Synergistic Multi-Agent Framework with Trajectory Learning for Knowledge-Intensive Tasks

Shengbin Yue[1], **Siyuan Wang**[2], **Wei Chen**[3], **Xuanjing Huang**[1], **Zhongyu Wei**[1]*

[1] Fudan University, Shanghai, China
[2] University of Southern California, Los Angeles, USA
[3] Huazhong University of Science and Technology, Wuhan, China
sbyue23@m.fudan.edu.cn, sw_641@usc.edu, lemuria_chen@hust.edu.cn, {xjhuang,zywei}@fudan.edu.cn

Abstract

Recent advancements in Large Language Models (LLMs) have led to significant breakthroughs in various natural language processing tasks. However, generating factually consistent responses in knowledge-intensive scenarios remains a challenge due to issues such as hallucination, difficulty in acquiring long-tailed knowledge, and limited memory expansion. This paper introduces SMART, a novel multi-agent framework that leverages external knowledge to enhance the interpretability and factual consistency of LLM-generated responses. SMART comprises four specialized agents, each performing a specific sub-trajectory action to navigate complex knowledge-intensive tasks. We propose a multi-agent co-training paradigm, Long Short- Trajectory Learning, which ensures synergistic collaboration among agents while maintaining fine-grained execution by each agent. Extensive experiments on five knowledge-intensive tasks demonstrate SMART's superior performance compared to widely adopted knowledge internalization and knowledge enhancement methods. Our framework can extend beyond knowledge-intensive tasks to more complex scenarios.

Code — https://github.com/yueshengbin/SMART

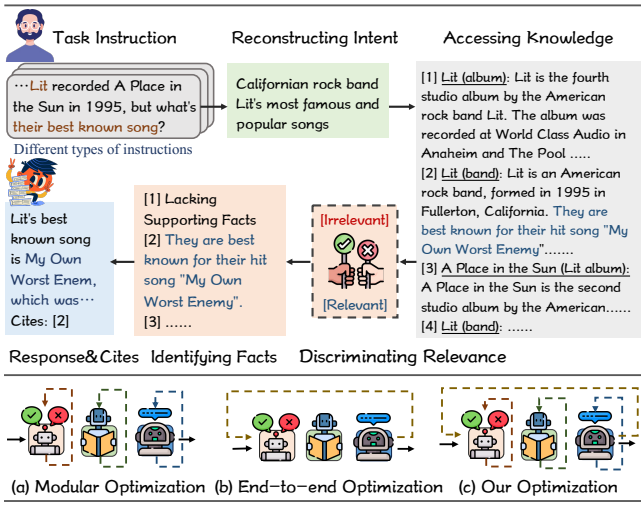


Figure 1: Example of our long trajectory for knowledge-intensive scenarios (Top) and optimization comparison of multi-agent frameworks (Bottom). Solid and dashed arrows indicate inference and optimization paths, respectively.

Introduction

Researchers continue to pursue empowering intelligent systems to generate factually consistent responses in knowledge-intensive tasks (Singhal et al. 2022; Yue et al. 2023a; Wang et al. 2022a). Although Large Language Models (LLMs) internalize substantial world knowledge within their parameter memory, they still suffer from fabricating facts, due to their inherent drawbacks, *e.g.*, hallucination (Ji et al. 2023), trouble in acquiring long-tailed knowledge (Kandpal et al. 2023) and struggle to expand their memory (De Cao, Aziz, and Titov 2021). These issues significantly underscore the necessity of incorporating external knowledge from non-parametric (*i.e.*, retrieval-based) memories.

Current methods typically augment LLMs with retrieved knowledge to generate responses, which face three main challenges. (1) *Complex query intent*: the diverse nature (semantics and form) of instructions (*e.g.*, multiple choice, multi-turn dialogue, and complex questions) leads to confusion regarding the query intent of knowledge. (2) *Distractors in retrieved knowledge*: knowledge retrieval inevitably introduces noises of varying granularity (document and sentence), with irrelevant documents and superfluous spans distracting the response and resulting in more severe hallucinations. (3) *Insufficient knowledge utilization*: LLMs tend to rely more on their implicit knowledge (parameter memory) rather than fully exploiting provided external facts (Huang et al. 2023). This fact-following disloyalty invalidates the knowledge incorporation process. Existing knowledge enhancement efforts (Shi et al. 2023; Ma et al. 2023; Asai et al. 2023) do not comprehensively address these multi-stage challenges. To this end, we propose a multi-agent framework, **SMART**, to integrate different actions to tackle all challenges within complex knowledge-intensive tasks, where each agent performs a specific action. This comprises an Intent Reconstructor to clarify knowledge intents, a Knowledge Retriever to access external knowledge based on intent, a Fact Locator to evaluate retrieved knowledge and

*Corresponding author

identify factual spans, and a Response Generator that faithfully utilizes and cites available facts. This process can enhance the knowledge interpretability and response factuality.

However, a major concern remains in how to equip each agent with the necessary capability for corresponding actions while minimizing errors during agent streamline for better overall knowledge-intensive performance. This has been a longstanding challenge in improving multi-agent frameworks, especially as most (Yao et al. 2023; Hong et al. 2023) operate in a non-training manner. Specifically, *On one hand*, modular operations, where separate learned modules are pipelined with each dedicated to a specific agent, can streamline the processing. However, this can lead to error accumulation as mistakes in earlier modules propagate through the pipeline. *On the other hand*, encouraging LLM variants to imitate the entire trajectory, while mitigating the fragmentation and error propagation seen in modular systems, this long-term and global supervision cannot guarantee the precise fine-grained execution by each agent, as it fails to balance the attention each agent devotes to diverse input signals. Overall, maintaining synergy while ensuring the contribution of various stakeholders is essential.

To address this, we propose a multi-agent cooperative training method, namely **Long Short- Trajectory Learning**, which consists of two stages. In the first stage, short trajectory learning activates each specific agent in the framework. Next, long trajectory learning ensures synergy across multi-agents through trajectory skeleton learning. To establish a common supervisory signal for both phases while achieving different training objectives for each, we design special tokens (*i.e.*, trajectory head-end tokens) to allow each agent to identify the attributed trajectories and learn inter-agent interaction signals during training. Specifically, the former phase learns the task output under the prompt of the trajectory-head token, so that the framework learns to distinguish between different agents and confirm the fine-grained information of interest. This independence enables more efficient training with the utilization of existing NLP datasets for pre-training and targeted optimization. The latter stage requires both predictions of task output and intermittent trajectory tokens throughout the process, *i.e.*, establishing a navigation path from the previous agent to the next. Our learning approach enables multi-agent systems to collaboratively navigate a long and complex trajectory while concurrently upholding a nuanced representation of each agent.

