PSSD: Making Large Language Models Self-denial via Human Psyche Structure

Anonymous Author(s)
Submission Id: 2224*

Abstract

The enhance of accuracy in reasoning results of LLMs arouses the community's interests, wherein pioneering studies investigate post-hoc strategies to rectify potential mistakes. Despite extensive efforts, they are all stuck in a state of resource competition demanding significant time and computing expenses. The cause of the situation lies in failing to identify the fundamental feature of the solutions in this line, coined as the self-denial of LLMs. In other words, LLMs should confidently determine the potential mistakes and carefully execute the targeted correction. As the whole procedure conducts within LLMs, supporting and persuasive references are hard to acquire, while the absence of specific steps towards refining mistakes persists even when errors are acknowledged. In response to the challenges, we present PSSD, which refers to and implements the human psyche structure such that three distinct and interconnected roles contribute to human reasoning. Specifically, PSSD leverages the recent multi-agent paradigm, and is further enhanced with three innovatively conceived roles: (1) the intuition-based id role that provides initial attempts based on benign LLMs; (2) the rule-driven superego role that summarizes rules to regulate the above attempts, and returns specific key points as guidance; and (3) the script-centric ego role that absorbs all procedural information to generate executable script for the final answer prediction. Extensive experiments demonstrate that the proposed design not only better enhance reasoning capabilities, but also seamlessly integrate with current models, leading to superior performance.

CCS Concepts

• Information systems → Information retrieval; **• Computing methodologies → Natural language processing**.

Keywords

Mistake correction, Multi-agent debate, Self-denial

ACM Reference Format:
Anonymous Author(s). 2025. PSSD: Making Large Language Models Self-denial via Human Psyche Structure. In *Proceedings of The Web Conference 2025 (WWW '25).* ACM, New York, NY, USA, 11 pages. https://doi.org/XXXXXXX.XXXXXXX

1 Introduction

Recently, the NLP community has witnessed the booming development of LLMs, where many downstream tasks manifested new milestones, especially in reasoning tasks [15, 18]. Albeit superior, LLMs still exhibit weakness in guaranteeing the reasoning correctness. That is, LLMs tend to make up facts and details, hence *misleading* the direction of inference and generating the errorneous results [16, 31, 32].

Amidst this backdrop, pioneering studies have investigated the post-hoc strategy, which emphasizes refining generated results, primarily influenced by the concept of *correcting based on mistakes* [11]. According to the achievement, these methods mainly can be put into three categories: (1) *Fine-tuning LLMs* aims to proactively prevent the recurrence of mistakes by fine-tuning on previous correct and incorrect samples [1, 5, 19, 27]; (2) *Leveraging tools* iteratively polishes results through interaction with external assistance to correct the mistakes in each step [17, 23, 25, 30]; (3) *Multi-agent debate* defines multiple roles of LLMs, with each individually generating a response and engaging in multi-round debates to reach a consistent answer [4, 6, 12, 29]. The training phase in first class does not meet the timely demands in practical applications. Though methods in second class can provide immediate results, their performance relies heavily on availability of high-quality external resources. In order to ensure the timeliness and avoid the mentioned dependency, our research focuses on exploiting the *inherent potential* of LLMs, which belongs to last stream (vividly shown in Figure 3).

As aforementioned, current studies in multi-agent debate tries to allocate several roles for agents initializing discussions, where the round of interaction depends on reach of the consistency and number of roles relies on the empirical study. These intuitive designs, instead of approaching the fundamental features of learning from mistakes, are stuck in the *resource competition* (ref. Figure 5), in which satisfactory results are highly dependent on the times of invoking LLMs. To avoid the dilemma, we first identify the feature in this category as: *facilitating agents' self-denial*, where the reasoning direction of LLMs is guided and refined by experience, particularly through exploiting mistakes. These enhanced agents consciously exert control over the generation by appropriate interventions, corrections, or completions in reasoning, based on confidential determination. The process theoretically prevents an uncontrollable stack of resources to obtain a consistent result.

