Reflective Multi-Agent Collaboration based on Large Language Models

Xiaohe Bo[1], **Zeyu Zhang**[1], **Quanyu Dai**[2], **Xueyang Feng**[1],
Lei Wang[1], **Rui Li**[1], **Xu Chen**[1]*, **Ji-Rong Wen**[1]
[1] Gaoling School of Artificial Intelligence, Renmin University of China
[2] Huawei Noah's Ark Lab
`{xiaohe,zeyuzhang,xueyangfeng,wanglei154,lirui121200,xu.chen,jrwen}@ruc.edu.cn`,
`daiquanyu@huawei.com`

Abstract

Benefiting from the powerful language expression and planning capabilities of Large Language Models (LLMs), LLM-based autonomous agents have achieved promising performance in various downstream tasks. Recently, based on the development of single-agent systems, researchers propose to construct LLM-based multi-agent systems to tackle more complicated tasks. In this paper, we propose a novel framework, named **COPPER**, to enhance the collaborative capabilities of LLM-based agents with the self-reflection mechanism. To improve the quality of reflections, we propose to fine-tune a shared reflector, which automatically tunes the prompts of actor models using our counterfactual PPO mechanism. On the one hand, we propose counterfactual rewards to assess the contribution of a single agent's reflection within the system, alleviating the credit assignment problem. On the other hand, we propose to train a shared reflector, which enables the reflector to generate personalized reflections according to agent roles, while reducing the computational resource requirements and improving training stability. We conduct experiments on three datasets to evaluate the performance of our model in multi-hop question answering, mathematics, and chess scenarios. Experimental results show that COPPER possesses stronger reflection capabilities and exhibits excellent generalization performance across different actor models.

1 Introduction

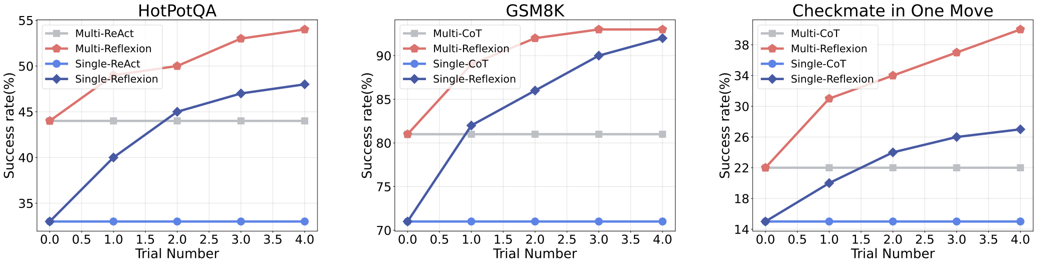
With the emergence of Large Language Models, LLM-based autonomous agents are becoming a research hotspot in the field of artificial intelligence. Leveraging the impressive planning and reasoning ability of LLMs, these agents can understand and generate human-like instructions, engage in sophisticated interactions, and make decisions in a wide range of contexts, leading to remarkable success in various downstream tasks [26, 24, 25, 5, 22]. Recently, based on the development of single-agent systems, researchers propose to construct multi-agent systems in response to the growing task complexity. Prior works [31, 12, 8] suggest that multiple agents can help improve factuality and reasoning, encourage divergent thinking, and effectively facilitate task completion.

To improve the collaborative performance of multi-agent systems, various cooperation frameworks [31, 4, 23, 39] have been developed, which generally encode intricately crafted agent profiles and cooperation mechanisms into prompts. However, hindered by the contextual understanding ability of LLMs, such frameworks fall short of fully exploiting the collaborative capacities of agents. To tackle this challenge, one natural idea is to gather extensive collaborative data for agents' fine-tuning. Yet this strategy risks diminishing the model's general abilities [35], contradicting the aspiration to

---

[*] Corresponding Author.

38th Conference on Neural Information Processing Systems (NeurIPS 2024).



(a) HotpotQA. (b) GSM8K. (c) Checkmate in One Move.

Figure 1: Performance comparison between single-agent and multi-agent systems on three datasets. We use GPT-3.5 as the base model and prompts are shown in the Appendix H.

attain artificial general intelligence (AGI). Considering that, in this paper, we propose to optimize the collaboration process through the self-reflection mechanism [27], with which binary or scalar rewards from environments can be converted into verbal reflections, providing additional context to improve task performance. To accommodate the characteristics of multi-agent systems, we additionally incorporate agent profiles in reflector prompts for agent-specific reflections and adopt a fully observable setting to facilitate agents' error detection.

Although the self-reflection mechanism enables iterative refinement, deriving useful reflections from a pre-trained, frozen LLM is challenging. In dealing with that, Retroformer [38] suggests approximating reflection rewards with the difference between two consecutive episode returns and training a plug-in reflector through policy optimization. However, extending the method to multi-agent systems is not straightforward. On the one hand, the episode difference score from the environment can only capture the overall contribution from all agents' reflections, while the credit to each agent's reflection is unknown and nontrivial to obtain. Employing the overall score directly and uniformly for reflector training of multiple different agents could lead to lazy reflectors, and thus the credit assignment problem becomes critical. On the other hand, to achieve personalized reflections of intelligent agents, the number of reflectors to be fine-tuned will expand proportionally with the number of agents in the system, posing challenges for practical applications in real-world scenarios.

To address the challenges above, we propose a reflective framework: **CO**unterfactual **PP**O **E**nhanced Shared **R**eflector for LLM-based Multi-Agent Collaboration, named **COPPER**. For the first challenge, we propose counterfactual rewards as supervision signals for individual agent reflections. Specifically, we first integrate the reflections of all agents into the corresponding actor model prompts and utilize the episode return difference score as the *overall reward*. Then we sequentially marginalize out the reflection of one agent, repeat the interaction process, and attain a new task score. The resulting episode return difference is referred to as *marginal reward*. The *counterfactual reward* is then calculated by subtracting the marginal reward from the overall reward to reflect the contribution of the removed reflection. For the second challenge, considering the homogeneity between different reflectors, which means their action space (reflection) is consistent and the optimization objectives are aligned (to assist in solving the overall task), we propose to train a shared reflector for agents in the collaboration system. With carefully designed prompts, the shared reflector can grasp the role information of each agent, simultaneously reducing the demand for computational resources and enriching the training data pool for stable training. The counterfactual reflection data from all agents are then collected and utilized to train a shared reflector through proximal policy optimization (PPO).

Our contributions can be summarized as follows:

- We propose a novel reflection framework, named COPPER, to improve the multi-agent collaboration. We incorporate agent profiles into reflector prompts for agent-specific reflections and adopt a fully observable setting to assist in error detection.

- We propose to train a shared reflector using our counterfactual PPO mechanism. To alleviate the credit assignment problem in multi-agent systems, we design counterfactual rewards to rate each agent's reflection. Besides, we propose to train a shared reflector, which could generate personalized reflections while reducing computational resource demands and improving training stability.

- Experimental results on three open-source datasets demonstrate that COPPER possesses stronger reflection capabilities against baselines. More concretely, compared to the initial success rate, COPPER brought improvements of 31.8%, 18.5%, and 86.4% on the HotPotQA, GSM8K, and Checkmate in One Move datasets, respectively.

2 Related Work

2.1 LLM-based Multi-Agent Systems

Based on the development of single-agent systems, LLM-based multi-agent systems have been rapidly studied and achieved significant progress in complex task resolution and world simulation. Within the task resolution domain, various agents, each with specialized expertise are developed to collaborate on complex problems. For instance, [9, 32, 3, 29, 17] suggest improving the accuracy of scientific question-answering tasks through multi-agent debates. [23, 13] suggest constructing multi-agent systems for software development following the waterfall or Standardized Operating Procedures (SOPs) workflow. Another mainstream application scenario of LLM-based multi-agent systems is the world simulation, which mainly leverages the role-playing abilities of agents to represent different roles and perspectives within a simulated environment. Research in this area is advancing quickly and encompasses a wide variety of fields, including social sciences [42, 11], gaming [33, 34, 18, 20, 1], psychology [2], economics [16, 41], policy making [15], etc. In this paper, we focus on improving the complex problem-solving abilities of multi-agent systems.

2.2 Self-Reflection of Large Language Models

Various studies on reflection mechanisms have been proposed, which play a crucial role in enabling LLM-based agents to learn from the environment and improve themselves autonomously. Early works primarily focus on refining responses based on a single feedback [19, 6] or contrast between multiple models [10, 40] and fail to form a comprehensive understanding of the task based on past experiences. Recently, Reflexion [27] involves prior trajectories and environmental rewards to generate reflections, which are further incorporated into the context of subsequent episodes. Although it enables iterative enhancements, the effectiveness of reflections heavily relies on the model's inherent reflective capabilities. In light of that, Retroformer [38] proposes to fine-tune the reflector using environmental rewards with a standard RLHF [21] process. However, optimizing reflection in multi-agent systems remains challenging, which is crucial for improving agents' cooperation capacity and task performance.

3 Preliminary

In this paper, we use a tuple $(N, \mathcal{S}, \mathcal{A}, \mathcal{P}_{\xi_o}, \mathcal{R})$ to denote the LLM-based multi-agent cooperation system, where N stands for the number of agents, $\mathcal{S} = S_1 \times S_2 \times \cdots \times S_N$ is the joint space of environment states, $\mathcal{A} = A_1 \times A_2 \times \cdots \times A_N$ is the joint action space and $\mathcal{P}_{\xi_o} : \mathcal{S} \times \mathcal{A} \rightarrow \mathcal{S}$ is the state transition function. Here, we denote the randomness associated with the state transition using ξ_o according to [38]. In cooperative settings, all agents share an aligning goal, and the reward function $\mathcal{R} : (\mathcal{S}, \mathcal{A}) \rightarrow \mathbb{R}$ is typically designed to promote collaboration. One major challenge in cooperation settings is credit assignment, which means we need to decompose $\mathcal{R}$ into $R_1 \times R_2 \times ... \times R_N$ and evaluate the agent contribution with respect to their objective in the form of a scalar value. The multi-agent systems complete the target task through interactions with the environment. Here we use trajectory $\tau = \{s_0, a_0, s_1, a_1, \cdots, s_T, a_T\}$ to denote the process and describe the accumulative reward using $R(\tau)$, where $R(\tau) = \sum_{t=0}^{T} \mathcal{R}(s_t, a_t)$ and T is the length of the trajectory. In most of the situations, rewards from the environment are sparse, which means $\mathcal{R}(s_t, a_t)$ are mostly zero except very few states, such as the terminal state for indicating task success or failure.

Specifically, for each agent i, we consider its actor model as a function $\mathcal{M}^i_{\xi_l} : \mathcal{X}_i \rightarrow A_i$, where $\mathcal{X}_i$ is the space of the prompts and ξ_l represents the random variables involved in the sampling process. To maintain the general abilities of agents, in this paper, we select LLMs with frozen parameters such as ChatGPT and GPT-4 as actor models. Current environment states are incorporated into prompts in the form of natural language and actions are selected based on the contextual learning ability of LLMs. Meanwhile, we propose to improve collaboration through self-reflection and introduce a reflector model $\mathcal{M}^i_{\xi_r,\theta} : (\mathcal{T}, \mathbb{R}) \rightarrow \mathcal{X}_i$ for each agent, where $\mathcal{T}$ denotes the space of trajectories, ξ_r represents the randomness in the reflector model and θ denotes the learnable parameters. The reflector model takes the prior trajectory and the reward signal from the environment as input, outputting reflections to refine the prompt of the corresponding actor model. We fine-tune reflector models through policy optimization for better task performance in specific environments.

4 Method

4.1 Multi-Agent Collaboration

LLM-based multi-agent systems aim to deliver advanced capabilities by leveraging collective intelligence and specializing LLMs into agents with distinct capabilities. In this paper, each agent in the system operates in a predetermined sequence, taking turns to produce responses, while a shared message pool is maintained to facilitate efficient communication. We illustrate the details of the collaboration settings in Appendix A.

Specifically, in dealing with problem k at time t, agent i ($i = t \mod N$) first subscribes the preceding interaction records $[s_{k,i}, a_{k,i}]_{i=0}^{t-1}$ from the message pool, and then acquires the current environment state $s_{k,t}$. The decision-making process can be expressed as:

$$a_{k,t} = \text{Actor}^i(p^i, [s_{k,i}, a_{k,i}]_{i=0}^{t-1}, s_{k,t}), \tag{1}$$

where p^i is the profile of the current agent, which encompasses its role, action space, and additional constraints in the form of natural language. Once reaching a decision, the agent publishes a new message $\{s_{k,t}, a_{k,t}\}$ to the message pool.

However, during implementation, the interaction history could potentially exceed the token limit of LLMs, given the large number of agents and decision steps. To address this challenge, we introduce a context model to recursively update the interaction history from each agent's perspective, serving as its short-term memory. New messages since the agent's last action and the profile will be integrated to form a new short-term memory based on the previous one. The process can be written as:

$$sm^i_{k,t} = \text{Context}^i(p^i, sm^i_{k,t-1}, \{s_i, a_i\}^t_{i=\max(0,t-N+1)}), \tag{2}$$

where $sm^i_{k,t}$ represent the short-term memory of agent i when solving problem k at time t.