We conduct experiments on five knowledge-intensive tasks, including fact verification, multiple-choice reasoning, open-domain question answering and long-form generation. Results demonstrate that our framework significantly outperforms pre-trained and instruction-tuned LLMs with more parameters (knowledge internalization methods), and widely adopted knowledge enhancement methods. Further analysis reveals that our long-short trajectory learning enables flexible plug-in combinations of agents while maintaining performance, which is beyond the reach of current end-to-end training systems. Additionally, the framework achieves impressive performance using only over 40 % of long trajectory data, substantially reducing the cost and complexity of developing a high-performance multi-agent framework. We envision our framework as a general paradigm that extends beyond knowledge-intensive tasks to more complex scenarios, enabling any multi-agent framework to internalize tailored trajectories.

Method

Figure 2 provides an overview of our co-framework. We first introduce our multi-agent framework with four key agents performing distinct trajectories. Next, we explain the data construction method and detail the Long-Short Trajectory Learning for optimizing framework synergies.

Multi-Agent Framework

To address multi-stage complex challenges in knowledge-intensive scenarios, we design a multi-agent framework to execute complex long trajectories. This framework incorporates four key agents: intent reconstructor ($\mathcal{A}_\mathrm{i}$), knowledge retriever ($\mathcal{A}_\mathrm{r}$), fact locator ($\mathcal{A}_\mathrm{l}$), and response generator ($\mathcal{A}_\mathrm{g}$). Each agent serves a specific sub-trajectory, and the final response is obtained by synergizing these agents.

Intent Reconstructor. The $\mathcal{A}_\mathrm{i}$ agent aims to clarify the knowledge query intent from user instructions. It possesses four primary capabilities: integrating contextual clues, identifying key query, unifying task formulation, and intent decomposition, to handle diverse instructions. For example, in multi-turn dialogues, $\mathcal{A}_\mathrm{i}$ models long-term history for intent. For noisy instructions, it filters out irrelevant information to identify key queries. For various task formats such as multi-choice QA, $\mathcal{A}_\mathrm{i}$ formulate all inputs as a query format for subsequent processing. When handling multi-hop queries like "Who was born earlier, person A or person B?", $\mathcal{A}_\mathrm{i}$ breaks them down into multiple sub-intents, *i.e.,* each person's birth date. By flexibly applying these capabilities, this agent obtains clear query intent to access external knowledge.

Knowledge Retriever. The $\mathcal{A}_\mathrm{r}$ agent accesses external knowledge bases (e.g., Wikipedia) and obtains relevant knowledge candidates based on reconstructed intents. Specifically, it is driven by an off-the-shelf retrieval model (Izacard et al. 2021) and acquires top-k knowledge document candidates from the knowledge base for each knowledge intent. Details of our knowledge retriever setup and the corpus are described in Appendix Sec. B.3.

Fact Locator. The $\mathcal{A}_\mathrm{l}$ agent aims to locate factual evidence from knowledge candidate sets via document- and sentence-level assessments. Specifically, it assesses the relevance of each knowledge document to the given instruction to determine relevant ones. It then identifies the factual spans from relevant documents as evidence. The fact locator serves two primary purposes: 1) It enables the agent to check its relevance judgments to minimize the distraction of extraneous spans of the document, and allows the response phase to focus more on fact spans. 2) By explicitly learning to locate facts, it enhances the interpretability of the knowledge application process and bolsters user credibility.

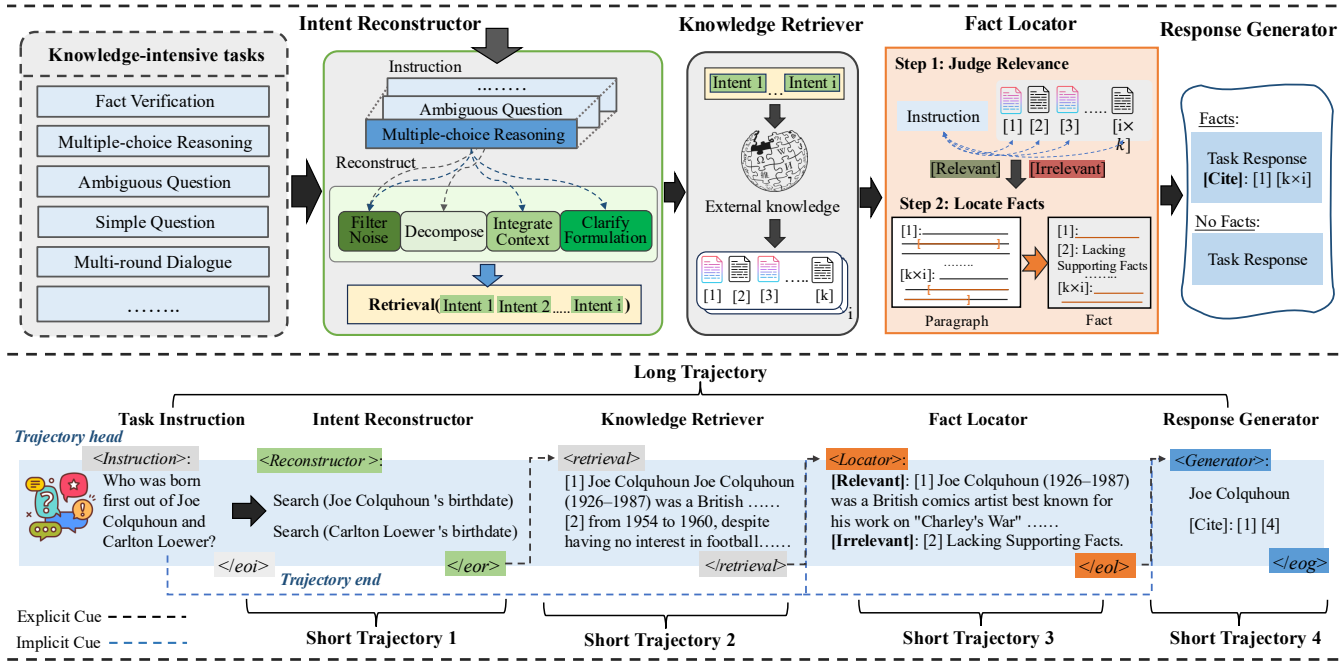


Figure 2: Overview of our multi-agent framework with long- and short-trajectory learning. This framework incorporates four agents: intent reconstructor, knowledge retriever, fac locator, and response generator.