The identification, meanwhile, reveals notorious challenges in pursuit of the goal. (1) To ensure accurate refinement, LLMs must maintain confidentiality in their determination about the correctness of results, which necessitates persuasive evidences. Without access to external knowledge resources, the acquisition of supporting information through direct means is limited. (2) The rectification of the wrong generation for LLMs in many situations is even more challenging, as reasons resulting in this outcome are intricate and imperceptible. If LLMs can discern these underlying factors, they would follow a logical reasoning path instead of generating incorrect answers. (3) It is significant to make the required resource under control. When the procedure merely relies on development of refinement, it transforms into a variant of resource competition.

To address the task, we refer to Freudian theory of human psyche [7], in which *the id*, *the ego*, and *the superego* coexist simultaneously to attempt, regulate and adapt human behaviors; that is, human undergoes growth through internal debating among three personalities. The theory is in line with our goal to acquire the intuitive attempt, regular guidance, and detailed rectification via multi-agent framework. In this connection, we are motivated to implement the human psyche structure, and conceive a psyche structure for self-denial (i.e., PSSD) of LLMs, and further enhanced by exploiting tailored designs of multiple agents, thereby comprehensively activating LLMs' inherent capabilities.

Specifically, we design an *intuition-based id* role that functions as an innate drive, solely utilizing the LLMs' reasoning ability to answer the given question. Through directly generating multiple reasoning paths, the original attempts are collected as intuitive responses. Second, another LLM performs as the *rule-driven superego* role, embodying a rational entity which provides supporting references from methodological perspective. The rules, derived and processed empirically from training data, guide the role in abstracting key points of reasoning to support the denial. Third, the *script-centric ego* role serves as a mediator to bridge the aforementioned roles. This agent generates an executable script that adheres to intuitive attempts and key points, guiding its detailed execution to produce the final answer. Last but not least, as the novel cooperativity of roles in PSSD, different from other multi-agent methods, these roles can be unified in a single LLM to complete the procedure (namely PSSD-SFT). The results compared with methods in fine-tuning LLMs class demonstrate its superiority. Besides, the focus of PSSD lies in exploiting inherent abilities of LLMs, making it orthogonal to other solution categories and positioning it as a seamless integration into current research endeavors (ref. Section 4.5).

To summarize, our contributions in this work are as follows:

- To the best of our knowledge, we are the first to explore the inherent ability of LLMs in the line of leveraging mistakes;
- We introduce the idea of human psyche and propose a novel paradigm for LLMs, namely PSSD, comprising three roles, i.e., intuition-based id, rule-driven superego, and script-centric ego, to achieve self-denial of LLMs for further accurate generation;
- Extensive experiments demonstrate that the proposed method outperforms competitors in aforementioned three categories, and is effective in addressing the misleading problem.

2 Related Work

This section briefs relevant efforts from three perspectives.

2.1 Fine-tuning LLMs

Methods in this approach treat LLMs as a supervisor to obtain feedback for a provided mistake to fine-tune LLMs. STaR [28] emphasizes the rationales leading to the correct results. With the gold annotations, an iterative generation strategy is employed to obtain the ideal reasoning path for each question to fine-tune LLMs. LeMa [1] focuses on inaccurate reasoning paths, and employs GPT-4 to generate details, reasons, and final answer for them. These samples then perform as annotations for fine-tuning LLMs. Similar to LeMa, GWfS [2] directly guides LLMs to generate harmful responses and informs LLMs to evaluate its output with specific critique. This mistake analysis data is then used for model fine-tuning. Self-rethinking [19]pre-defines some error classes to provide LLMs with typical correction examples to prepare fine-tuning samples. SALAM [20] designs an assistant LLM to interact with the main LLM and leverages the resulting conversations to fine-tune assistant LLM, thereby enhancing the supervisor's flexibility compared with aforementioned approaches.

The utilization of domain fine-tuning can enhance the performance of LLMs; however, a new training phase encounters delayed response issue. Additionally, a substantial amount of computational powers are required to execute the learning process.