We then replace the interaction history with the agent's short-term memory. Therefore, the decision-making process can be further rewritten as:

$$a_{k,t} = \text{Actor}^i(p^i, sm^i_{k,t}, s_{k,t}). \tag{3}$$

4.2 Multi-Agent Reflection Framework

To bolster the collaborative performance of multi-agent systems in specific scenarios, while preserving the general capabilities of agents, we introduce a self-reflection mechanism to multi-agent systems, of which the details are shown on the left part of Figure 2. Using environmental rewards as guidance, the generated reflections could act as semantic gradient signals by providing a concrete direction for improvement, thereby helping the agent learn from prior errors and perform better on the task.

Different from reflections in single-agent systems, we integrate agent profiles into the multi-agent reflection process to obtain role-specific reflections, and take a fully observable setting to assist the agent in error detection by offering interaction histories from each agent's perspective. The reflection process of the agent can be defined as:

$$y^i_{k,\lambda} = \text{Reflector}^i(p^i, [sm^i_{k,\lambda,T}]^N_{i=1}, r_{k,\lambda}), \tag{4}$$

where k represents the problem, λ indicates λ-th trial of answer to question k, T is the length of the trajectory $\tau_{k,\lambda}$ and $r_{k,\lambda}$ is the environmental rewards. Due to the iterative updating nature of the short-term memory, $sm^i_{k,\lambda,T}$ contains the complete action information of agent i in trajectory $\tau_{k,\lambda}$.

We store all previous reflections of agent i in its long-term memory, which are then added as additional context of the actor model. The decision-making process in Equation 3 can be further defined as:

$$a_{k,\lambda,t} = \text{Actor}^i(p^i, lm^i_{k,\lambda}, sm^i_{k,\lambda,t}, s_{k,\lambda,t}), \tag{5}$$

where we additionally incorporate subscript λ due to the introduction of the reflection mechanism.

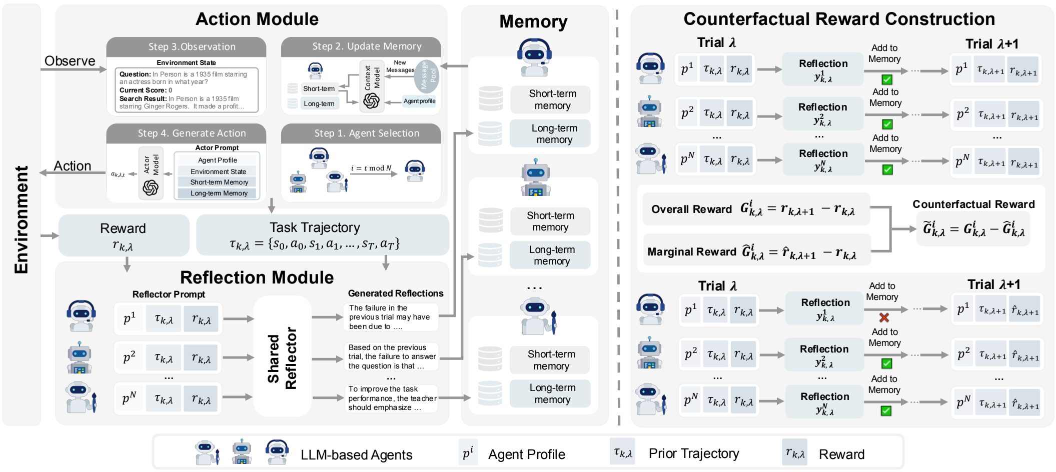

Environment
Observe
Action
Action Module
Step 3.Observation
Environment State
Question: In Person is a 1935 film starring an actress born in what year?
Current Score: 0
Search Result: In Person is a 1935 film starring Ginger Rogers. It made a profit…
Step 2. Update Memory
Short-term
Long-term
Context Model
New Messages
Message Pool
Agent profile
Step 4. Generate Action
$a_{k,\lambda,t}$
Actor Model
Actor Prompt
Agent Profile
Environment State
Short-term Memory
Long-term Memory
Step 1. Agent Selection
$i = t \bmod N$
Reward
$r_{k,\lambda}$
Task Trajectory
$\tau_{k,\lambda} = \{s_0, a_0, s_1, a_1, \ldots, s_T, a_T\}$
Reflection Module
Reflector Prompt
p^1 $\tau_{k,\lambda}$ $r_{k,\lambda}$
p^2 $\tau_{k,\lambda}$ $r_{k,\lambda}$
p^N $\tau_{k,\lambda}$ $r_{k,\lambda}$
Shared Reflector
Generated Reflections
The failure in the previous trial may have been due to ….
Based on the previous trial, the failure to answer the question is that …
To improve the task performance, the teacher should emphasize …
Memory
Short-term memory
Long-term memory
Counterfactual Reward Construction
Trial λ
Trial λ+1
Reflection $y^1_{k,\lambda}$
Reflection $y^2_{k,\lambda}$
Reflection $y^N_{k,\lambda}$
Add to Memory
p^1 $\tau_{k,\lambda+1}$ $r_{k,\lambda+1}$
p^2 $\tau_{k,\lambda+1}$ $r_{k,\lambda+1}$
p^N $\tau_{k,\lambda+1}$ $r_{k,\lambda+1}$
Overall Reward $G^i_{k,\lambda} = r_{k,\lambda+1} - r_{k,\lambda}$
Marginal Reward $\hat{G}^i_{k,\lambda} = \hat{r}_{k,\lambda+1} - r_{k,\lambda}$
Counterfactual Reward
$\tilde{G}^i_{k,\lambda} = G^i_{k,\lambda} - \hat{G}^i_{k,\lambda}$
p^1 $\tau_{k,\lambda+1}$ $\hat{r}_{k,\lambda+1}$
p^2 $\tau_{k,\lambda+1}$ $\hat{r}_{k,\lambda+1}$
p^N $\tau_{k,\lambda+1}$ $\hat{r}_{k,\lambda+1}$
LLM-based Agents
p^i Agent Profile
$\tau_{k,\lambda}$ Prior Trajectory
$r_{k,\lambda}$ Reward

Figure 2: The overview of our proposed COPPER. The left side illustrates the multi-agent reflection framework. The system first computes the identifier i of the agent to respond at the current time (Step 1). Then, agent i updates its memory, including reflections of previous trials and the current trial's historical interactions (Step 2), perceives the environmental state such as the question and current task scores (Step 3), and generates the action (Step 4). After several rounds of interaction, the task trajectory and the reward score are fed into reflectors along with agent profiles to generate reflections, which are then stored in long-term memories and serve as additional context for the continuous optimization of actor prompts. On the right side, we depict the construction of counterfactual rewards, which are further employed for fine-tuning the shared reflector.

4.3 Optimization of the Shared Reflector

Generating useful reflective feedback with frozen LLMs in multi-agent systems proves to be challenging, since it demands a profound grasp of agent characteristics and collaborative environments. Hence, in this paper, we propose to fine-tune a shared reflector using open-source LLMs (such as Llama) with our counterfactual enhanced proximal policy optimization mechanism.

4.3.1 Instruction and Response Collection

In the episode λ of problem k, the multi-agent system first interacts with the environment to produce a trajectory $\tau_{k,\lambda}$, after which the reward function returns a score $r_{k,\lambda}$. Agents in the system then reflect on the prior failed trajectory and generate verbal feedback to refine the corresponding actor prompt. In the process, Reflector^i takes $\{p^i, [sm^i_{k,\lambda,T}]^N_{i=1}, r_{k,\lambda}\}$ as the instruction $x^i_{k,\lambda}$ and is prompted to produce a reflection response $y^i_{k,\lambda}$. Considering the homogeneity between agent reflectors as well as the training efficiency, we gather reflection data from all agents across tasks and trials to train a shared reflector. The offline training data D can be defined as follows:

$$D = \{(x^i_{k,\lambda}, y^i_{k,\lambda}) | 1 \le i \le N, 1 \le \lambda \le \Lambda, 1 \le k \le K\}, \tag{6}$$

where Λ is the maximum trial count and K is the total number of problems.

4.3.2 Counterfactual Reward

In this paper, to alleviate the credit assignment issue, we propose the counterfactual reward to achieve agent-specific reflection ratings for multi-agent collaboration. The construction of counterfactual rewards is shown on the right side of Figure 2.

Specifically, we first calculate an overall reward of the multi-agent system $G^i_{k,\lambda}$ following Retroformer [38], i.e, $G^i_{k,\lambda} = r_{k,\lambda+1} - r_{k,\lambda}$. Then, we sequentially marginalize out a piece of reflection from agent i (which means we do not add the reflection to the actor model's prompt in the subsequent trial), while keeping other agents' reflections fixed. A new reward score $\hat{r}_{k,\lambda+1}$ is then returned after an interaction trajectory, based on which we calculate a marginal reward $\hat{G}^i_{k,\lambda} = \hat{r}_{k,\lambda+1} - r_{k,\lambda}$. Finally, the counterfactual reward of a reflection pair $(x^i_{k,\lambda}, y^i_{k,\lambda})$ is calculated by subtracting the marginal reward from the overall reward:

$$\tilde{G}^i_{k,\lambda} = G^i_{k,\lambda} - \hat{G}^i_{k,\lambda}. \tag{7}$$

Our counterfactual dataset D_{CF} can be further denoted as:

$$D_{CF} = \{(x^i_{k,\lambda}, y^i_{k,\lambda}, \tilde{G}^i_{k,\lambda}) | 1 \leq i \leq N, 1 \leq \lambda \leq \Lambda, 1 \leq k \leq K\}. \tag{8}$$

4.3.3 Counterfactual Proximal Policy Optimization

Following previous works that tackle Reinforcement Learning from Human Feedback (RLHF) [21], we adopt a similar three-step approach to fine-tune the shared reflector with counterfactual rewards.

For the first step, we take the reflections with positive scores as demonstration data and train a supervised reflector π^{SFT} with Supervised Fine-Tuning (SFT), which can be written as:

$$\mathcal{L}_{SFT}(\boldsymbol{\theta}) = -\mathbb{E}_{(x,y)\sim D_{CF}}[\sum_{k=1}^{m} \log \pi_\theta(y_k | x, y_{<k})], \tag{9}$$

where x is the reflection prompt, and y represents the generated reflection.

For the second step, taking construction expenses into account, instead of collecting pairwise responses for each input, we train a regression model to assess prompt and reflection pairs. We optimize the reward model R_{CF_ϕ} with counterfactual dataset D_{CF} by minimizing the Mean Square Error (MSE) loss:

$$\mathcal{L}_{RM}(\boldsymbol{\phi}) = \mathbb{E}_{(x,y,r)\sim D_{CF}}[(R_{CF_\phi}(x,y) - r)^2]. \tag{10}$$

For the third step, we utilize the counterfactual reward model to optimize the supervised reflector via PPO. We begin by initializing π^{SFT}, which is used to produce predictions $\hat{y}$ for randomly chosen samples x from the entire dataset D_{CF}. Subsequently, the counterfactual reward model R_{CF_ϕ} assigns a reward to each response. Our goal is to optimize the reflector model by maximizing the total reward, which can be accomplished by minimizing the following loss objective:

$$\mathcal{L}_{PPO}(\boldsymbol{\theta}) = -\mathbb{E}_{x\sim D_{CF}} \mathbb{E}_{y\sim \pi^{RL}_\theta(x)}[R_{CF_\phi}(x,y) - \beta \log \frac{\pi^{RL}_\theta(y|x)}{\pi^{SFT}(y|x)}]. \tag{11}$$

5 Experiments

5.1 Datasets

We choose HotPotQA [36], GSM8K [7], and Checkmate in One Move [28] to evaluate the collaborative abilities of multi-agent systems in multi-hop question answering, mathematics and chess.

HotPotQA HotPotQA is a multi-hop question-answering dataset designed to evaluate models' complex reasoning ability. It contains 90,447 question-answer pairs that generally require multiple reasoning steps across documents to arrive at an answer.

GSM8K GSM8K is a collection of 8.5K diverse and high-quality math word problems for grade school students. Each problem requires between 2 to 8 steps to solve, with solutions mainly involving a series of fundamental calculations with basic arithmetic operations.

Checkmate in One Move Checkmate in One Move is a dataset from The Beyond the Imitation Game Benchmark (BIG-bench), featuring 3,500 games to assess language models' proficiency in playing chess using standard algebraic notation (SAN). When presented with a move sequence leading to a potential checkmate, the model is tasked with identifying the move that achieves checkmate.

5.2 Baselines

We compare the following baseline models to verify the effectiveness of COPPER: 1) **CoT** [30]. CoT suggests bridging the gap between question and answer by generating intermediate reasoning and is useful for simple questions without tool needs. We adopt CoT in math and chess environments following [8] to represent the initial success rate of the system. 2) **ReAct** [37]. This is the state-of-the-art frozen language agent architecture, which mainly relies on the reasoning and planning ability of LLMs. It serves as a baseline in HotPotQA to denote how the agent performs without using environmental feedback. 3) **Reflexion** [27]. This is a classic framework to learn from environment signals and generate verbal feedback to improve task performance. We extend the method to multi-agent systems and respectively employ GPT-3.5 and LongChat as reflectors to reflect on multi-agent ReAct or CoT trajectories, without fine-tuning the reflectors. 4) **Retroformer** [38]. The paper proposes an effective method for enhancing the reflective capability of agents in single-agent systems. Here, we treat the agents in a multi-agent environment as mutually independent and fine-tune the reflector of each agent following Retroformer as a baseline.