Response Generator. The $\mathcal{A}_{\mathrm{g}}$ agent finally generates responses to user instructions. When facts are provided, it adjusts its knowledge preferences to adhere to them, and ultimately outputs citations to validate loyalty further. In the absence of such information, the response generator relies on its knowledge memory to formulate responses.

Inference Overview. The systematic procedure is delineated in the following steps: $\mathcal{A}_{\mathrm{i}}$ first mines the explicit intent $\bar{q} = \{q_1, q_2, ..., q_m\}$ from the instruction x. Next, $\mathcal{A}_{\mathrm{r}}$ retrieves top-k knowledge documents $\bar{d} = \{d_1, d_2, ..., d_{k\times m}\}$ using each intent q_m. Then, $\mathcal{A}_{\mathrm{l}}$ determines each relevant knowledge passage and further locates the fact span $f \subset d_{k\times m}$. Finally, $\mathcal{A}_{\mathrm{g}}$ utilizes the previous execution trajectory to generate response y and citations when facts exist, otherwise $\mathcal{A}_{\mathrm{g}}$ utilizes only x. In the t-th step, the Agent $\mathcal{A}$ generates a response r_t and a head token h_{t+1} of the next trajectory based on the current state of the system:

$$r_t, h_{t+1} = \mathcal{A}\left(x, \tau_{t-1}\right), \tag{1}$$

where $\tau_{t-1} = \{h_1, r_1, e_1, ..., h_{t-1}, r_{t-1}, e_{t-1}\}$ denotes the previous execution trajectory. e denotes the trajectory end token. In addition, $\mathcal{A}_{\mathrm{i}}$, $\mathcal{A}_{\mathrm{l}}$ and $\mathcal{A}_{\mathrm{g}}$are built upon same LLMs to fulfill their roles. The pseudo-code for inference is referenced in Appendix.

Trajectory Dataset Construction

To implement long-short trajectory learning to optimize our multi-agent framework, we construct the Trajectory dataset. We collect samples from over 12 knowledge-intensive tasks to ensure coverage of various instruction semantics and formats, such as fact verification (Thorne et al. 2018), dialogue (Dinan et al. 2018; Anantha et al. 2021), open-domain Q&A (Kwiatkowski et al. 2019; Stelmakh et al. 2022; Geva et al. 2021), and commonsense reasoning (Mihaylov et al. 2018; Huang et al. 2019). Detailed statistics are in Table 5 of Appendix. Our dataset contains two components: the long-trajectory subset and the short-trajectory subset. The data construction follows two distinct principles:

Long-trajectory subset. The long-trajectory subset aims to precisely mimic our multi-agent framework inference-time process, which emphasizes the synergy and logical interaction between agents. Existing work (Asai et al. 2023) has demonstrated the effectiveness of the powerful LLM (*e.g.*, GPT3.5, GPT4 (Achiam et al. 2023)) as a critic model. Given an input-output pair (x, y), we create supervised data under the guide of the retrieval ($\mathcal{R}$) and critic model ($\mathcal{C}$). We enable $\mathcal{C}$ to unleash the knowledge intents $\bar{q}$ in x according to the instruction type. Then, $\mathcal{R}$ retrieves the top-k knowledge documents based on every $\bar{q}$. For each document, $\mathcal{C}$ further evaluates whether the passage is relevant based on (x, y). If a passage is relevant, $\mathcal{C}$ further locates and extracts the fact spans. Finally, we combine the data and insert the trajectory header and end token (*e.g.*,<Reconstructor>, </eor>) into each trajectory. Trajectory tokens are identifiers that serve as the skeleton of the multi-agent framework. In total, we construct 142,507 elaborated instances.

Short-trajectory subset. Unlike the long-trajectory subset, the short-trajectory subset facilitates the training of individual capabilities for each intelligent agent. This isolation allows us to acquire data directly from a huge amount of existing knowledge-intensive tasks through some simple processing. Thus, we sample from the established NLP and SFT datasets, appending the requisite trajectory header and

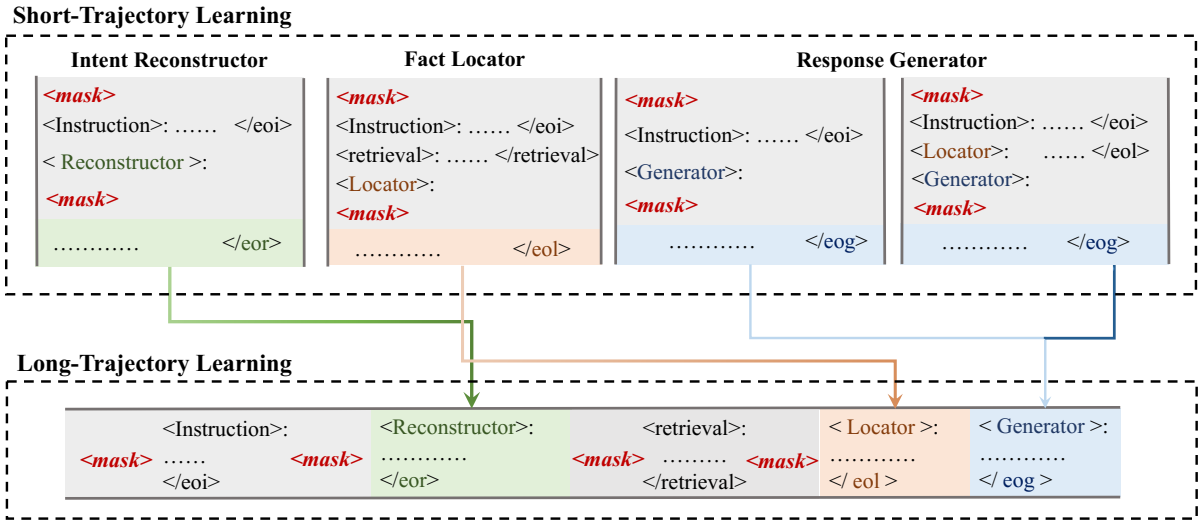


Figure 3: Overview of Long-Short Trajectory Learning. It consists of two stages, for short trajectory learning, under a given trajectory head, requires insight into the various explicit and implicit signals in each particular task. For long-trajectory learning, LLM executes the entire process by predicting different trajectory tokens, ensuring the synergism of different short-trajectories.