2.2 Leveraging tools

The main idea in this line is to leverage the feedback to verify LLMs' outputs. TRAN [23] accumulates rules from incorrect cases summarized by LLMs to form a rule base. When encountering new queries, it extracts relevant rules as external clues for the reasoning process. VE [30] transforms the attention to leverage external knowledge, e.g., Wikipedia and Google. The information performs as the facts to verify the results and supports the re-answer procedure. To put the thoughts deeply, ReAct [25] proposes an interaction framework to enforce LLMs to execute the thought-action-observation circle based on goolge engine. In each phrase, the model will make a feedback-dependent decision and prepare for the next state. Reflexion [17] tries reinforcement learning for reflection to further explore decision-making capability.

External assistance can help users to obtain immediate results; however, the performance is heavily reliant on the quality of knowledge source or effective tools. Once the assistance is inaccessible, the essential components of these methods are compromised.

2.3 Multi-agent Debate

Several agents are defined to invoke discussion within the same topic. LM vs LM [4] facilitates a multi-turn conversation between the claim-generating LM and an examining LM, which introduces questions to discover inconsistencies. Multiagent Debate [6] makes multiple agents propose and debate their individual responses and reasoning processes over multiple rounds to reach a common final answer. MAD [12] facilitates the expression of arguments among multiple agents in a "tit for tat" manner, while a judge oversees the debate process to obtain a final solution. Self-Contrast [29] employs diverse agents to offer distinct perspectives on a query. Subsequently, a new agent compares and summarizes discrepancies to generate the final answer.

The multi-agent debate framework effectively ensures timeliness and independence, but the omission of agents' self-denial results in perpetual resource competition. PSSD tries to tackle the identified challenges via the initial attempts, the regular guidance, and detailed executable steps from tailored roles.

3 Proposed Method

This section presents an overview of the proposed method and detailed design of each role.

3.1 Framework

As response to identified challenges, we follow the idea of human psyche and design three tailored roles. The overall framework of PSSD is illustrated in Figure 1.

