng, and Meng Jiang. 2022. Generate rather than retrieve: Large language models are strong context generators. *arXiv preprint arXiv:2209.10063*.

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

Shengbin Yue, Siyuan Wang, Wei Chen, Xuanjing Huang, and Zhongyu Wei. 2025. Synergistic multi-agent framework with trajectory learning for knowledge-intensive tasks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 25796–25804.

Tianjun Zhang, Shishir G Patil, Naman Jain, Sheng Shen, Matei Zaharia, Ion Stoica, and Joseph E Gonzalez. 2024. Raft: Adapting language model to domain specific rag. *arXiv preprint arXiv:2403.10131*.

A Dataset

Here, we introduce in detail the datasets we used, which are seven datasets on four tasks.

2WikiMultiHopQA (Ho et al., 2020b) and **HotpotQA** (Ho et al., 2020a): Both datasets are multi-hop question answering datasets based on Wikipedia. Considering the limitation of experimental cost, we used the sub-sampling set published by Trivedi et al. (2022); Kim et al. (2024a), which is obtained by extracting 500 questions from the validation set of each dataset.

WebQuestions (Berant et al., 2013): Constructed from questions posed by the Google Suggest API, where the answers are specific entities listed in Freebase.

NaturalQA (Kwiatkowski et al., 2019): A dataset designed to support comprehensive QA systems. It consists of questions from real Google search queries. The corresponding answers are text spans from Wikipedia articles, carefully identified by human annotators.

TriviaQA (Joshi et al., 2017): A compilation of trivia questions paired with answers, both originally pulled from online sources.

PopQA_longtail (Asai et al., 2023): A long-tail subset of PopQA (Mallen et al., 2022), consisting of 1,399 rare entity queries whose monthly Wikipedia page views are less than 100.

Training Data We sampled subsets from the training sets of HotpotQA (Ho et al., 2020a), 2WikiMultiHopQA (Ho et al., 2020b) and WebQuestions (Berant et al., 2013), then used the CoCoA-zero framework to synthesize data and filtered them with gold answers. Finally, we selected 6.8k filtered samples, including 3k, 3k, and 0.8k from the three datasets, respectively. For the DPO training data, we screen out 1151 samples, which are the ones that are answered incorrectly by zero-shot but correctly by the CoCoA-zero. For each sample, we gathered 5 relevant passages using the most common retriever Contriever (Izacard et al., 2021).

B Training Details

We fine-tune LLaMA3.1-8B with LoRA (r=16, α=16, dropout=0.05) on a maximum input length of 2048. LoRA is applied to attention projection layers. During SFT, we trained for 5 epochs with a batch size of 1, gradient accumulation of 4, and a learning rate of 3e-5. For DPO, a β value of 0.2 is applied, using a sigmoid loss function, while RPO is configured with an α value of 0.2. The learning rate was set to 5e-6 and other settings are the same as SFT. During inference, we use the vllm (Kwon et al., 2023) accelerated inference framework, and to ensure repeatability, we set the temperature to 0.0. All experiments are conducted on a single A100 GPU with 80GB or 40GB memory.

C Optimization Analysis

We analyze the difference between independent training and long chain training in terms of the form of loss. We simplify the steps in this analysis, i.e., there are only two steps in the chain, pre-generation processing first and then answering.

When the two agents optimize independently, the loss takes the following form:

$$\mathcal{L}_{\text{indep}} = -\log P_\theta\big(s \mid x, d\big) \;-\; \log P_\phi\big(\hat{a} \mid s\big). \tag{12}$$

Here, θ and θ' are optimized independently.

When two agents use long chain optimization, the loss is as follows:

$$\begin{aligned}\mathcal{L}_{\text{chain}} &= -\log P_\theta\big(s, \hat{a} \mid x, d\big) \\ &= -\log P_\theta\big(s \mid x, d\big) \;-\; \log P_\theta\big(\hat{a} \mid x, d, s\big).\end{aligned} \tag{13}$$

Gradient propagation:

The gradient of the first term in Eq. (12) is,

$$\frac{\partial \mathcal{L}_{\text{indep}}}{\partial \theta} = \frac{\partial \left[-\log P_\theta(s \mid x, d)\right]}{\partial \theta} \tag{14}$$

The gradient of the Eq. (13) is,

$$\begin{aligned}\frac{\partial \mathcal{L}_{\text{chain}}}{\partial \theta} &= \frac{\partial \left[-\log P_\theta(s \mid x, d)\right]}{\partial \theta} \\ &\quad + \frac{\partial \left[-\log P_\theta(\hat{a} \mid x, s, d)\right]}{\partial \theta} \\ &= (14) \;+\; \Delta_g\end{aligned} \tag{15}$$

$$\Delta_g \;:=\; \frac{\partial \left[-\log P_\theta(\hat{a} \mid x, s, d)\right]}{\partial \theta}. \tag{16}$$

Here, Δ_g is the additional gradient that the answer-loss naturally back-propagates to the preprocessing parameters when the *same* network θ produces both tokens. In the independent setting $\Delta_g = 0$ by construction, so the preprocessor never "hears" whether the answer is correct, which is not conducive to the consistency of the response. The chain objective restores this missing credit assignment signal, thus performing a special kind of multi-task learning on both stages, optimizing them instead of each in isolation, potentially helping to escape from local optimal solutions.