Type	Trajectory Tokens Head	End	Input	Output
$\mathcal{A}_{\text{i}}$	<Reconstructor>	</eor>	x	$\bar{q}$
$\mathcal{A}_{\text{r}}$	**<retrieval>**	**</retrieval>**	$\bar{q}$	$\bar{d}$
$\mathcal{A}_{\text{l}}$	<Locator>	</eol>	x, $\bar{d}$	γ,$\bar{f}$
$\mathcal{A}_{\text{g}}$	<Generator>	</eog>	x,$\bar{d}$ / x	y

Table 1: Four types of trajectory tokens. x, $\bar{q}$, $\bar{d}$, γ, $\bar{f}$ and $\bar{y}$ indicate instruction, intent, knowledge document, relevance tag, fact evidence and response, respectively.

end token. Note that the existing NLP datasets do not fulfill our requirements for intent reconstructing, we employ the methodology utilized in the long-trajectory subset collection. Table 1 exhibits the inputs and outputs of each short trajectory under the responsibility of each agent. In addition, the response generator contains two types of inputs to help adapt its knowledge preferences. We construct a total of 359,791 instances.

To summarize. Two keys are in the construction: the Long-trajectory subset is crafted to emphasize synergy, and the Short-trajectory subset can be easily accessed in large quantities to emphasize uniqueness. Refer to Appendix Sec.A for the detail of data construction.

Long-Short Trajectory Learning

Effectively fine-tuning a trajectory system consisting of multi-agents is a complex task: on the one hand, each agent has its specific trajectory signals of attention. On the other hand, the transformation between different trajectories requires the collaboration of the agents. In addition, the cost of trajectory data construction for a multi-agent framework greatly hinders the development of such systems. To this end, we propose Long-Short Trajectory Learning for our multi-agent framework, which consists of two stages, Short Trajectory and Long Trajectory Learning. As shown in Figure 3, Under the guidance of the trajectory head-end token pairs, the intuition is that Short Trajectory Learning first delineates the responsibilities of each agent to develop their unique capabilities, and then Long Trajectory Learning learns the interactions between them. This can be understood as initially activating each agent that masters short trajectories within a broader trajectory framework, and then exploring the interconnections between those agents to navigate the full long trajectory.

Short Trajectory Learning. Short Trajectory Learning is the training of individual capabilities for a single agent. In the context of a long trajectory, it is important to note that short trajectories spanning multiple steps do not necessarily exhibit a strong dependence on preceding short trajectories. To illustrate this point, consider the case of a fact locator, which primarily relies on the original user query and the retrieved results, rather than having a strict dependence on the queries generated in Intent Reconstructor. Similarly, the Response Generator necessitates only the question itself or a combination of the question and the located facts. As shown in Figure 3, the short trajectory learning first activates each short agent in the framework to focus on the fine-grained signals. Given the short-trajectory subset $\mathcal{D}_{\text{short}} = \{\mathcal{D}_{\text{intent}}, \mathcal{D}_{\text{locator}}, \mathcal{D}_{\text{generator}}\}$, we initialize a pre-trained LLM and train it on $\mathcal{D}_{\text{short}}$. For each example $\{(x_i; h_i), (y_i; e_i)\} \subset \mathcal{D}_{short}$, we use a standard conditional language modeling objective, maximizing likelihood:

$$\mathcal{L}\left(\mathcal{D}_{short}\right) = \sum_{i} \log P_{LM}\left(y_i; e_i \mid x_i; h_i\right), \tag{2}$$

Given the inputs and trajectory header, the agent learns to predict the outputs, *i.e.*, delineate different belonging trajectories for the agent to make them understand the fine-grained

Task	Health	ARC-C	PopQA	Squad1	ASQA		
Metric	Acc	Acc	Acc	Acc	Str_EM	R-L	Mauve
Knowledge internalization methods							
Alpaca2 $_{7B}\star$	44.78	36.43	25.58	11.50	14.42	28.72	51.24
Mistral-Instruct $_{7B}$	65.45	57.84	22.37	14.97	20.80	32.20	33.47
Llama-2-Chat $_{7B}$	47.95	47.95	25.44	14.13	16.79	32.35	24.21
Vicuna-v1.5 $_{13B}$	63.01	57.59	17.94	15.25	31.95	22.99	68.41
Llama-2-Chat $_{13B}$	62.20	48.72	21.22	15.97	19.97	30.37	40.23
ChatGPT	**76.08**	**77.3**	29.30	22.90	39.94	35.73	44.63
Knowledge enhancement methods							
Alpaca2 $_{7B}\star$	26.44	35.15	33.38	21.41	23.59	27.21	50.09
REPLUG $_{7B}\star$	41.72	47.26	37.24	24.23	26.54	33.25	54.03
VANILLA $_{7B}\star$	29.52	42.74	37.52	25.92	32.25	34.93	39.54
RAIT $_{7B}\star$	52.98	62.10	38.02	23.86	25.68	15.99	12.35
INTERACT $_{7B}\star$	65.45	48.12	<u>41.31</u>	**31.52**	<u>34.54</u>	35.51	43.45
SelfRag $_{7B}$	68.99	65.52	40.67	22.39	28.68	34.11	83.00
MMAgent $_{3*7B}\star$	70.82	63.99	36.88	23.79	33.04	<u>36.49</u>	<u>88.98</u>
SMART (OURS)	<u>73.18</u>	<u>65.58</u>	**42.60**	<u>27.80</u>	**41.16**	**40.66**	**91.47**

Table 2: Comparison results against knowledge internalization and knowledge enhancement methods. $\star$ denotes the method we reproduce based on the same base. $\star$ denotes re-implemented methods based on the same initial model. The **bold** numbers represent the best results and the <u>underlined</u> numbers represent the second.

representations of the corresponding tasks. This phase utilizes easily accessible and extensive data to build the basic capabilities of the trajectory, reducing the cost of such a framework while maintaining the creativity and versatility of the agent.

Long Trajectory Learning. After the above stage, the framework is equipped with four independent agents. Long Trajectory Learning further grooms the LLM to establish logical associations between agents in an end-to-end manner. We train based on the previous stage on the long-trajectory subset $\mathcal{D}_{long}$. Specifically, given instruction x, long trajectory learning forces the LLM to learn the long trajectory process:

$$\begin{aligned} \mathcal{L}\left(\mathcal{D}_{Long}\right) &= \sum_i \log P_{LM}\left(\tau_i^R; \tau_i^I; \tau_i^G \mid x_i\right), \\ \tau_i^T &= \left[h_i^T; y_i^T; e_i^T\right], T \subset \{R, I, G\}. \end{aligned} \tag{3}$$

where R, I and G denote the Intent Reconstructor, Fact Locator and Response Generator, respectively. Unlike short trajectory learning (Eq. 2), the framework learns both to predict the target output for each short trajectory as well as from the previous trajectory end e^T to the next trajectory head h^{T+1}. In essence, the trajectory token serves as a skeleton in the learning process, guiding the agent not only to grasp a fine-grained representation of the intra-trajectory but also inter-trajectory interactions.