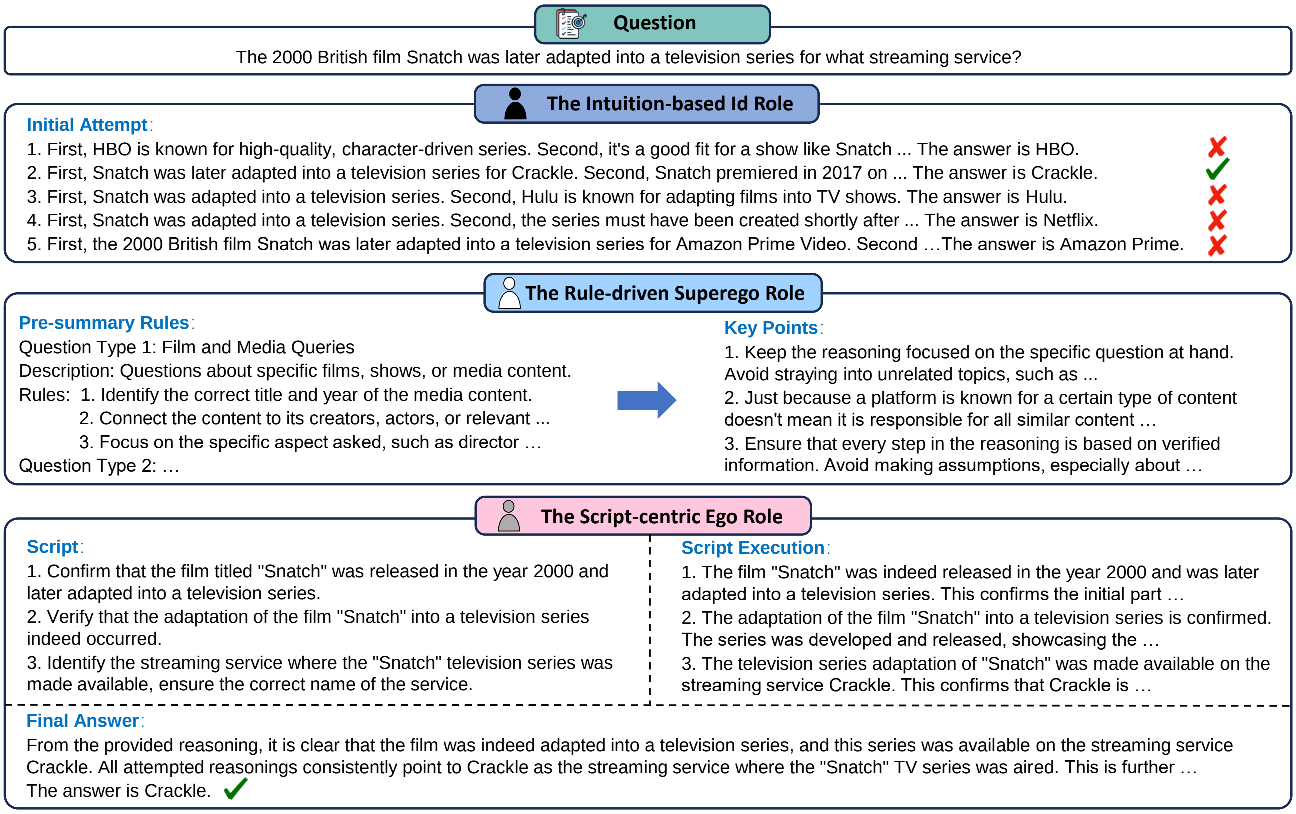


Figure 1: The framework of PSSD. It is mainly constituted of the id role, the superego role, and the ego role. Here we expect that the discussion of the three roles will enlighten LLMs to reason better.

For a given question, the intuition-based id role relies on LLMs' (e.g., GPT-4) inherent abilities to generate multiple reasoning paths (ref. Section 3.2). Subsequently, these intuitive attempts serve as fundamental materials for the rule-driven superego to judge and criticize. To enhance reliability of LLMs' self-supervision determination, we introduce general persuasive rules summarization based on the training samples from a methodological perspective. Considering that even for human beings it is challenging to definitively assert the correctness of the original answer, these rules mainly help the agent in abstracting guiding key points based on the specific question and its associated attempts (ref. Section 3.3). Third, the script-centric ego role comprehensively synthesizes question, original results, and key points to construct a script. The script provides step-by-step guidance for refining the potential mistakes in the reasoning process, thus facilitating the generation of the final answer (ref. Section 3.4).[1] Last but not least, according to the theory, the functions of above three roles possess sequential relevance. Consequently, we apply open-source LLMs (e.g., LLaMa and Mistral) to fine-tune these roles into an integration (ref. Section 3.5).

[1]We provide all prompts of PSSD on an anonymous GitHub repository: https://anonymous.4open.science/r/PSSD-41D7.

3.2 The Intuition-based Id Role

The intuition-based id role performs as an initial reasoning attempt on a given question by leveraging the LLM's inherent abilities, as defined by Freud [7], to capture its intuitive responses as a foundation for subsequent analysis.

Specifically, given a question q', we provide the LLM with a id role prompt (i.e., $\mathcal{M}_{Id}$) with several examples as below:

$$\mathbf{E} = \{\mathbf{E}_i\}_{i=1}^{z} = \{(q_i^t, a_i^t)\}_{i=1}^{z},$$

where z denotes the number of selected examples, q_i^t and a_i^t, used for few-shot learning, denote the i-th question along with its labelled answer from the training dataset.

We set the number of returned responses to l, and then $\mathcal{M}_{Id}$ generates the initial attempts, formally,

$$\mathbf{A} = \{a_j\}_{j=1}^{l} = \mathcal{M}_{Id}(\mathbf{E}, q'). \tag{1}$$

As shown in Figure 1, the intuition-based id role generates multiple reasoning paths based on the LLM's basic capabilities, wherein the quality and accuracy of results vary randomly. Through carefully comparing the similarities and differences among these paths, some delaying relevance might be obtained as the foundational reference for the next role.

3.3 The Rule-driven Superego Role

The rule-driven superego, representing internalized values and norms [7], is responsible for evaluating and regulating reasoning paths $\mathbf{A}$ to ensure that specific key points are accurately acquired. To strengthen the reliability of this process, we abstract a set of highly relevant rules as a guideline to support generating key points. These rules guide LLMs in identifying key points across various question types, thus providing a systematic framework that transcends individual responses. This framework helps LLMs overcome their limitations, especially when reasoning falls outside their factual knowledge base.