5.3 Implementation Details

Model We use GPT-3.5 (model: gpt-3.5-turbo) as the frozen actor models as well as the context models of agents and fine-tune LongChat (model: longchat-7b-16k) as the shared reflector. We choose gpt-2 as the regression reward model for counterfactual PPO training.

Collaboration Settings We adopt a cooperative debate paradigm on GSM8K and Checkmate in One Move following [8], while on HotPotQA, in alignment with [27, 38], we design a teacher-student paradigm to enable agents to call the retrieval tool.

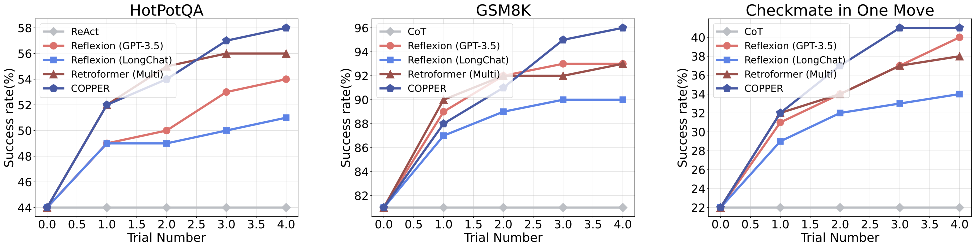
Data Collection We randomly select 2,000 tasks to collect reflection data on HotPotQA and Checkmate in One Move, while on the GSM8K dataset, due to the higher initial success rate and fewer reflections, we randomly select 3,000 instances. We set the maximum number of trials to 5, the temperature of GPT-3.5 to 0, and the temperature of LongChat to 0.9. We use the F1 score as the reward function of HotPotQA following [38] and exact match score in other environments. Comprehensive details regarding the quantity of collected datasets can be found in Appendix B.

Training We use LoRA [14] for efficient fine-tuning of the shared reflector and implement RLHF through the trl package of HuggingFace. For SFT training, we tune the epoch in $\{1, 2, 3, 4\}$, batch size in $\{64, 128, 256\}$, and learning rate in $\{1e\text{-}4, 2e\text{-}4, 3e\text{-}4, 5e\text{-}4\}$ through grid search on a validation set with 100 instances, while for counterfactual PPO, we change the search range of learning rate to $\{1e\text{-}5, 2e\text{-}5, 3e\text{-}5, 5e\text{-}5\}$. As for the reward model, we set learning rate to 5e-5, training epoch to 3 and batch size to 16. We conduct all experiments on four NVIDIA A800-80G GPUs.

Evaluation In alignment with constraints imposed by computational resources and following precedents set by earlier research [38, 27], we randomly sample 100 instances as the test set. We set the temperature of both GPT-3.5 and LongChat to 0 during the test phase to ensure reproducibility. We measure the performance of the system in exact match accuracy during the test phase.

5.4 Main Results

We compare the performance of COPPER against different baselines on HotPotQA, GSM8K, and Checkmate in One Move after 5 trials as main results, which are shown in Figure 3. By observing the results, we find that the results of different methods on the three datasets show roughly the same pattern: (1) Contrasted with the outcomes of multi-agent ReAct or CoT, employing the multi-agent reflection framework outlined in Section 4.2 can notably enhance the performance of multi-agent systems in specific tasks. For instance, in HotPotQA environment, the inclusion of LongChat and GPT-3.5 as reflectors leads to improvements of 15.9% and 22.7%, respectively, over the initial success rate. (2) Compared to the original LongChat and GPT-3.5, COPPER demonstrates stronger reflective abilities. The fine-tuned reflector is proficient in identifying the cause of task failure and devising personalized improvement strategies for diverse intelligent agents. Compared to the initial success rate, COPPER brought improvements of 31.8%, 18.5%, and 86.4% on the HotPotQA, GSM8K, and Checkmate in One Move datasets, respectively. (3) Compared to Retroformer, COPPER can improve the performance of multi-agent collaboration faster. We speculate that the improved performance is brought by our special designs for multi-agent settings, such as the counterfactual reward and the shared reflector.

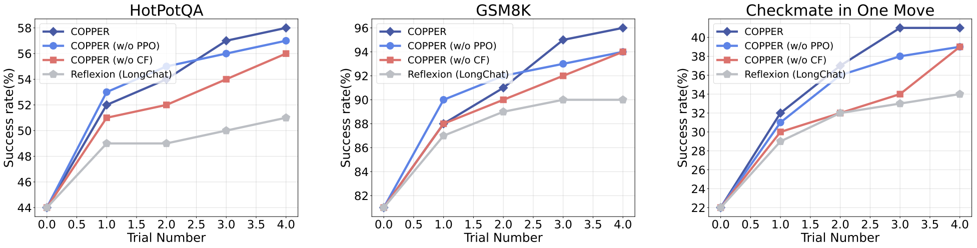


(a) HotpotQA. (b) GSM8K. (c) Checkmate in One Move.

Figure 3: Performance of COPPER against baselines on three datasets.

5.5 Ablation Study

We conduct an ablation study on three datasets to explore the effectiveness of each component of COPPER. We exclude the counterfactual reward (w/o CF) and proximal policy optimization (w/o PPO) individually and illustrate the outcome of Reflexion (LongChat) for comparison purposes (equivalent to eliminating the entire fine-tuning process). Experimental results are shown in Figure 4. From the results, we can conclude that both counterfactual reward and PPO fine-tuning are crucial for COPPER, and removing any part will lead to a decrease in performance. On the one hand, substituting counterfactual rewards with episode return difference rewards will lead to uniform rewards for all agents' reflections, meaning the contribution of reflection by each agent is equal. This could elevate the reward score for reflections that offer little assistance in enhancing collaboration performance, presenting a challenge in refining the reflector. On the other hand, fine-tuning PPO on the basis of SFT can further enhance the reflective ability of the shared reflector. This indicates that by maximizing environmental rewards, PPO can refine the model's output to better suit human preferences. For HotPotQA and GSM8K, we notice that the enhancement from COPPER during the initial two rounds is comparatively lower than solely fine-tuned with SFT. However, COPPER exhibits the highest success rate after five trials. This may be due to the fact that during the PPO training process, the reflector learns to sacrifice early performance for greater ultimate benefits.



(a) HotpotQA. (b) GSM8K. (c) Checkmate in One Move.

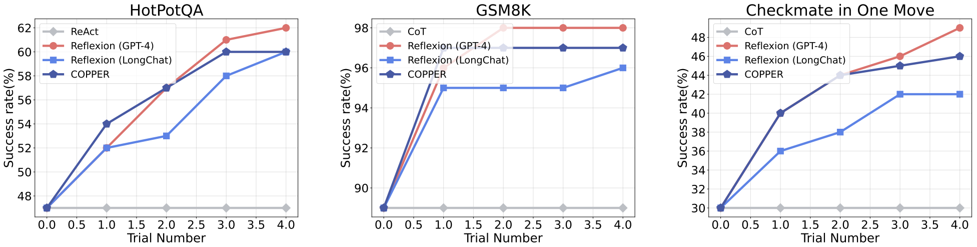
Figure 4: Ablation study.

5.6 Generalizability of the Shared Reflector

We conduct experiments on three datasets to investigate the generalizability of COPPER, with the outcomes visualized in Figure 5. Specifically, we implement COPPER trained in multi-agent systems with GPT-3.5 actors to systems with GPT-4 (model: gpt-4-turbo) actors. We compare generalized COPPER against two baselines: one featuring GPT-4 as the reflector and the other utilizing LongChat as the reflector. We conclude that COPPER remains proficient in reflection capabilities within the systems featuring GPT-4 actors. Compared to the initial success rate, COPPER demonstrates improvements of 27.7%, 9.0%, and 53.3% in HotPotQA, GSM8K, and Checkmate in One Move respectively, and achieves comparable performance to GPT-4 reflectors after 5 trials.

5.7 Generality of Counterfactual Rewards

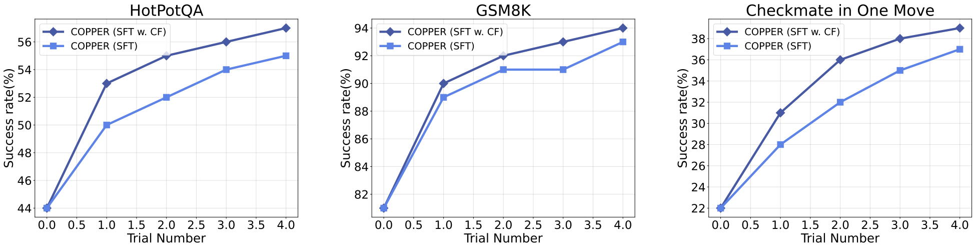
To tackle the credit assignment challenge in multi-agent systems, the paper suggests deriving scores for individual agent reflections using counterfactual rewards, which is essentially a data augmentation approach. Hence, in this section, we delve into the suitability of counterfactual rewards for LLM fine-tuning techniques beyond RLHF. Specifically, we evaluate the performance of CF SFT (employing



(a) HotpotQA. (b) GSM8K. (c) Checkmate in One Move.

Figure 5: Apply the shared reflector trained for GPT-3.5 to GPT-4.

counterfactual rewards to screen positive examples) against that of typical SFT fine-tuning (utilizing episode difference rewards to filter positive examples), as illustrated in Figure 6. Analysis of the results reveals that CF SFT outperforms regular SFT across all three scenarios. This underscores the effectiveness of counterfactual rewards in offering a more objective score based on model reflection contributions, thereby ensuring the selection of positive examples of higher quality.

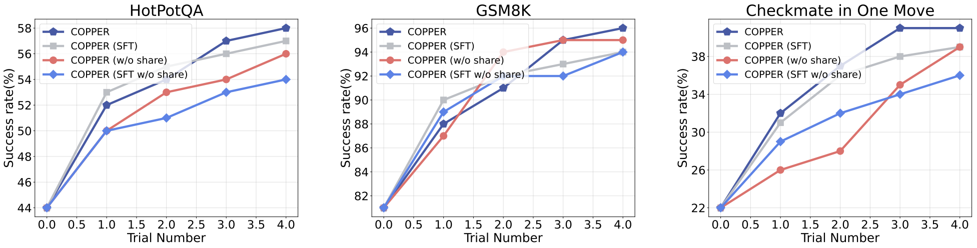


(a) HotpotQA. (b) GSM8K. (c) Checkmate in One Move.

Figure 6: Applying counterfactual rewards to SFT.

5.8 Effect of the Shared Reflector

In multi-agent systems, the quantity of reflectors will increase with the number of agents. This will lead to an excessive search space of hyper-parameters, posing challenges for practical applications. Therefore, we suggest training a shared reflector that employs carefully designed prompts to enhance the training efficiency and stability, without compromising personalized reflective abilities. In this section, we explore the effectiveness of shared reflector and present results in Figure 7. During the implementation of non-shared reflectors, given the uniformity among agents, we streamline the hyper-parameter search by aligning the hyper-parameters of each reflector. Experiments indicate that shared reflector can deliver better reflection effects, possibly because it can access more training data, leading to superior training outcomes.

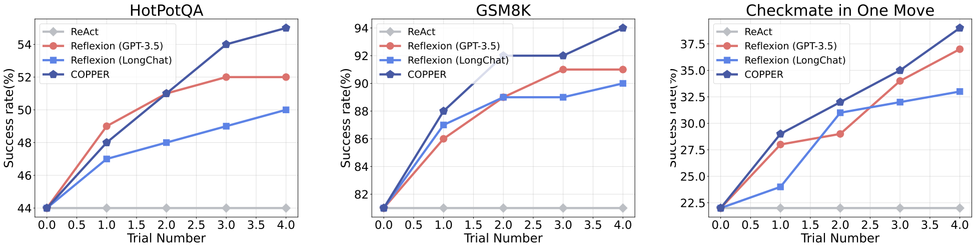


(a) HotpotQA. (b) GSM8K. (c) Checkmate in One Move.

Figure 7: Exploring the effectiveness of shared reflector.

5.9 Effectiveness of Agents' Profiles

In order to reduce training costs while generating personalized reflections in multi-agent systems, we propose to add agent profiles to the input of reflectors and train a shared reflector. In this section, we further verify the necessity of role information in multi-agent reflection scenarios. Specifically, we remove the agents' profiles from the input of the reflector, and the experimental results are shown in Figure 8. By comparing Figure 3 and Figure 8, we can observe that when using pre-trained LMs (LongChat and GPT-3.5) to reflect, the removal of the agent profile has a greater impact on the GPT-3.5 reflector. This may be due to GPT-3.5's better contextual understanding ability. Our COPPER can further improve the model's reflection ability under no-profile setting. However, the results are slightly worse than the setting with agent profiles.



(a) HotpotQA.

(b) GSM8K.

(c) Checkmate in One Move.

Figure 8: Performance under no-profile setting.

5.10 Different LLMs as Base Reflectors

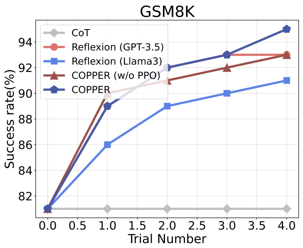


Figure 9: Fine-tuning Llama-3 as the shared reflector.