Experiment Setting

Setup

Task and Dataset. We evaluate our framework in a range of knowledge-intensive downstream tasks. Including (1) Fact verification: PubHealth (Akhtar, Cocarascu, and Simperl 2022) is a fact verification dataset about public health; (2) Multiple-choice reasoning: ARC-Challenge (Clark et al. 2018) is a multiple-choice questions dataset about science exam. (3) Open-domain question answering: contains two short-form QA datasets, PopQA (Mallen et al. 2022), and SQuAD 1.1 (Rajpurkar et al. 2016). (4) Ambiguous question answering: ASQA (Gao et al. 2023) is ambiguous factoid question of the long form response. Details of evaluation data, including size, and evaluation metrics are available in Appendix Sec. B.1.

Baselines. We compare our framework with a wide range of baseline methods in two categories. (1) Knowledge internalization methods (General-purpose LLMs): ChatGPT (gpt-3.5-turbo-0125) (Zheng et al. 2023) (Ouyang et al. 2022), Mistral-Instruct-v0.2-7B (Jiang et al. 2023), Llama-2-Chat-7B/13B (Touvron et al. 2023), Vicuna-v1.5-13B (Zheng et al. 2023) and Alpaca2-7B [1] (Zheng et al. 2023). (2) Knowledge enhancement methods: REPLUG-7 (Shi et al. 2023), VANILLA-7B (Gao et al. 2023), INTERACT-7B (Gao et al. 2023), RAIT-7B (Lin et al. 2023), SelfRAG-7B (Asai et al. 2023), MMAgent-3*7B (modular approach). More details are in Appendix Sec. B.2.

Implementation Details

Due to page limitations, details of our training and evaluation are in Appendix Sec. B.3.

Experiment Result

Main Result

Comparison against knowledge internalization methods. As shown in Table 2, our framework shows a significant performance advantage over equivalently sized fine-tuned LLMs across all tasks. In comparison to larger LLMs (Vicuna-v1.5-13B and Llama-2-Chat-13B), which possess greater internalized knowledge, our SMART framework also exhibits superior performance in all metrics. Furthermore,

[1] https://github.com/tatsu-lab/stanford_alpaca

	Health (Acc)	ARC-C (Acc)	Pop (Acc)	AS (Em)
Training ablation				
SMART (L)	**72.15**	**60.22**	**37.27**	**36.10**
w/o $\mathcal{A}_{\text{f}}$	70.13	58.95	34.31	34.77
w/o $\mathcal{A}_{\text{i}}$	69.82	54.94	35.17	34.41
w/o All	57.95	56.99	21.15	20.05
Inference ablation				
SMART (L+S)	**73.18**	**65.58**	**42.60**	**41.16**
w/o $\mathcal{A}_{\text{f}}$	71.63	62.45	37.45	36.10
w/o $\mathcal{A}_{\text{i}}$	71.22	60.11	39.88	35.30
w/o All	69.32	58.81	16.79	31.32

Table 3: Training Ablation and inference ablation for the contribution of different agents. L and S denote long-trajectory and short-trajectory learning, respectively. w/o $\mathcal{A}_{\text{f}}$, w/o $\mathcal{A}_{\text{i}}$, and w/o All denote no fact Locator, no intent reconstructor, and only response generator.

our framework surpasses ChatGPT in all evaluated metrics for PopQA (long-tail knowledge evaluation), Squad1, and ASQA. Experimental results indicate that our method more effectively addresses long-tail knowledge, delivering more accurate and fluent responses compared to knowledge internalization methods, which necessitate extensive fine-tuning and training on large volumes of private data.

Comparison against knowledge enhancement methods. Considering fairness and persuasiveness, we compared knowledge enhancement methods based on the same size as ours. As shown in Table 2, our SMART performs better on most tasks compared to other knowledge enhancement methods. Compared to the SOTA retrieval method, SelfRag (Asai et al. 2023), our model shows great superiority in both accuracy and fluency. Our method exceeds MMAgent (four independent agents coupled together) in all metrics. This demonstrates that our learning paradigm improves multi-agent collaboration, resulting in more accurate responses. Note that INTERACT (Gao et al. 2023) is better than us on Squad1, the reason is that INTERACT allows the response model to do more reasoning steps, which is beneficial for hitting answers in short-format generation tasks. RAIT (Lin et al. 2023) is trained with SMART same data and initialized model without fact location and intent reconstruction, lagging behind us. Overall, our SMART delivers excellent performance in a diverse range of knowledge-intensive tasks. This result indicates SMART gains are not solely from the multi-agent framework and demonstrate the effectiveness of the long-short trajectory learning.

Ablation Studies

Training ablation of different agents. Training ablation aims to verify the superiority of the entire multi-agent combination setup. To save the experiment cost, we implement long-trajectory learning using 60,000 samples from the long-trajectory subset to evaluate the performance of the co-framework under different agent absence scenarios.

Methods	Health	PopQA	ASQA	
	(Acc)	(Acc)	(Em)	(R-L)
Vanilla LLM	9.80	22.69	14.11	6.45
+ Short	62.00	32.23	23.95	19.91
+ Long	72.9	37.66	39.86	39.51
+ Short & Long	**73.18**	**42.60**	**41.16**	**40.66**

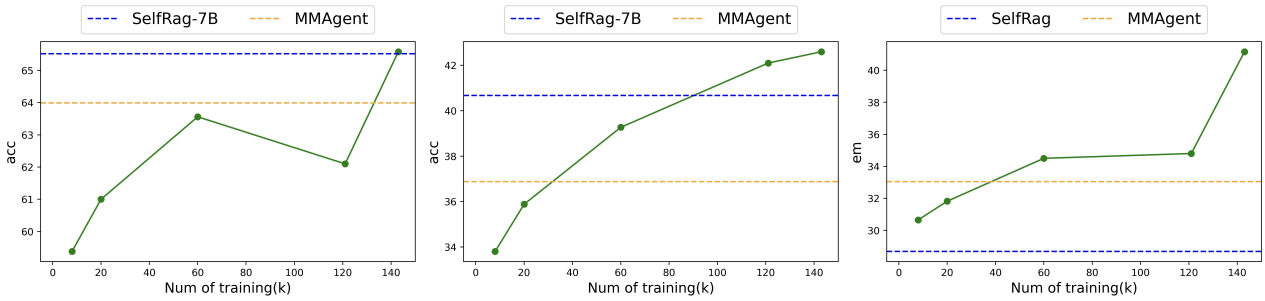
Table 4: Ablation studies of long-trajectory (Long) and short-trajectory (Short) learning.