Method	2WikiMultiHopQA			HotpotQA			WebQuestions		
	EM	F1	Avg	EM	F1	Avg	EM	F1	Avg
CoCoA-zero	31.40	31.92	31.66	37.40	41.20	39.30	43.11	39.13	41.12
w/o Thinking	30.00	30.76	30.38	36.00	38.34	37.17	39.17	40.32	39.75
w/o Internal	22.60	23.93	23.26	34.00	39.11	36.56	40.01	38.20	39.10
w/o External	28.40	29.53	28.97	30.00	31.92	30.96	39.81	38.13	38.97
Zero-Shot	17.60	19.51	18.55	33.20	36.81	35.01	34.45	36.31	35.38
Standard RAG	26.80	25.07	25.94	31.40	34.16	32.78	37.65	37.32	37.49

Table 5: Ablation study of internal/external induction and reasoning in decision making. In addition, a zero-shot method for explicit internal and external knowledge integration is added for comparison. For simplicity and fairness, the average of EM and F1 is used as the metric.

Method	2WikiMultiHopQA			HotpotQA			WebQuestions		
	EM	F1	Avg	EM	F1	Avg	EM	F1	Avg
Long-DPO$_{8B}$	42.00	40.58	41.29	39.00	43.39	41.20	44.83	42.21	43.52
Long-SFT$_{8B}$	41.00	36.87	38.94	39.40	46.31	42.86	42.96	41.32	42.14
Short-SFT$_{8B}$	28.60	28.03	28.31	39.00	42.15	40.58	41.19	38.48	39.84
Short-SFT$_{8B\times3}$	35.00	32.81	33.91	37.60	42.48	40.04	41.29	38.96	40.13

Table 6: Ablation study of the training strategy for CoCoA. For simplicity and fairness, the average of EM and F1 is used as the metric

D Full Results

We supplemented the detailed results of the ablation experiment as shown in Table 5 and Table 6.

E Prompt Templates

All the prompt templates used by our proposed CoCoA are shown in Table 9 and Table 8. And special instructions are added to section 3.8 corresponding to different tasks as shown in Table 7.

Task	Task Instruction
ARC-C	Given four answer candidates, A, B, C and D, choose the best answer choice. Please answer with the capitalized alphabet only, without adding any extra phrase or period. Do not exceed one word.
PubHealth	Is the following statement correct or not? Say true if it's correct; otherwise say false. Don't capitalize or add periods, just say "true" or "false". Do not exceed one word.

Table 7: Full list of instructions used during zero-shot evaluations. For open-domain QA, we don't use any task specific instruction.

Task:Prompt used by "CoCoA"

Instruction:
1. First, provide background for the question. Write a passage that is relevant to the question only based on your knowledge.
2. Second, refer to the provided passages to generate a summary. Cite and write a passage that is relevant to the question only based on the provided passages.
3. Third, refer to the information from the above two sources, verify the accuracy of the facts and the consistency of the logic, and predict the final answer.
Passages:\n{$passages$}\n
Question:\n{$question$}
Generate Format:
<Internal>\nxxx (your background based on your knowledge)\n<\\Internal>
<External>\nxxx (your summary based on the provided passages)\n<\\External>
<Thinking>\nxxx\n<\\Thinking>
<Answer>\nxxx (your short answer consisting of only a few words)<\\Answer>

Table 8: The prompt used by "CoCoA".

Task	Task Instruction
External Candidate	### Passages:\n {$passages$}\n\n ### Instruction:\n Answer the question below concisely in a few words.\n\n ### Input:\n{$question$}\n
External Induction	### Instruction:\n Refer to the provided passages to generate a summary that meets the following conditions:\n 1. Cite and Write a passage that can support the prediction about the question only based on the provided passages.\n 2. No more than 200 words.\n 3. Do not respond with anything other than the S̈ummary:.\n ### Passages:\n {$passages$}\n\n ### Question:\n {$question$}\n ### Prediction:\n {$answer$}\n\n ### Generate Format:\n ### Summary: xxx\n
Internal Candidate	### Instruction:\n Answer the question below concisely in a few words.\n\n ### Input:\n{$question$}\n
Internal Induction	### Instruction:\n Please provide background for the question that meets the following conditions:\n 1. Write a passage that can support the prediction about the question only based on your knowledge.\n 2. No more than 200 words.\n 3. Do not respond with anything other than the B̈ackground:.\n ### Question:\n {$question$}\n ### Prediction:\n {$answer$}\n\n ### Generate Format:\n ### Background: xxx\n
Decision-Making	### Internal Reasoning Path: \n{$induction_{in}$}\n\n ### Internal Prediction 1: \n{$answer_{in}$}\n\n ### External Reasoning Path: \n{$induction_{ex}$}\n\n ### External Prediction 2: \n{$answer_{ex}$}\n\n ### Instruction:\n Refer to the information from the above two sources, verify the accuracy of the facts and the consistency of the logic, and choose the best prediction. ### Question:\n{$question$}\n ### Generate Format:\n ### Thingking: xxx (Please think step by step)\n ### Short Answer: xxx (just in a few words)\n

Table 9: A list of prompts used by CoCoA-zero.