In this section, we replace the base reflector from LongChat with Llama-3 (model: llama-3-8b-16k) and explore the applicability of COPPER for different base models on the GSM8K dataset. Experimental results shown in Figure 9 demonstrate that our proposed COPPER has good performance across different base models. When comparing to the initial success rate, fine-tuning Llama-3 with counterfactual PPO shows a 17.3% enhancement, surpassing the performance of the GPT-3.5 reflector after 5 trials. Additionally, we include the outcome from fine-tuning Llama-3 exclusively with SFT. From the results, we find that PPO can further improve the reflective capabilities of the shared reflector.

6 Limitations

While counterfactual rewards can mitigate the credit assignment issue in multi-agent collaboration, constructing such rewards with LLMs imposes additional data requirements. Though our proposal involves training a shared reflector and updating the reward model's loss function to MSE, investigating more efficient data collection approaches is still needed. Besides, in this study, we restrict long-term memory to a sliding window with a maximum capacity. We believe extending the agent's memory to more advanced structures such as vector embeddings presents a promising direction for development.

7 Conclusion

In this paper, we consider leveraging the self-reflection mechanism to improve multi-agent collaboration, and propose an elegant framework COPPER. Towards more efficient reflection, we train a shared reflector using the counterfactual PPO mechanism. The counterfactual reward can be evaluated according to the impact of each agent reflection on enhancing task performance. To enhance the training efficiency and stability, we gather reflection data across agents and train a shared reflector. Experiments on three datasets indicate that our COPPER exhibits superior reflective ability and effective generalization across various actor models.

Acknowledgements

This work is supported in part by National Key R&D Program of China (2023YFF0905402), National Natural Science Foundation of China (No.62102420), Intelligent Social Governance Platform, Major Innovation & Planning Interdisciplinary Platform for the "DoubleFirst Class" Initiative, Renmin University of China, Public Computing Cloud, Renmin University of China, fund for building world-class universities (disciplines) of Renmin University of China, Intelligent Social Governance Platform. The work is also supported by Huawei Innovation Research Programs. We gratefully acknowledge the support from Mindspore[2], CANN (Compute Architecture for Neural Networks) and Ascend AI Processor used for this research.

References

[1] Saaket Agashe, Yue Fan, and Xin Eric Wang. Evaluating multi-agent coordination abilities in large language models. *CoRR*, abs/2310.03903, 2023.

[2] Gati V. Aher, Rosa I. Arriaga, and Adam Tauman Kalai. Using large language models to simulate multiple humans and replicate human subject studies. In *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, pages 337–371, 2023.

[3] Chi-Min Chan, Weize Chen, Yusheng Su, Jianxuan Yu, Wei Xue, Shanghang Zhang, Jie Fu, and Zhiyuan Liu. Chateval: Towards better llm-based evaluators through multi-agent debate. *CoRR*, abs/2308.07201, 2023.

[4] Guangyao Chen, Siwei Dong, Yu Shu, Ge Zhang, Jaward Sesay, Börje F. Karlsson, Jie Fu, and Yemin Shi. Autoagents: A framework for automatic agent generation. *CoRR*, abs/2309.17288, 2023.

[5] Xinshi Chen, Shuang Li, Hui Li, Shaohua Jiang, Yuan Qi, and Le Song. Generative adversarial user model for reinforcement learning based recommendation system. In *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pages 1052–1061, 2019.

[6] Xinyun Chen, Maxwell Lin, Nathanael Schärli, and Denny Zhou. Teaching large language models to self-debug. *CoRR*, abs/2304.05128, 2023.

[7] Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems. *CoRR*, abs/2110.14168, 2021.

[8] Yilun Du, Shuang Li, Antonio Torralba, Joshua B. Tenenbaum, and Igor Mordatch. Improving factuality and reasoning in language models through multiagent debate. *CoRR*, abs/2305.14325, 2023.

[9] Yilun Du, Shuang Li, Antonio Torralba, Joshua B. Tenenbaum, and Igor Mordatch. Improving factuality and reasoning in language models through multiagent debate. *CoRR*, abs/2305.14325, 2023.

[10] Yilun Du, Shuang Li, Antonio Torralba, Joshua B. Tenenbaum, and Igor Mordatch. Improving factuality and reasoning in language models through multiagent debate. *CoRR*, abs/2305.14325, 2023.

[11] Chen Gao, Xiaochong Lan, Zhihong Lu, Jinzhu Mao, Jinghua Piao, Huandong Wang, Depeng Jin, and Yong Li. S^3: Social-network simulation system with large language model-empowered agents. *CoRR*, abs/2307.14984, 2023.

[12] Taicheng Guo, Xiuying Chen, Yaqi Wang, Ruidi Chang, Shichao Pei, Nitesh V. Chawla, Olaf Wiest, and Xiangliang Zhang. Large language model based multi-agents: A survey of progress and challenges. *CoRR*, abs/2402.01680, 2024.

[13] Sirui Hong, Xiawu Zheng, Jonathan Chen, Yuheng Cheng, Jinlin Wang, Ceyao Zhang, Zili Wang, Steven Ka Shing Yau, Zijuan Lin, Liyang Zhou, Chenyu Ran, Lingfeng Xiao, and Chenglin Wu. Metagpt: Meta programming for multi-agent collaborative framework. *CoRR*, abs/2308.00352, 2023.

[14] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*, 2022.

[15] Wenyue Hua, Lizhou Fan, Lingyao Li, Kai Mei, Jianchao Ji, Yingqiang Ge, Libby Hemphill, and Yongfeng Zhang. War and peace (waragent): Large language model-based multi-agent simulation of world wars. *CoRR*, abs/2311.17227, 2023.

[16] Nian Li, Chen Gao, Yong Li, and Qingmin Liao. Large language model-empowered agents for simulating macroeconomic activities. *CoRR*, abs/2310.10436, 2023.

[17] Ruosen Li, Teerth Patel, and Xinya Du. PRD: peer rank and discussion improve large language model based evaluations. *Trans. Mach. Learn. Res.*, 2024, 2024.

---

[2] `https://www.mindspore.cn`

[18] Jonathan Light, Min Cai, Sheng Shen, and Ziniu Hu. From text to tactic: Evaluating llms playing the game of avalon. *CoRR*, abs/2310.05036, 2023.

[19] Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. Self-refine: Iterative refinement with self-feedback. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023.

[20] Gabriel Mukobi, Hannah Erlebach, Niklas Lauffer, Lewis Hammond, Alan Chan, and Jesse Clifton. Welfare diplomacy: Benchmarking language model cooperation. *CoRR*, abs/2310.08901, 2023.

[21] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F. Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022.

[22] Boci Peng, Yun Zhu, Yongchao Liu, Xiaohe Bo, Haizhou Shi, Chuntao Hong, Yan Zhang, and Siliang Tang. Graph retrieval-augmented generation: A survey, 2024.

[23] Chen Qian, Xin Cong, Cheng Yang, Weize Chen, Yusheng Su, Juyuan Xu, Zhiyuan Liu, and Maosong Sun. Communicative agents for software development. *CoRR*, abs/2307.07924, 2023.

[24] Yujia Qin, Shihao Liang, Yining Ye, Kunlun Zhu, Lan Yan, Yaxi Lu, Yankai Lin, Xin Cong, Xiangru Tang, Bill Qian, Sihan Zhao, Runchu Tian, Ruobing Xie, Jie Zhou, Mark Gerstein, Dahai Li, Zhiyuan Liu, and Maosong Sun. Toolllm: Facilitating large language models to master 16000+ real-world apis. *CoRR*, abs/2307.16789, 2023.

[25] Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Eric Hambro, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. Toolformer: Language models can teach themselves to use tools. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023.

[26] Yongliang Shen, Kaitao Song, Xu Tan, Dongsheng Li, Weiming Lu, and Yueting Zhuang. Hugginggpt: Solving AI tasks with chatgpt and its friends in huggingface. *CoRR*, abs/2303.17580, 2023.

[27] Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: language agents with verbal reinforcement learning. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023.

[28] Aarohi Srivastava, Abhinav Rastogi, Abhishek Rao, Abu Awal Md Shoeb, Abubakar Abid, Adam Fisch, Adam R. Brown, Adam Santoro, Aditya Gupta, Adrià Garriga-Alonso, Agnieszka Kluska, Aitor Lewkowycz, Akshat Agarwal, Alethea Power, Alex Ray, Alex Warstadt, Alexander W. Kocurek, Ali Safaya, Ali Tazarv, Alice Xiang, Alicia Parrish, Allen Nie, Aman Hussain, Amanda Askell, Amanda Dsouza, Ameet Rahane, Anantharaman S. Iyer, Anders Andreassen, Andrea Santilli, Andreas Stuhlmüller, Andrew M. Dai, Andrew La, Andrew K. Lampinen, Andy Zou, Angela Jiang, Angelica Chen, Anh Vuong, Animesh Gupta, Anna Gottardi, Antonio Norelli, Anu Venkatesh, Arash Gholamidavoodi, Arfa Tabassum, Arul Menezes, Arun Kirubarajan, Asher Mullokandov, Ashish Sabharwal, Austin Herrick, Avia Efrat, Aykut Erdem, Ayla Karakas, and et al. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *CoRR*, abs/2206.04615, 2022.

[29] Xiangru Tang, Anni Zou, Zhuosheng Zhang, Yilun Zhao, Xingyao Zhang, Arman Cohan, and Mark Gerstein. Medagents: Large language models as collaborators for zero-shot medical reasoning. *CoRR*, abs/2311.10537, 2023.

[30] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022.

[31] Qingyun Wu, Gagan Bansal, Jieyu Zhang, Yiran Wu, Shaokun Zhang, Erkang Zhu, Beibin Li, Li Jiang, Xiaoyun Zhang, and Chi Wang. Autogen: Enabling next-gen LLM applications via multi-agent conversation framework. *CoRR*, abs/2308.08155, 2023.

[32] Kai Xiong, Xiao Ding, Yixin Cao, Ting Liu, and Bing Qin. Examining inter-consistency of large language models collaboration: An in-depth analysis via debate. In *Findings of the Association for Computational Linguistics: EMNLP 2023, Singapore, December 6-10, 2023*, pages 7572–7590, 2023.

[33] Lin Xu, Zhiyuan Hu, Daquan Zhou, Hongyu Ren, Zhen Dong, Kurt Keutzer, See-Kiong Ng, and Jiashi Feng. Magic: Investigation of large language model powered multi-agent in cognition, adaptability, rationality and collaboration. *CoRR*, abs/2311.08562, 2023.

[34] Yuzhuang Xu, Shuo Wang, Peng Li, Fuwen Luo, Xiaolong Wang, Weidong Liu, and Yang Liu. Exploring large language models for communication games: An empirical study on werewolf. *CoRR*, abs/2309.04658, 2023.

[35] Haoran Yang, Yumeng Zhang, Jiaqi Xu, Hongyuan Lu, Pheng-Ann Heng, and Wai Lam. Unveiling the generalization power of fine-tuned large language models. *CoRR*, abs/2403.09162, 2024.

[36] Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W. Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 2369–2380, 2018.

[37] Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R. Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*, 2023.

[38] Weiran Yao, Shelby Heinecke, Juan Carlos Niebles, Zhiwei Liu, Yihao Feng, Le Xue, Rithesh Murthy, Zeyuan Chen, Jianguo Zhang, Devansh Arpit, Ran Xu, Phil Mui, Huan Wang, Caiming Xiong, and Silvio Savarese. Retroformer: Retrospective large language agents with policy gradient optimization. *CoRR*, abs/2308.02151, 2023.

[39] Yifan Zeng, Yiran Wu, Xiao Zhang, Huazheng Wang, and Qingyun Wu. Autodefense: Multi-agent LLM defense against jailbreak attacks. *CoRR*, abs/2403.04783, 2024.

[40] Wenqi Zhang, Yongliang Shen, Linjuan Wu, Qiuying Peng, Jun Wang, Yueting Zhuang, and Weiming Lu. Self-contrast: Better reflection through inconsistent solving perspectives. *CoRR*, abs/2401.02009, 2024.

[41] Qinlin Zhao, Jindong Wang, Yixuan Zhang, Yiqiao Jin, Kaijie Zhu, Hao Chen, and Xing Xie. Competeai: Understanding the competition behaviors in large language model-based agents. *CoRR*, abs/2310.17512, 2023.

[42] Xuhui Zhou, Hao Zhu, Leena Mathur, Ruohong Zhang, Haofei Yu, Zhengyang Qi, Louis-Philippe Morency, Yonatan Bisk, Daniel Fried, Graham Neubig, and Maarten Sap. SOTOPIA: interactive evaluation for social intelligence in language agents. *CoRR*, abs/2310.11667, 2023.

A Collaboration Settings

In HotPotQA scenario, we adopt a teacher-student collaboration paradigm, while on GSM8K and Checkmate in One Move datasets, we employ a collaborative debate setting follow [8]. We illustrate the process of the collaborations in Figure 10 and detailed introductions are as follows.

HotPotQA In the HotPotQA scenario, both teacher and student agents adopt the ReAct method for action selection. Among them, student agents can call retrieval tools to search for relevant text segments provided by the HotPotQA dataset. The retrieval tool is constructed using SimCSE (model: unsup-simcse-roberta-base).

The action space of student agent includes: **(1) Search[entity]**, which invokes a local searcher to provide relevant information. **(2) Finish[answer]**, which returns the answer and finishes the task.