As the top part of Table 3, the absence of the fact Locator and the intent reconstructor significantly degrades the framework's performance. The intent reconstructor provides substantial benefits for multiple-choice reasoning (ARC-C) and ambiguous questions (ASQA), while the fact Locator is crucial for long-tail knowledge Q&A (PopQA). The experiment proved the effectiveness of different agents in our SMART, especially the fact Locator and the intent reconstructor.

Inference ablation of different agents. We use the full version of SMART with short long-trajectory learning to ignore the trajectories of different agents during the inference phase. As the bottom part of Table 3, each agent plays an important role in the collaboration framework. The effect degradation of the fact-checking task (Health) was not severe, which may be related to the large amount of knowledge injected during the short trajectory learning. In addition, note that if the inference process is missing a particular agent, most multi-agent frameworks that use end-to-end training become terrible, due to the loss of signals from the missing agent. Benefiting from our Short-Trajectory Learning through the trajectory tokens, our SMART does not collapse in performance when an agent is missing, demonstrating flexibility while maintaining performance.

Effects of Long-Short Trajectory Learning. Long-Short Trajectory Learning optimising a Multi-agent framework through two-stage learning. we demonstrate its effectiveness progressively by training it on vanilla models, Llama-2-7B-hf (Touvron et al. 2023). As shown in Table 4, short-trajectory learning and long-trajectory learning enable huge performance improvements in the framework for all tasks. Short-trajectory learning enhances the system by optimizing each agent's base capability, though its impact is not as substantial as that of long-trajectory learning. Long-trajectory learning, by optimizing agent synergy, underscores the importance of collaborative optimization in a multi-agent framework, despite the challenges posed by complex data construction. Overall, the combined approach of long-short trajectory learning yields the best performance, highlighting the significance of simultaneous collaboration and individual uniqueness.

Effects of training data size. To examine the impact of long-trajectory training data on long-short trajectory learning, we randomly selected subsets of 8k, 20k, 60k, and 121k instances from the initial 140k training instances and fine-tuned four SMART variants on these subsets. Subsequently, we compared the model performance on ARC-C, PopQA,



(a) ARC-C (b) PopQA (c) ASQA

Figure 4: Effects of long-trajectory training data size (K) on three tasks, ARC-C, PopQA and ASQA.

and ASQA with our SelfRAG and MMagent models. As shown in Figure 4, an increase in data size generally leads to improved performance across all datasets. Notably, by utilizing 60k data instances, SMART outperformed SelfRAG, which employs 120k samples. This demonstrates the significant advantage of our learning approach in markedly enhancing the performance of multi-agent framework.

Related Work

Trajectory Learning. Trajectory learning aims to allow agent systems to complete a complex task or scenario through a series of interconnected phases, which requires a profound understanding of both global and local dimensions. Some methods (Chen et al. 2023; Song et al. 2023; Kong et al. 2023; Asai et al. 2023; Sun et al. 2022; Mou, Wei, and Huang 2024) enable agent learning trajectory via providing crafted prompt or tuning, which may not consistently yield high performance in every phase. Moreover, independently modules (Liu et al. 2023; Shen et al. 2024; Ma et al. 2023; Xu, Shi, and Choi 2023; Wang et al. 2023) can be combined with agent to implement trajectory inference, while this integration confers robust isolated capabilities, the gap between modules might lead to cumulative errors throughout the trajectory process. In this paper, we introduce long-short trajectory learning, which equips multi-agent systems with the ability to not only grasp the logic connecting steps but also to refine each step. Our approach is scalable to increasingly complex scenarios.

Knowledge Enhancement Methods. Ensuring fact-consistent responses is a core goal of intelligent systems research (Wang et al. 2022b; Tu et al. 2024b,a, 2023; Yue et al. 2024, 2023b; Gao et al. 2024). LLMs parameterize knowledge by training on gargantuan textual corpora. However, LLMs suffer from hallucination (Ji et al. 2023), trouble in acquiring long-tailed fact (Kandpal et al. 2023) and struggle to expand their parametric knowledge. For knowledge-intensive scenarios, existing methods (Izacard et al. 2023; Sun et al. 2020) usually assist LLMs by integrating non-parametric knowledge. Recent advances incorporated retrievers (Asai et al. 2023; Shi et al. 2023; Lin et al. 2023) to augment LLMs. The efficacy of non-parametric knowledge collaboration in improving task performance significantly relies on the relevance of the acquired knowledge and the level of knowledge utilization by the LLM itself. However, existing work has not comprehensively confronted these challenges Some works (Xu, Shi, and Choi 2023; Ma et al. 2023) simply select relevant knowledge and demonstrate better intentions by combining separate modules. Self-RAG (Asai et al. 2023) integrates specialized feedback tokens into the language model to assess the necessity for retrieval and to verify the relevance, support, or completeness of the output. Unlike existing approaches, we introduce a novel multi-agent framework that addresses these challenges with trajectory learning.

Conclusions

In this paper, we introduce SMART, a novel multi-agent framework that addresses the challenges of generating factually consistent responses in knowledge-intensive tasks. By leveraging external knowledge and employing specialized agents, SMART enhances the interpretability and factual consistency of LLMs generated responses. Our proposed Long- and Short-Trajectory Learning paradigm ensures synergistic collaboration among agents while maintaining fine-grained execution, enabling the framework to navigate complex knowledge-intensive tasks effectively. Empirical results on five diverse tasks demonstrate SMART's superior performance compared to SOTA pre-trained and instruction-tuned LLMs, as well as widely adopted methods. SMART highlights the importance of integrating external knowledge and employing multi-agent systems to tackle the limitations of LLMs in knowledge-intensive scenarios.