The action space of teacher agent includes: **(1)[Continue]**, which means the student made a good decision and should continue the process. **(2)[Rethink]**, which means the student's previous step is wrong and should consider another step.

GSM8K In the GSM8K scenario, we set three debater agents to engage in two rounds of debate, and the final answer of the system is determined by voting on the last round answers. During the debate, intelligent agents utilize CoT to analyze and answer questions, or update the answer based on solutions of other agents.

Checkmate in One Move In the Checkmate in One Move scenario, we adopt the same collaborative debate approach to construct multi-agent system as GSM8K. The only difference is that we employ three intelligent agents for three rounds of debate.

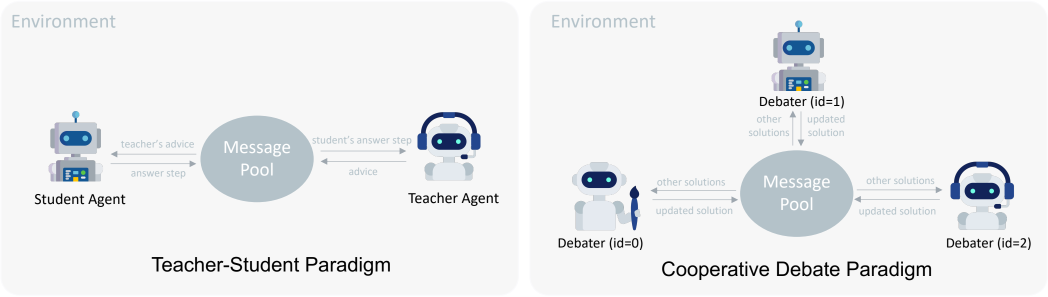


Figure 10: Multi-agent collaboration framework.

B Details of Training Data

We first use the original LongChat to construct counterfactual training data and select positive examples for supervised fine-tuning of the reflector model. The detailed data information generated in this stage is shown in Table 1.

Table 1: Statistics of training data generated by original LongChat. We show the total data volume on the left, the number of positive examples on the middle, and the number of negative examples on the right side.

Dataset	Episode Difference Reward				Counterfactual Reward			
	all	agent_0	agent_1	agent_2	all	agent_0	agent_1	agent_2
HotPotQA	8714/1946/1240	4357/973/620	4357/973/620	-/-/-	8714/2137/1330	4357/1065/686	4357/1072/644	-/-/-
GSM8K	3147/852/0	1049/284/0	1049/284/0	1049/284/0	3147/871/417	1049/138/145	1049/128/131	1049/132/141
Checkmate.	17466/906/0	5822/302/0	5822/302/0	5822/302/0	17466/682/461	5822/234/165	5822/223/137	5822/228/159

Afterwards, we employ the reflector model fine-tuned using counterfactual SFT to generate training data of the PPO stage. We use the collected counterfactual reward data to train a reward model using

MSE loss and score the reflections predicted during the PPO training process. The data collected in this stage is shown in Table 2.

Table 2: Statistics of training data generated by LongChat fine-tuned with SFT. We show the total data volume on the left, the number of positive examples on the middle, and the number of negative examples on the right side.

Dataset	Episode Difference Reward				Counterfactual Reward			
	all	agent_0	agent_1	agent_2	all	agent_0	agent_1	agent_2
HotPotQA	8822/2218/1410	4411/1109/705	4411/1109/705	-/-/-	8822/1679/1985	4411/855/1027	4411/824/958	-/-/-
GSM8K	3204/789/0	1068/263/0	1068/263/0	1068/263/0	3204/384/409	1068/129/143	1068/128/132	1068/127/134
Checkmate.	17892/591/0	5964/197/0	5964/197/0	5964/197/0	17892/412/444	5964/138/147	5965/138/169	5966/136/128

C Experimental Results on ALFWorld

ALFWorld is a classical dataset designed for training and evaluating AI agents in interactive environments. To further improve our study, we additionally conduct experiments on ALFWorld. We follow the same multi-agent collaboration setting as [31] and test the model with 134 instances. The experiment results are presented in Figure 11. From the results, we can observe that COPPER achieves better reflection performance than the original LongChat and GPT-3.5. Besides, compared to the initial success rate, COPPER brings an improvement of 37.2% with 4 times of reflections.

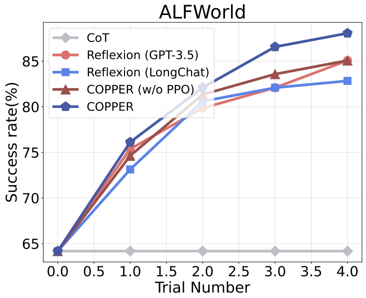
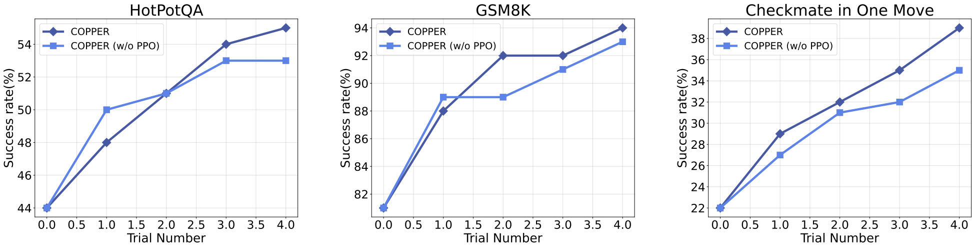


Figure 11: Experimental results on ALFWorld.

D Comparison between SFT and RLHF under No-profile Setting

We also provide experimental results comparison of different fine-tuning methods under no profile setting. The results are shown in Figure 12. We can find that even without agent profiles, the PPO method still provides enhancement to the fine-tuning of the shared reflector, further validating the effectiveness of our method.



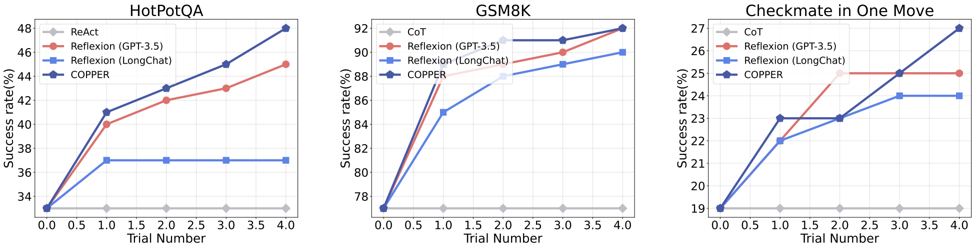
(a) HotpotQA. (b) GSM8K. (c) Checkmate in One Move.

Figure 12: Comparison between SFT and RLHF under no-profile setting.

E Experiments under Partial Information Settings

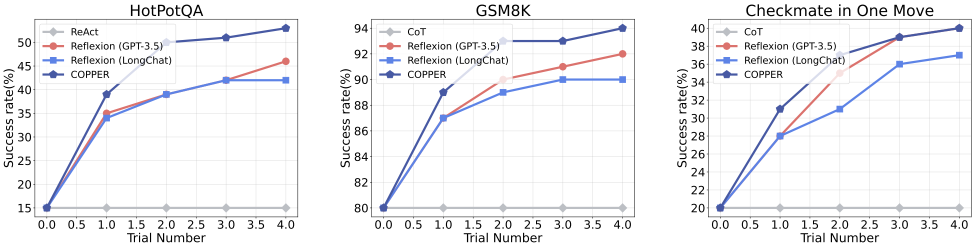
To improve our paper, we conduct more experiments to investigate the settings with partial information. In specific, we introduce two models. For the first one, we remove the information of the other agents.

For the second one, we use a proxy model to predict the information of the other agents. We present the experiment results in Figure 13 and Figure 14, respectively. We can find from the results that our COPPER still achieves better reflection performance under these partial settings.



(a) HotpotQA. (b) GSM8K. (c) Checkmate in One Move.

Figure 13: Main results without other agents' trajectories.

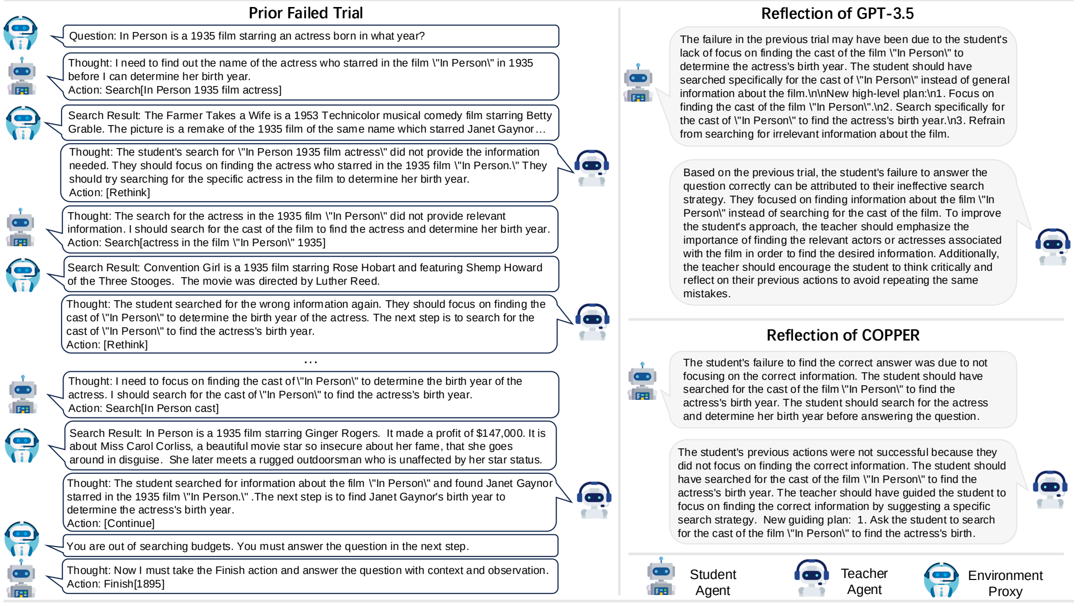


(a) HotpotQA. (b) GSM8K. (c) Checkmate in One Move.

Figure 14: Main results when predicting other agents' information with a proxy model.

F Case Study

We conduct case studies on three datasets to visually verify the effectiveness of our method, as shown in Figure 15, 16 and 17.



Question: In Person is a 1935 film starring an actress born in what year?

Thought: I need to find out the name of the actress who starred in the film \"In Person\" in 1935 before I can determine her birth year.
Action: Search[In Person 1935 film actress]

Search Result: The Farmer Takes a Wife is a 1953 Technicolor musical comedy film starring Betty Grable. The picture is a remake of the 1935 film of the same name which starred Janet Gaynor…

Thought: The student's search for \"In Person 1935 film actress\" did not provide the information needed. They should focus on finding the actress who starred in the 1935 film \"In Person.\" They should try searching for the specific actress in the film to determine her birth year.
Action: [Rethink]

Thought: The search for the actress in the 1935 film \"In Person\" did not provide relevant information. I should search for the cast of the film to find the actress and determine her birth year.
Action: Search[actress in the film \"In Person\" 1935]

Search Result: Convention Girl is a 1935 film starring Rose Hobart and featuring Shemp Howard of the Three Stooges. The movie was directed by Luther Reed.

Thought: The student searched for the wrong information again. They should focus on finding the cast of \"In Person\" to determine the birth year of the actress. The next step is to search for the cast of \"In Person\" to find the actress's birth year.
Action: [Rethink]

…

Thought: I need to focus on finding the cast of \"In Person\" to determine the birth year of the actress. I should search for the cast of \"In Person\" to find the actress's birth year.
Action: Search[In Person cast]

Search Result: In Person is a 1935 film starring Ginger Rogers. It made a profit of $147,000. It is about Miss Carol Corliss, a beautiful movie star so insecure about her fame, that she goes around in disguise. She later meets a rugged outdoorsman who is unaffected by her star status.

Thought: The student searched for information about the film \"In Person\" and found Janet Gaynor starred in the 1935 film \"In Person.\" .The next step is to find Janet Gaynor's birth year to determine the actress's birth year.
Action: [Continue]

You are out of searching budgets. You must answer the question in the next step.

Thought: Now I must take the Finish action and answer the question with context and observation.
Action: Finish[1895]

The failure in the previous trial may have been due to the student's lack of focus on finding the cast of the film \"In Person\" to determine the actress's birth year. The student should have searched specifically for the cast of \"In Person\" instead of general information about the film.\n\nNew high-level plan:\n1. Focus on finding the cast of the film \"In Person\".\n2. Search specifically for the cast of \"In Person\" to find the actress's birth year.\n3. Refrain from searching for irrelevant information about the film.