Future work. One is that our framework currently executes sequentially without iterative optimization, which may lead to insufficient knowledge retrieval for multi-hop problems. However, this can be addressed by adding loop arrows between the Fact Locator and Intent Reconstructor agents. Another is that our retriever is not trained in the whole process, although it can be incorporated into the training process using existing techniques. We envision our framework as a general paradigm that extends beyond knowledge-intensive tasks to more complex scenarios, enabling any multi-agent framework to internalize tailored trajectories.

Acknowledgments

This research is supported by National Key R&D Program of China (2023YFF1204800) and National Natural Science Foundation of China (No.62176058). The project's computational resources are supported by CFFF platform of Fudan University.

References

Achiam, J.; Adler, S.; Agarwal, S.; Ahmad, L.; Akkaya, I.; Aleman, F. L.; Almeida, D.; Altenschmidt, J.; Altman, S.; Anadkat, S.; et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Akhtar, M.; Cocarascu, O.; and Simperl, E. 2022. PubHealthTab: A public health table-based dataset for evidence-based fact checking. In *Findings of the Association for Computational Linguistics: NAACL 2022*, 1–16.

Anantha, R.; Vakulenko, S.; Tu, Z.; Longpre, S.; Pulman, S.; and Chappidi, S. 2021. Open-Domain Question Answering Goes Conversational via Question Rewriting. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 520–534.

Asai, A.; Wu, Z.; Wang, Y.; Sil, A.; and Hajishirzi, H. 2023. Self-rag: Learning to retrieve, generate, and critique through self-reflection. *arXiv preprint arXiv:2310.11511*.

Chen, W.; Ma, X.; Wang, X.; and Cohen, W. W. 2023. Program of Thoughts Prompting: Disentangling Computation from Reasoning for Numerical Reasoning Tasks. *Transactions on Machine Learning Research*.

Clark, P.; Cowhey, I.; Etzioni, O.; Khot, T.; Sabharwal, A.; Schoenick, C.; and Tafjord, O. 2018. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv preprint arXiv:1803.05457*.

De Cao, N.; Aziz, W.; and Titov, I. 2021. Editing Factual Knowledge in Language Models. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 6491–6506.

Dinan, E.; Roller, S.; Shuster, K.; Fan, A.; Auli, M.; and Weston, J. 2018. Wizard of Wikipedia: Knowledge-Powered Conversational Agents. In *International Conference on Learning Representations*.

Gao, L.; Lu, J.; Shao, Z.; Lin, Z.; Yue, S.; Ieong, C.; Sun, Y.; Zauner, R. J.; Wei, Z.; and Chen, S. 2024. Fine-tuned large language model for visualization system: A study on self-regulated learning in education. *IEEE Transactions on Visualization and Computer Graphics*.

Gao, T.; Yen, H.; Yu, J.; and Chen, D. 2023. Enabling Large Language Models to Generate Text with Citations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 6465–6488.

Geva, M.; Khashabi, D.; Segal, E.; Khot, T.; Roth, D.; and Berant, J. 2021. Did aristotle use a laptop? a question answering benchmark with implicit reasoning strategies. *Transactions of the Association for Computational Linguistics*, 9: 346–361.

Hong, S.; Zheng, X.; Chen, J.; Cheng, Y.; Wang, J.; Zhang, C.; Wang, Z.; Yau, S. K. S.; Lin, Z.; Zhou, L.; et al. 2023. Metagpt: Meta programming for multi-agent collaborative framework. *arXiv preprint arXiv:2308.00352*.

Huang, L.; Le Bras, R.; Bhagavatula, C.; and Choi, Y. 2019. Cosmos QA: Machine Reading Comprehension with Contextual Commonsense Reasoning. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Association for Computational Linguistics.

Huang, L.; Yu, W.; Ma, W.; Zhong, W.; Feng, Z.; Wang, H.; Chen, Q.; Peng, W.; Feng, X.; Qin, B.; et al. 2023. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *arXiv preprint arXiv:2311.05232*.

Izacard, G.; Caron, M.; Hosseini, L.; Riedel, S.; Bojanowski, P.; Joulin, A.; and Grave, E. 2021. Unsupervised dense information retrieval with contrastive learning. *arXiv preprint arXiv:2112.09118*.

Izacard, G.; Lewis, P.; Lomeli, M.; Hosseini, L.; Petroni, F.; Schick, T.; Dwivedi-Yu, J.; Joulin, A.; Riedel, S.; and Grave, E. 2023. Atlas: Few-shot learning with retrieval augmented language models. *Journal of Machine Learning Research*, 24(251): 1–43.

Ji, Z.; Lee, N.; Frieske, R.; Yu, T.; Su, D.; Xu, Y.; Ishii, E.; Bang, Y. J.; Madotto, A.; and Fung, P. 2023. Survey of hallucination in natural language generation. *ACM Computing Surveys*, 55(12): 1–38.

Jiang, A. Q.; Sablayrolles, A.; Mensch, A.; Bamford, C.; Chaplot, D. S.; Casas, D. d. l.; Bressand, F.; Lengyel, G.; Lample, G.; Saulnier, L.; et al. 2023. Mistral 7B. *arXiv preprint arXiv:2310.06825*.

Kandpal, N.; Deng, H.; Roberts, A.; Wallace, E.; and Raffel, C. 2023. Large language models struggle to learn long-tail knowledge. In *International Conference on Machine Learning*, 15696–15707. PMLR.

Kong, Y.; Ruan, J.; Chen, Y.; Zhang, B.; Bao, T.; Shi, S.; Du, G.; Hu, X.; Mao, H.; Li, Z.; et al. 2023. Tptu-v2: Boosting task planning and tool usage of large language model-based agents in real-world systems. *arXiv preprint arXiv:2311.11315*.

Kwiatkowski, T.; Palomaki, J.; Redfield, O.; Collins, M.; Parikh, A.; Alberti, C.; Epstein, D.; Polosukhin, I.; Devlin, J.; Lee, K.; et al. 2019. Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7: 453–466.

Lin, X. V.; Chen, X.; Chen, M.; Shi, W.; Lomeli, M.; James, R.; Rodriguez, P.; Kahn, J.; Szilvasy, G.; Lewis, M.; et al. 2023. Ra-dit: Retrieval-augmented dual instruction tuning. *arXiv preprint arXiv:2310.01352*.

Liu, B.; Jiang, Y.; Zhang, X.; Liu, Q.; Zhang, S.; Biswas, J.; and Stone, P. 2023. Llm+ p: Empowering large language models with optimal planning proficiency. *arXiv preprint arXiv:2304.11477*.

Ma, X.; Gong, Y.; He, P.; Zhao, H.; and Duan, N. 2023. Query Rewriting in Retrieval-Augmented Large Language

Models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 5303–5315.