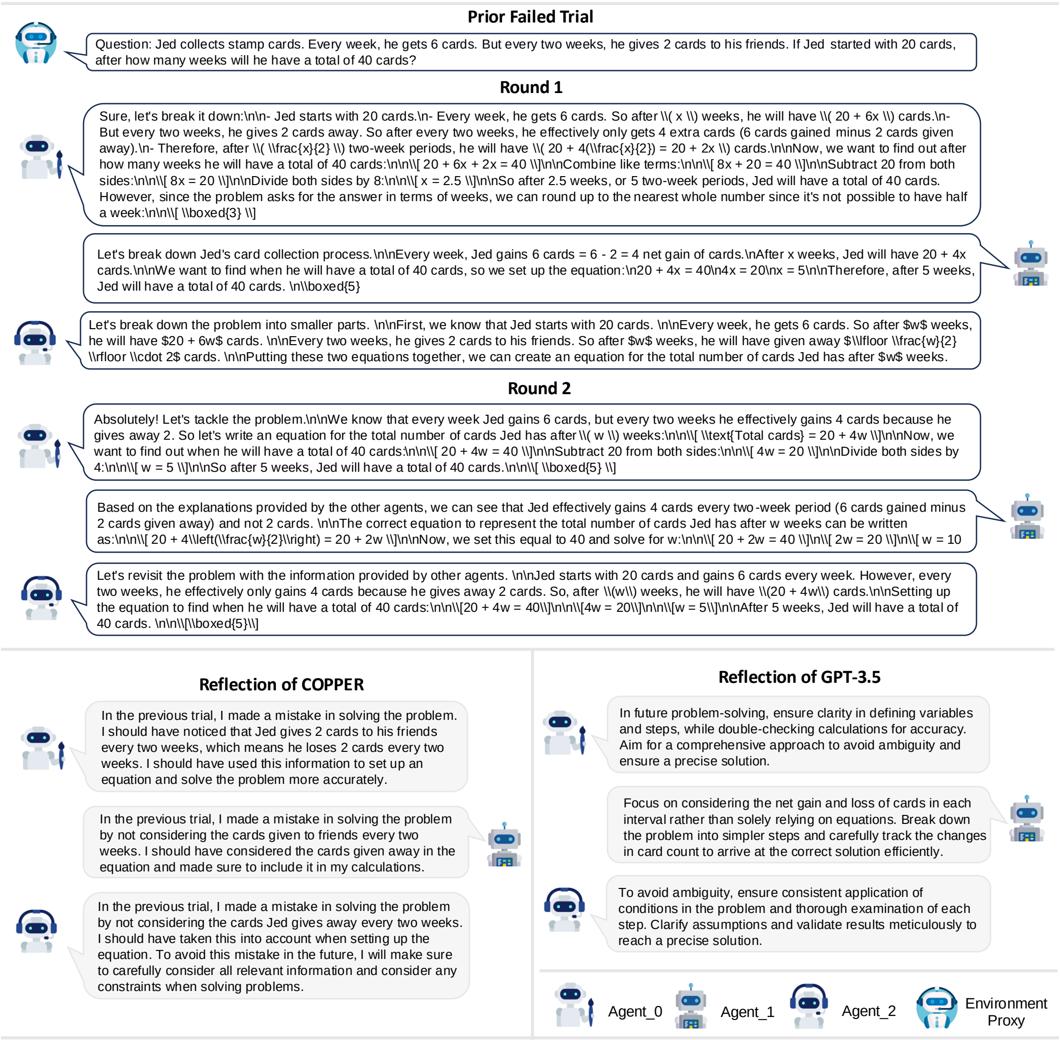
Based on the previous trial, the student's failure to answer the question correctly can be attributed to their ineffective search strategy. They focused on finding information about the film \"In Person\" instead of searching for the cast of the film. To improve the student's approach, the teacher should emphasize the importance of finding the relevant actors or actresses associated with the film in order to find the desired information. Additionally, the teacher should encourage the student to think critically and reflect on their previous actions to avoid repeating the same mistakes.

The student's failure to find the correct answer was due to not focusing on the correct information. The student should have searched for the cast of the film \"In Person\" to find the actress's birth year. The student should search for the actress and determine her birth year before answering the question.

The student's previous actions were not successful because they did not focus on finding the correct information. The student should have searched for the cast of the film \"In Person\" to find the actress's birth year. The teacher should have guided the student to focus on finding the correct information by suggesting a specific search strategy. New guiding plan: 1. Ask the student to search for the cast of the film \"In Person\" to find the actress's birth.

Figure 15: Case study on HotPotQA dataset.

For example, Figure 15 shows a prior failed trajectory in HotPotQA scenario, as well as the reflections generated by COPPER and GPT-3.5. During interactions with the environment, the student agent



Prior Failed Trial

Question: Jed collects stamp cards. Every week, he gets 6 cards. But every two weeks, he gives 2 cards to his friends. If Jed started with 20 cards, after how many weeks will he have a total of 40 cards?

Round 1

```
Sure, let's break it down:\n\n- Jed starts with 20 cards.\n- Every week, he gets 6 cards. So after \\( x \\) weeks, he will have \\( 20 + 6x \\) cards.\n-
But every two weeks, he gives 2 cards away. So after every two weeks, he effectively only gets 4 extra cards (6 cards gained minus 2 cards given
away).\n- Therefore, after \\( \\frac{x}{2} \\) two-week periods, he will have \\( 20 + 4(\\frac{x}{2}) = 20 + 2x \\) cards.\n\nNow, we want to find out after
how many weeks he will have a total of 40 cards:\n\n\\[ 20 + 6x + 2x = 40 \\]\n\nCombine like terms:\n\n\\[ 8x + 20 = 40 \\]\n\nSubtract 20 from both
sides:\n\n\\[ 8x = 20 \\]\n\nDivide both sides by 8:\n\n\\[ x = 2.5 \\]\n\nSo after 2.5 weeks, or 5 two-week periods, Jed will have a total of 40 cards.
However, since the problem asks for the answer in terms of weeks, we can round up to the nearest whole number since it's not possible to have half
a week:\n\n\\[ \\boxed{3} \\]
```

```
Let's break down Jed's card collection process.\n\nEvery week, Jed gains 6 cards = 6 - 2 = 4 net gain of cards.\nAfter x weeks, Jed will have 20 + 4x
cards.\n\nWe want to find when he will have a total of 40 cards, so we set up the equation:\n20 + 4x = 40\n4x = 20\nx = 5\n\nTherefore, after 5 weeks,
Jed will have a total of 40 cards. \n\\boxed{5}
```

```
Let's break down the problem into smaller parts. \n\nFirst, we know that Jed starts with 20 cards. \n\nEvery week, he gets 6 cards. So after $w$ weeks,
he will have $20 + 6w$ cards. \n\nEvery two weeks, he gives 2 cards to his friends. So after $w$ weeks, he will have given away $\\lfloor \\frac{w}{2}
\\rfloor \\cdot 2$ cards. \n\nPutting these two equations together, we can create an equation for the total number of cards Jed has after $w$ weeks.
```

Round 2

```
Absolutely! Let's tackle the problem.\n\nWe know that every week Jed gains 6 cards, but every two weeks he effectively gains 4 cards because he
gives away 2. So let's write an equation for the total number of cards Jed has after \\( w \\) weeks:\n\n\\[ \\text{Total cards} = 20 + 4w \\]\n\nNow, we
want to find out when he will have a total of 40 cards:\n\n\\[ 20 + 4w = 40 \\]\n\nSubtract 20 from both sides:\n\n\\[ 4w = 20 \\]\n\nDivide both sides by
4:\n\n\\[ w = 5 \\]\n\nSo after 5 weeks, Jed will have a total of 40 cards.\n\n\\[ \\boxed{5} \\]
```

```
Based on the explanations provided by the other agents, we can see that Jed effectively gains 4 cards every two-week period (6 cards gained minus
2 cards given away) and not 2 cards. \n\nThe correct equation to represent the total number of cards Jed has after w weeks can be written
as:\n\n\\[ 20 + 4\\left(\\frac{w}{2}\\right) = 20 + 2w \\]\n\nNow, we set this equal to 40 and solve for w:\n\n\\[ 20 + 2w = 40 \\]\n\\[ 2w = 20 \\]\n\\[ w = 10
```

```
Let's revisit the problem with the information provided by other agents. \n\nJed starts with 20 cards and gains 6 cards every week. However, every
two weeks, he effectively only gains 4 cards because he gives away 2 cards. So, after \\(w\\) weeks, he will have \\(20 + 4w\\) cards.\n\nSetting up
the equation to find when he will have a total of 40 cards:\n\n\\[20 + 4w = 40\\]\n\n\\[4w = 20\\]\n\n\\[w = 5\\]\n\nAfter 5 weeks, Jed will have a total of
40 cards. \n\n\\[\\boxed{5}\\]
```

Reflection of COPPER

In the previous trial, I made a mistake in solving the problem. I should have noticed that Jed gives 2 cards to his friends every two weeks, which means he loses 2 cards every two weeks. I should have used this information to set up an equation and solve the problem more accurately.

In the previous trial, I made a mistake in solving the problem by not considering the cards given to friends every two weeks. I should have considered the cards given away in the equation and made sure to include it in my calculations.

In the previous trial, I made a mistake in solving the problem by not considering the cards Jed gives away every two weeks. I should have taken this into account when setting up the equation. To avoid this mistake in the future, I will make sure to carefully consider all relevant information and consider any constraints when solving problems.

Reflection of GPT-3.5

In future problem-solving, ensure clarity in defining variables and steps, while double-checking calculations for accuracy. Aim for a comprehensive approach to avoid ambiguity and ensure a precise solution.

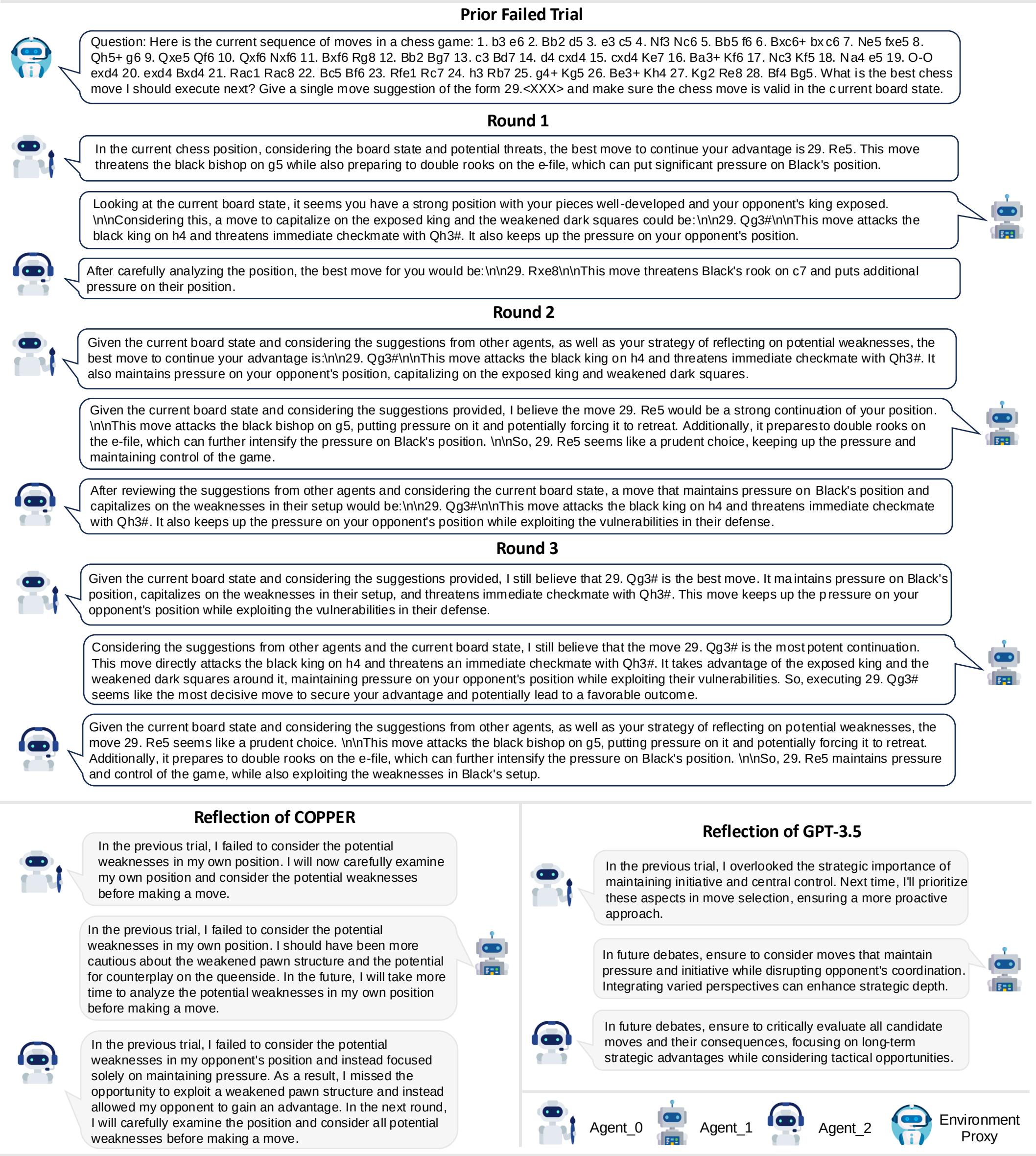
Focus on considering the net gain and loss of cards in each interval rather than solely relying on equations. Break down the problem into simpler steps and carefully track the changes in card count to arrive at the correct solution efficiently.

To avoid ambiguity, ensure consistent application of conditions in the problem and thorough examination of each step. Clarify assumptions and validate results meticulously to reach a precise solution.

Figure 16: Case study on GSM8K dataset.

needs to constantly modify keywords of the retriever API to obtain information related to the problem. However, in the previous trajectory, the student agent underwent multiple modifications to complete the first step of inference, which results in reaching the pre-defined maximum search step limit. Therefore, the agent is forced to answer the question directly based on search result of the first reasoning step, leading to an incorrect answer. By analyzing the reflections of COPPER and GPT-3.5, we can observe that: (1) Compared to GPT-3.5, COPPER has a deeper understanding of agent roles and thus generate corresponding reflections for each agent. For example, in the reflection of the teacher agent, the reflection generated by GPT-3.5 contains lots of descriptions of problems related to the student agent, while COPPER focuses more on reflecting on the role of the teacher. (2) Compared to GPT-3.5, COPPER can conduct more effective and comprehensive root cause analysis to identify the cause of failure. In the reflection of the student agent, both GPT-3.5 and COPPER mention that "the case of In Person" should be used as the search keyword, but COPPER additionally suggests that the relevant information of the actress should be retrieved before answering the question.

As for the case studies on GSM8K (Figure 16) and Checkmate in One Move (Figure 17) datasets, we can find that in the context of the multi-agent debate, compared to GPT-3.5, COPPER can accurately identify key issues where mistakes occurred in previous trials and generate more specific reflections to guide agents in improving their responses. For example, in the case of GSM8K, COPPER identified that previous errors mainly stemmed from an incorrect analysis of equal relations and emphasized



Prior Failed Trial

Question: Here is the current sequence of moves in a chess game: 1. b3 e6 2. Bb2 d5 3. e3 c5 4. Nf3 Nc6 5. Bb5 f6 6. Bxc6+ bxc6 7. Ne5 fxe5 8. Qh5+ g6 9. Qxe5 Qf6 10. Qxf6 Nxf6 11. Bxf6 Rg8 12. Bb2 Bg7 13. c3 Bd7 14. d4 cxd4 15. cxd4 Ke7 16. Ba3+ Kf6 17. Nc3 Kf5 18. Na4 e5 19. O-O exd4 20. exd4 Bxd4 21. Rac1 Rac8 22. Bc5 Bf6 23. Rfe1 Rc7 24. h3 Rb7 25. g4+ Kg5 26. Be3+ Kh4 27. Kg2 Re8 28. Bf4 Bg5. What is the best chess move I should execute next? Give a single move suggestion of the form 29.<XXX> and make sure the chess move is valid in the current board state.

Round 1

In the current chess position, considering the board state and potential threats, the best move to continue your advantage is 29. Re5. This move threatens the black bishop on g5 while also preparing to double rooks on the e-file, which can put significant pressure on Black's position.

Looking at the current board state, it seems you have a strong position with your pieces well-developed and your opponent's king exposed. \n\nConsidering this, a move to capitalize on the exposed king and the weakened dark squares could be:\n\n29. Qg3#\n\nThis move attacks the black king on h4 and threatens immediate checkmate with Qh3#. It also keeps up the pressure on your opponent's position.

After carefully analyzing the position, the best move for you would be:\n\n29. Rxe8\n\nThis move threatens Black's rook on c7 and puts additional pressure on their position.

Round 2

Given the current board state and considering the suggestions from other agents, as well as your strategy of reflecting on potential weaknesses, the best move to continue your advantage is:\n\n29. Qg3#\n\nThis move attacks the black king on h4 and threatens immediate checkmate with Qh3#. It also maintains pressure on your opponent's position, capitalizing on the exposed king and weakened dark squares.

Given the current board state and considering the suggestions provided, I believe the move 29. Re5 would be a strong continuation of your position. \n\nThis move attacks the black bishop on g5, putting pressure on it and potentially forcing it to retreat. Additionally, it prepares to double rooks on the e-file, which can further intensify the pressure on Black's position. \n\nSo, 29. Re5 seems like a prudent choice, keeping up the pressure and maintaining control of the game.

After reviewing the suggestions from other agents and considering the current board state, a move that maintains pressure on Black's position and capitalizes on the weaknesses in their setup would be:\n\n29. Qg3#\n\nThis move attacks the black king on h4 and threatens immediate checkmate with Qh3#. It also keeps up the pressure on your opponent's position while exploiting the vulnerabilities in their defense.

Round 3

Given the current board state and considering the suggestions provided, I still believe that 29. Qg3# is the best move. It maintains pressure on Black's position, capitalizes on the weaknesses in their setup, and threatens immediate checkmate with Qh3#. This move keeps up the pressure on your opponent's position while exploiting the vulnerabilities in their defense.

Considering the suggestions from other agents and the current board state, I still believe that the move 29. Qg3# is the most potent continuation. This move directly attacks the black king on h4 and threatens an immediate checkmate with Qh3#. It takes advantage of the exposed king and the weakened dark squares around it, maintaining pressure on your opponent's position while exploiting their vulnerabilities. So, executing 29. Qg3# seems like the most decisive move to secure your advantage and potentially lead to a favorable outcome.

Given the current board state and considering the suggestions from other agents, as well as your strategy of reflecting on potential weaknesses, the move 29. Re5 seems like a prudent choice. \n\nThis move attacks the black bishop on g5, putting pressure on it and potentially forcing it to retreat. Additionally, it prepares to double rooks on the e-file, which can further intensify the pressure on Black's position. \n\nSo, 29. Re5 maintains pressure and control of the game, while also exploiting the weaknesses in Black's setup.

Reflection of COPPER

In the previous trial, I failed to consider the potential weaknesses in my own position. I will now carefully examine my own position and consider the potential weaknesses before making a move.

In the previous trial, I failed to consider the potential weaknesses in my own position. I should have been more cautious about the weakened pawn structure and the potential for counterplay on the queenside. In the future, I will take more time to analyze the potential weaknesses in my own position before making a move.

In the previous trial, I failed to consider the potential weaknesses in my opponent's position and instead focused solely on maintaining pressure. As a result, I missed the opportunity to exploit a weakened pawn structure and instead allowed my opponent to gain an advantage. In the next round, I will carefully examine the position and consider all potential weaknesses before making a move.

Reflection of GPT-3.5

In the previous trial, I overlooked the strategic importance of maintaining initiative and central control. Next time, I'll prioritize these aspects in move selection, ensuring a more proactive approach.

In future debates, ensure to consider moves that maintain pressure and initiative while disrupting opponent's coordination. Integrating varied perspectives can enhance strategic depth.

In future debates, ensure to critically evaluate all candidate moves and their consequences, focusing on long-term strategic advantages while considering tactical opportunities.

Figure 17: Case study on Checkmate in One Move dataset.

the need to pay attention to "every two weeks". In the case of Checkmate in One Move, COPPER pointed out the importance of focusing on the weakness in the own position.

G Ethical Consideration

In this paper, we propose using the self-reflection mechanism to enhance the collaborative ability of multi-agent systems, and explore the effectiveness of our method in three scenarios: question answering, mathematics, and chess. Experimental results show that our method can effectively enhance the ability of multi-agent systems to solve complex tasks, which helps to enhance the application of multi-agent systems in real-world scenarios, such as disaster response, intelligent transportation systems, and other scenarios. However, in the task of question-answering, the agent can call API to utilize retrieval tools, which may pose potential risks, such as tampering with the information in Wikipedia during the searching process. However, during the implementation, we limit the text content that intelligent agents can access, thus avoiding this issue.

H Prompts

H.1 Single-Agent Prompts

H.1.1 HotPotQA

For single-agent setting on HotPotQA, we adopt the same few-shot examples as [27].

The actor prompt for single-agent ReAct.

Solve a question answering task with interleaving Thought, Action, Observation steps. Thought can reason about the current situation, and Action can be three types:
(1) Search[entity], which searches the exact entity on Wikipedia and returns the first paragraph if it exists. If not, it will return some similar entities to search.
(2) Finish[answer], which returns the answer and finishes the task. You may take as many steps as necessary.

Here are some examples:
{examples}
(END OF EXAMPLES)

Question: {question}
{scratchpad}

The actor prompt for single-agent Reflexion.

Solve a question answering task with interleaving Thought, Action, Observation steps. Thought can reason about the current situation, and Action can be two types:
(1) Search[entity], which searches the exact entity on Wikipedia and returns the first paragraph if it exists. If not, it will return some similar entities to search.
(2) Finish[answer], which returns the answer and finishes the task. You may take as many steps as necessary.

Here are some examples:
{examples}
(END OF EXAMPLES)

{reflections}

Question: {question}{scratchpad}

The reflector prompt for single-agent Reflexion.

You are an advanced reasoning agent that can improve based on self refection. You will be given a previous reasoning trial in which you were given access to an Docstore API environment and a question to answer. You were unsuccessful in answering the question either because you guessed the wrong answer with Finish[<answer>], or you used up your set number of reasoning steps. In a few sentences, Diagnose a possible reason for failure and devise a new, concise, high level plan that aims to mitigate the same failure. Use complete sentences.

Here are some examples:
{examples}

Previous trial:
Question: {question}
{scratchpad}

Reflection:

H.1.2 GSM8K

We adopt a zero-shot setting on GSM8K and below are the prompts.

The actor prompt for single-agent CoT.

Can you solve the following math problem?
{question}
Explain your reasoning. Your final answer should be a single numerical number, in the form \\boxed{{answer}}, at the end of your response.

The actor prompt for single-agent Reflexion.

Reflections of previous trials:
{reflections}

Can you solve the following math problem?
{question}
Explain your reasoning. Your final answer should be a single numerical number, in the form \\boxed{{answer}}, at the end of your response.

The reflector prompt for single-agent Reflexion.

You are an advanced reasoning agent that can improve based on self refection. In previous trial, your task is to solve a math problem. However, you were unsuccessful in the previous trial. Now given previous interactions, you need to carefully examine the problem-solving ideas and calculation results, and form a reflection to avoid these problems in the next round. The reflection should be less than 50 words.

Previous Question:
{question}

Previous interactions:
{context}

Reflection:

H.1.3 Checkmate in One Move

We also adopt a zero-shot setting on Checkmate in One Move and below are the prompts.

The actor prompt for single-agent CoT.

Here is the current sequence of moves in a chess game:
{question}
What is the best chess move I should execute next?
Give a single move suggestion of the form {answer_step}.<XXX> and make sure the chess move is valid in the current board state.

The actor prompt for single-agent Reflexion.

Reflections of previous trials:
{reflections}

Here is the current sequence of moves in a chess game:
{question}
What is the best chess move I should execute next?
Give a single move suggestion of the form {answer_step}.<XXX> and make sure the chess move is valid in the current board state.

The reflector prompt for single-agent Reflexion.

You are an advanced reasoning agent that can improve based on self refection. In previous trial, your task is to give the best next chess move. However, you were unsuccessful in the previous trial. Now given previous interactions, you need to carefully examine the problem-solving ideas and calculation results, and form a reflection to avoid these problems in the next round. The reflection should be less than 50 words.

Previous Game:
{question}

Previous interactions:
{context}

Reflection:

H.2 Multi-Agent Prompts

H.2.1 HotPotQA

We follow [38] to design the prompts in this scenario.

The actor prompt of the student agent when deploying ReAct.

You are a student agent. Your task is to answer the question under the guidance of the teacher agent. You should make a reasonable plan at first.
Solve the task below with interleaving Thought, Action, Context steps. Context is the summary of historical interactions. Thought can reason about the current situation.
Your action can be two types:
(1) Search[entity], which invokes a local searcher to provide you with relevant information.
(2) Finish[answer], which returns the answer and finishes the task.

Please note: You only need to complete the thought step and output Search [Entity] in the action step, and we will return the relevant content in "Observation" for you. If you find an answer, submit it via "Finish [answer]". Identical searches will only return similar content. If the returned content has no relevant information, please actively try searching for different keywords. When submitting your answer, please try to submit the full answer if you think it is ambiguous, e.g. "movie director" is better than "director"! The answer to the question should be as accurate and concise as possible, i.e. try to answer the question with phrases instead of long sentences. Please answer yes-no question with either "yes" or "no".

Examples:
Context: Searched for Arthur's Magazine and found it was started in 1844. Teacher agreed with previous action and suggested finding the founding date of First for Women. Searched for First for Women. Question: Which magazine was started first Arthur's Magazine or First for Women?
Observation: Search Result: Search[First for Women] First for Women is a woman's magazine published by Bauer Media Group in the USA.[1] The magazine was started in 1989. Teacher's Suggestion: [Continue] Now you should answer the question.
Thought: First for Women was started in 1989. 1844 (Arthur's Magazine) < 1989 (First for Women), so Arthur's Magazine was started first.
Action: Finish[Arthur's Magazine]
END OF EXAMPLES

Context: {context}
Question: {question}
Observation: {observation}

The actor prompt of the teacher agent when deploying ReAct.

You are a teacher agent, and your task is to guide the student agent to answer the questions. You should analyze whether the student's step is logically helpful, provide an analysis, and give your final action. You can also give advice on the student's future step.
Solve the task below with interleaving Thought, Action, Context steps. Context is the summary of historical interactions. Thought can reason about the current situation.
Your action can be two types:
(1) [Continue], which means the student made a good decision and should continue the process.
(2) [Rethink], which means the student's previous step is wrong and should consider another step.

Please note: You can analyze the correctness of student agent's action, summarize the useful information the student found, or provide suggestions for subsequent steps.
Your action step only have two types: [Continue] or [Rethink]. You can only take one of the above actions. Most of the time, please be an encouraging teacher, that is, unless the student is completely wrong, use [Continue] action more often.