Mallen, A.; Asai, A.; Zhong, V.; Das, R.; Khashabi, D.; and Hajishirzi, H. 2022. When not to trust language models: Investigating effectiveness of parametric and non-parametric memories. *arXiv preprint arXiv:2212.10511*.

Mihaylov, T.; Clark, P.; Khot, T.; and Sabharwal, A. 2018. Can a suit of armor conduct electricity? a new dataset for open book question answering. *arXiv preprint arXiv:1809.02789*.

Mou, X.; Wei, Z.; and Huang, X. 2024. Unveiling the Truth and Facilitating Change: Towards Agent-based Large-scale Social Movement Simulation. *arXiv preprint arXiv:2402.16333*.

Ouyang, L.; Wu, J.; Jiang, X.; Almeida, D.; Wainwright, C.; Mishkin, P.; Zhang, C.; Agarwal, S.; Slama, K.; Ray, A.; et al. 2022. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35: 27730–27744.

Rajpurkar, P.; Zhang, J.; Lopyrev, K.; and Liang, P. 2016. Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250*.

Shen, W.; Li, C.; Chen, H.; Yan, M.; Quan, X.; Chen, H.; Zhang, J.; and Huang, F. 2024. Small llms are weak tool learners: A multi-llm agent. *arXiv preprint arXiv:2401.07324*.

Shi, W.; Min, S.; Yasunaga, M.; Seo, M.; James, R.; Lewis, M.; Zettlemoyer, L.; and Yih, W.-t. 2023. Replug: Retrieval-augmented black-box language models. *arXiv preprint arXiv:2301.12652*.

Singhal, K.; Azizi, S.; Tu, T.; Mahdavi, S. S.; Wei, J.; Chung, H. W.; Scales, N.; Tanwani, A.; Cole-Lewis, H.; Pfohl, S.; et al. 2022. Large language models encode clinical knowledge. *arXiv preprint arXiv:2212.13138*.

Song, C. H.; Wu, J.; Washington, C.; Sadler, B. M.; Chao, W.-L.; and Su, Y. 2023. Llm-planner: Few-shot grounded planning for embodied agents with large language models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2998–3009.

Stelmakh, I.; Luan, Y.; Dhingra, B.; and Chang, M.-W. 2022. ASQA: Factoid Questions Meet Long-Form Answers. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, 8273–8288.

Sun, T.; Shao, Y.; Qian, H.; Huang, X.; and Qiu, X. 2022. Black-box tuning for language-model-as-a-service. In *International Conference on Machine Learning*, 20841–20855. PMLR.

Sun, T.; Shao, Y.; Qiu, X.; Guo, Q.; Hu, Y.; Huang, X.-J.; and Zhang, Z. 2020. CoLAKE: Contextualized Language and Knowledge Embedding. In *Proceedings of the 28th International Conference on Computational Linguistics*, 3660–3670.

Thorne, J.; Vlachos, A.; Christodoulopoulos, C.; and Mittal, A. 2018. FEVER: a Large-scale Dataset for Fact Extraction and VERification. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, 809–819.

Touvron, H.; Martin, L.; Stone, K.; Albert, P.; Almahairi, A.; Babaei, Y.; Bashlykov, N.; Batra, S.; Bhargava, P.; Bhosale, S.; et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Tu, Y.; Li, L.; Su, L.; Yan, C.; and Huang, Q. 2024a. Distractors-Immune Representation Learning with Cross-modal Contrastive Regularization for Change Captioning. In *ECCV*, 311–328.

Tu, Y.; Li, L.; Su, L.; Zha, Z.-J.; Yan, C.; and Huang, Q. 2023. Self-supervised cross-view representation reconstruction for change captioning. In *ICCV*, 2805–2815.

Tu, Y.; Li, L.; Su, L.; Zha, Z.-J.; Yan, C.; and Huang, Q. 2024b. Context-aware Difference Distilling for Multi-change Captioning. In *ACL*, 7941–7956.

Wang, S.; Wei, Z.; Fan, Z.; Zhang, Q.; and Huang, X.-J. 2022a. Locate Then Ask: Interpretable Stepwise Reasoning for Multi-hop Question Answering. In *Proceedings of the 29th International Conference on Computational Linguistics*, 1655–1665.

Wang, S.; Zhong, W.; Tang, D.; Wei, Z.; Fan, Z.; Jiang, D.; Zhou, M.; and Duan, N. 2022b. Logic-Driven Context Extension and Data Augmentation for Logical Reasoning of Text. In *Findings of the Association for Computational Linguistics: ACL 2022*, 1619–1629.

Wang, Z.; Araki, J.; Jiang, Z.; Parvez, M. R.; and Neubig, G. 2023. Learning to filter context for retrieval-augmented generation. *arXiv preprint arXiv:2311.08377*.

Xu, F.; Shi, W.; and Choi, E. 2023. Recomp: Improving retrieval-augmented lms with compression and selective augmentation. *arXiv preprint arXiv:2310.04408*.

Yao, S.; Zhao, J.; Yu, D.; Du, N.; Shafran, I.; Narasimhan, K.; and Cao, Y. 2023. ReAct: Synergizing Reasoning and Acting in Language Models. In *International Conference on Learning Representations (ICLR)*.

Yue, S.; Chen, W.; Wang, S.; Li, B.; Shen, C.; Liu, S.; Zhou, Y.; Xiao, Y.; Yun, S.; Huang, X.; et al. 2023a. Disc-lawllm: Fine-tuning large language models for intelligent legal services. *arXiv preprint arXiv:2309.11325*.

Yue, S.; Liu, S.; Zhou, Y.; Shen, C.; Wang, S.; Xiao, Y.; Li, B.; Song, Y.; Shen, X.; Chen, W.; et al. 2024. LawLLM: Intelligent Legal System with Legal Reasoning and Verifiable Retrieval. In *International Conference on Database Systems for Advanced Applications*, 304–321. Springer.

Yue, S.; Tu, Y.; Li, L.; Yang, Y.; Gao, S.; and Yu, Z. 2023b. I3n: Intra-and inter-representation interaction network for change captioning. *IEEE Transactions on Multimedia*, 25: 8828–8841.

Zheng, L.; Chiang, W.-L.; Sheng, Y.; Zhuang, S.; Wu, Z.; Zhuang, Y.; Lin, Z.; Li, Z.; Li, D.; Xing, E.; et al. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in Neural Information Processing Systems*, 36: 46595–46623.