Examples:
Context: Searched for Arthur's Magazine and found it was started in 1844. Teacher's advice is continue.
Question: Which magazine was started first Arthur's Magazine or First for Women?
Observation: Previous advice: [Continue] Finding the start time of Arthur's Magazine is helpful. Next the student should search First for Women and find its founding date. Student's action: Search[First for Women] First for Women is a woman's magazine published by Bauer Media Group in the USA.[1] The magazine was started in 1989.
Thought: The founding date of First for Women is helpful. The student should take the Finish step next.
Action: [Continue]
END OF EXAMPLES

Context: {context}
Question: {question}
Observation: {observation}

The actor prompt of the student agent when deploying Reflexion.

You are a student agent. Your task is to answer the question under the guidance of the teacher agent. You should make a reasonable plan at first.
Solve the task below with interleaving Thought, Action, Context steps. Context is the summary of historical interactions. Thought can reason about the current situation.
Your action can be two types:
(1) Search[entity], which invokes a local searcher to provide you with relevant information.
(2) Finish[answer], which returns the answer and finishes the task.

Please note: You only need to complete the thought step and output Search [Entity] in the action step, and we will return the relevant content in "Observation" for you. If you find an answer, submit it via "Finish [answer]". Identical searches will only return similar content. If the returned content has no relevant information, please actively try searching for different keywords. When submitting your answer, please try to submit the full answer if you think it is ambiguous, e.g. "movie director" is better than "director"! The answer to the question should be as accurate and concise as possible, i.e. try to answer the question with phrases instead of long sentences. Please answer yes-no question with either "yes" or "no".

Reflections of previous trials:
{reflections}

Examples:
Context: Searched for Arthur's Magazine and found it was started in 1844. Teacher agreed with previous action and suggested finding the founding date of First for Women. Searched for First for Women. Question: Which magazine was started first Arthur's Magazine or First for Women?
Observation: Search Result: Search[First for Women] First for Women is a woman's magazine published by Bauer Media Group in the USA.[1] The magazine was started in 1989. Teacher's Suggestion: [Continue] Now you should answer the question.
Thought: First for Women was started in 1989. 1844 (Arthur's Magazine) < 1989 (First for Women), so Arthur's Magazine was started first.
Action: Finish[Arthur's Magazine]
END OF EXAMPLES

Context: {context}
Question: {question}
Observation: {observation}

The actor prompts of the teacher agent when deploying Reflexion.

You are a teacher agent, and your task is to guide the student agent to answer the questions. You should analyze whether the student's step is logically helpful, provide an analysis, and give your final action. You can also give advice on the student's future step.
Solve the task below with interleaving Thought, Action, Context steps. Context is the summary of historical interactions. Thought can reason about the current situation.
Your action can be two types:
(1) [Continue], which means the student made a good decision and should continue the process.
(2) [Rethink], which means the student's previous step is wrong and should consider another step.

Please note: You can analyze the correctness of student agent's action, summarize the useful information the student found, or provide suggestions for subsequent steps.
Your action step only have two types: [Continue] or [Rethink]. You can only take one of the above actions. Most of the time, please be an encouraging teacher, that is, unless the student is completely wrong, use [Continue] action more often.

Reflections of previous trials:
{reflections}

Examples:
Context: Searched for Arthur's Magazine and found it was started in 1844. Teacher's advice is continue.
Question: Which magazine was started first Arthur's Magazine or First for Women?
Observation: Previous advice: [Continue] Finding the start time of Arthur's Magazine is helpful. Next the student should search First for Women and find its founding date. Student's action: Search[First for Women] First for Women is a woman's magazine published by Bauer Media Group in the USA.[1] The magazine was started in 1989.
Thought: The founding date of First for Women is helpful. The student should take the Finish step next.
Action: [Continue]
END OF EXAMPLES

Context: {context}
Question: {question}
Observation: {observation}

The reflector prompt of the student agent when deploying Reflexion.

You are an advanced reasoning agent that can improve based on self refection. In previous trial, you are a student agent, and your task is to answer the question under the guidance of the teacher agent. The teacher agent provides guidance on your step and explains the reasons. You were unsuccessful in answering the question either because you guessed the wrong answer with Finish[answer], or you used up the set number of reasoning steps. Now given previous interactions from student's and teacher's perspective, you should diagnose a possible reason for failure and devise a new, concise, high level plan that aims to mitigate the same failure in a few sentences.
Please note: If you believe that the previous searching and collaboration process as well as the answer were correct, please try answering the question in a different way, e.g. try to provide more concise answers, or using the same words as the question itself.

Previous trial:
Question: {question}

Interaction:
Student: {student_context}
Teacher: {teacher_context}

Reflection:

The reflector prompt of the teacher agent when deploying Reflexion.

You are an advanced reasoning agent that can improve based on self refection. In previous trial, you are a teacher agent, and your task is to guide the student agent to answer the questions. The student is unsuccessful in answering the question either because he guessed the wrong answer, or he used up the set number of reasoning steps. Now given previous interactions from student's and teacher's perspective, you should diagnose a possible reason for failure and devise a new, concise, high level guiding plan.
Please note: If you believe that the previous searching and collaboration process as well as the answer were correct, please try answering the question in a different way, e.g. try to provide more concise answers, or using the same words as the question itself.

Previous trial:
Question: {question}

Interaction:
Student: {student_context}
Teacher: {teacher_context}

Reflection:

The prompt of the context model of the student agent.

You are a student agent. Your task is to answer the question under the guidance of the teacher agent. Now you are provided with a previous summary, as well as new messages that were not included in the original summary. Your summary should encapsulate the main points of the new messages and integrate them into the existing summary to create a comprehensive recap. Highlight the key issues discussed, decisions made, and any actions assigned. Record the helpful factual information given by the search engine.
Please ensure that the final summary does not exceed {char_limit} characters.

Examples:
Question: Which magazine was started first Arthur's Magazine or First for Women?
Previous Summary: Searched for Arthur's Magazine.
New Observation: Search Result: Search[Arthur's Magazine] Arthur's Magazine (1844-1846) was an American literary periodical published in Philadelphia in the 19th century. Teacher's Suggestion: [Continue] Finding the start time of Arthur's Magazine is helpful. Next the student should search First for Women and find its founding date.
New Thought: Arthur's Magazine was started in 1844. I need to search First for Women next.
New Action: Search[First for Women]
Summary: Searched for Arthur's Magazine and found it was started in 1844. Teacher agreed with previous action and suggested finding the founding date of First for Women. Searched for First for Women.
END OF EXAMPLES

Question: {question}
Previous Summary: {context}
New Observation: {observation}
New Thought: {thought}
New Action: {action}
Summary:

The prompt of the context model of the teacher agent.

You are a teacher agent, and your task is to guide the student agent to answer the questions. Now you are provided with a previous summary, as well as new messages that were not included in the original summary. Your summary should encapsulate the main points of the new messages and integrate them into the existing summary to create a comprehensive recap. Highlight the key issues discussed, decisions made, and any actions assigned. Record the helpful factual information given by the search engine.

Please ensure that the final summary does not exceed {char_limit} characters.

Examples:

Question: Which magazine was started first Arthur's Magazine or First for Women?

Previous Summary: The student searched for Arthur's Magazine and found it was started in 1844. The action is helpful and the next step is finding the founding date of First for Women.

New Observation: Previous advice: [Continue] Finding the start time of Arthur's Magazine is helpful. Next the student should search First for Women and find its founding date. Student's action: Search[First for Women] First for Women is a woman's magazine published by Bauer Media Group in the USA.[1] The magazine was started in 1989.

New Thought: The founding date of First for Women is helpful. The student should take the Finish step next.

New Action: [Continue]

Summary: The student searched Arthur's Magazine and found it was started in 1844. The student then searched First for women and found it was started in 1989. Previous actions are helpful. The student should give the answer in the next step.

END OF EXAMPLES

Question: {question}

Previous Summary: {context}

New Observation: {observation}

New Thought: {thought}

New Action: {action}

Summary:

H.2.2 GSM8K

We follow [8] to design the prompts of debaters in this scenario.

The actor prompt for the agent to generate initial answers when deploying CoT.

Can you solve the following math problem?

{question}

Explain your reasoning. Your final answer should be a single numerical number, in the form \\boxed{{answer}}, at the end of your response.

The actor prompt for the agent to generate updated answers when deploying CoT.

These are the solutions to the problem from other agents:

{solutions}

Using the solutions from other agents as additional information, can you provide your answer to the math problem? The original math problem is {question}.

Your final answer should be a single numerical number, in the form \\boxed{{answer}}, at the end of your response.

The actor prompt for the agent to generate initial answers when deploying Relfexion.

Reflections of previous trials:
{reflections}

Can you solve the following math problem?
{question}
Explain your reasoning. Your final answer should be a single numerical number, in the form \\boxed{{answer}}, at the end of your response.

The actor prompt for the agent to generate updated answers when deploying Reflexion.

Reflections of previous trials:
{reflections}

These are the solutions to the problem from other agents:
{solutions}
Using the solutions from other agents as additional information, can you provide your answer to the math problem? The original math problem is {question}.
Your final answer should be a single numerical number, in the form \\boxed{{answer}}, at the end of your response.

The reflector prompt of each debater agent.

You are an advanced reasoning agent that can improve based on self refection. In previous trial, you are {role} and you are supposed to solve a math problem through debating with other agents. However, you were unsuccessful in the previous trial. Now given previous interactions, you need to carefully examine the problem-solving ideas and calculation results, and form a reflection to avoid these problems in the next round. The reflection should be less than 50 words.

Previous Question:
{question}

Previous interactions:
{context}

Reflection:

The prompt of the context model of each agent.

Please summarize the following process in concise language, including the opinions of all agents.

Please note: Please summarize the viewpoints of each agent and retain the role of the agent in the summary, such as agent_0, agent_1, agent_2.

{scratchpad}

Summary:

H.2.3 Checkmate in One Move

We follow [8] to design the prompts of debaters in this scenario.

The actor prompt for the agent to generate initial answers when deploying CoT.

Here is the current sequence of moves in a chess game:
{question}
What is the best chess move I should execute next?
Give a single move suggestion of the form {answer_step}.<XXX> and make sure the chess move is valid in the current board state.

The actor prompt for the agent to generate updated answers when deploying CoT.

Here are other chess move suggestions from other agents:
{solutions}
Using the chess suggestions from other agents as additional advice, can you give me your updated thoughts on the best next chess move I should play given the chess sequence?
The current sequence of moves in a chess game is: {self.question}
Give a single move suggestion of the form {answer_step}.<XXX> and make sure the chess move is valid in the current board state.

The actor prompt for the agent to generate initial answers when deploying Reflexion.

Reflections of previous trials:
{reflections}

Here is the current sequence of moves in a chess game:
{question}
What is the best chess move I should execute next?
Give a single move suggestion of the form {answer_step}.<XXX> and make sure the chess move is valid in the current board state.

The actor prompt for the agent to generate updated answers when deploying Reflexion.

Reflections of previous trials:
{reflections}

Here are other chess move suggestions from other agents:
{solutions}
Using the chess suggestions from other agents as additional advice, can you give me your updated thoughts on the best next chess move I should play given the chess sequence?
The current sequence of moves in a chess game is: {self.question}
Give a single move suggestion of the form {answer_step}.<XXX> and make sure the chess move is valid in the current board state.

The reflector prompt of each debater agent.

You are an advanced reasoning agent that can improve based on self refection. In previous trial, you are {role} you are supposed to give the best next chess move through debating with other agents. However, you were unsuccessful in the previous trial. Now given previous interactions, you need to carefully examine the problem-solving ideas and calculation results, and form a reflection to avoid these problems in the next round. The reflection should be less than 50 words.

Previous Game:
{question}

Previous interactions:
{context}

Reflection:

The prompt of the context model of each agent.

Please summarize the following process in concise language, including the opinions of all agents.

Please note: Please summarize the viewpoints of each agent and retain the role of the agent in the summary, such as agent_0, agent_1, agent_2.

{scratchpad}

Summary:

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: In this paper, we suggest improving multi-agent collaboration through self-reflection and propose COPPER, which fine-tunes a shared reflector model with counterfactual PPO mechanism. We accurately introduce COPPER in abstract and introduction and highlight our contribution at end of introduction.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We discuss the limitations of our work in Section 6

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [NA]

Justification: We focus on the application of multi-agent systems on downstream tasks and do not include theoretical results in this paper.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: We provide all the information for reproducibility including data and coding details in Section 5 and Appendix A.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: We provide all the information in Section 5 and Appendix A. Our code can be found at: `https://anonymous.4open.science/r/copper-F72A/`

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: We provide this information in Section 5.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [NA]

   Justification: We do not provide error bars because using GPT-3.5 or GPT-4 for test is costly. Previous relevant works [38, 27] do not provide error bars, either.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We provide the information in Section 5.3

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: Our paper conforms with the NeurIPS Code of Ethics in every respect.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We discuss the potential societal impacts in Appendix G

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.

- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [Yes]

    Justification: We discuss the question in Appendix G.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We cite them in the paper properly.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.

- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The details are well introduced in the paper and our code can be found at `https://anonymous.4open.science/r/copper-F72A/`

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: We do not involve research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: We do not involve research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.