3.3.1 Development of Rules. A well-constructed set of rules is critical for effective key point generation. To develop these rules, we employ a contrastive approach using GPT-4. For each question q_b^t in the training set, we instruct GPT-4 to generate high-quality key points K_b^h, which are then manually reviewed for accuracy. In parallel, we instruct LLaMA-2-7B-Chat to generate suboptimal key points K_b^s. The comparison between these two sets of outputs provides a gradient that allows GPT-4 to learn what constitutes a high-quality key point. This process is divided into two stages: pattern extraction and rule summary.

Pattern Extraction. For each question q_b^t, we instruct GPT-4 to contrast high-quality key points K_b^h with suboptimal ones K_b^s to identify meaningful patterns P_b. These patterns explain the distinctions between the two key points, mathematically,

$$\begin{aligned} P_b &= \arg\max_b \ \mathrm{P}\left(\mathcal{T} \mid q_b^t, K_b^h, K_b^s\right) \\ &= \{t_1, t_2, \cdots, t_m\}, \end{aligned} \tag{2}$$

where $\mathcal{T}$ represents all possible patterns, and the top m patterns are selected. We extract patterns for each question in the training set, and this process results in a pattern set $\mathbf{P} = \{P_1, P_2, \cdots, P_{|P|}\}$.

Rule Summary. Based on extracted pattern set $\mathbf{P}$, we instruct GPT-4 to categorize all questions $\mathbf{Q}^t = \{q_1^t, q_2^t, \cdots, q_{|\mathbf{Q}^t|}^t\}$ into several types and generate corresponding rules for each type. These rules describe the key insights derived from patterns across similar questions, which can guide LLMs from a methodological perspective to better generate key points, formally,

$$\begin{aligned} U &= \arg\max_n \mathrm{P}\left(\mathcal{U} \mid \mathbf{Q}^t, \mathbf{P}\right) \\ &= \{u_1, u_2, \cdots, u_n\}, \end{aligned} \tag{3}$$

where $\mathcal{U}$ represents the rule set for all possible question types. The rules summarized by this method are used to guide LLMs in generating key points.[2]

3.3.2 Key Point Generation. Using a predefined superego role prompt, i.e., $\mathcal{M}_{Superego}$, we require the LLM to analyze reasoning attempts A and generate key points K for a given question q' under the guidance of rule U as below:

$$K = \mathcal{M}_{Superego}(U, q', \mathbf{A}). \tag{4}$$

As illustrated in Figure 1, the rule-driven superego role evaluates initial reasoning attempts, identifying incorrect logic (e.g., key point 2 in attempt 1) and guiding corrections in subsequent steps.

[2] The rules summarization process utilizes $m = 3$ and $n = 10$.

3.4 The Script-centric Ego Role

The script-centric ego role tries to mediate conflicts between the intuition-based id role and rule-driven superego role by striking a balance between the original attempts and the obtained key points. This process embodies the methodological guidance into a specific executable script that illustrates detailed steps for refining previous results, which briefly involves three main steps.

3.4.1 Script Generation. In detail, we provide a LLM with the predefined ego role prompt (i.e., $\mathcal{M}_{Ego}$) to analyze the initial attempt $\mathbf{A}$ and key points K. The step-wise script S is generated as below:

$$S = \mathcal{M}_{Ego}(q', \mathbf{A}, K). \tag{5}$$

The script synthesizes LLMs' reasoning capabilities with the summarized rules, distinguishing from key points in terms of levels. As depicted in Figure 1, key points emphasize what to do for enhancing the precise of the answer, such as "Ensure that every step in the reasoning is based on verified information", The script focuses on how to achieve these targets like "Verify that the adaptation of the film 'Snatch' into a television series indeed occurred".

3.4.2 Script Execution. In this step, the script-centric ego role executes the script step-by-step, ensuring that LLMs follow the reasoning path outlined in the generated script. In order to keep the execution process smoothly, we integrate the script with the key points as input to guide $\mathcal{M}_{Ego}$, formally,

$$S' = \mathcal{M}_{Ego}(q', K, S), \tag{6}$$

where S' denotes the answered script as illustrated in Figure 1.

After the execution, the script further confirms details relevant to the question and provides supplementary descriptions. These references contribute to the accuracy of the final decision.

3.4.3 Answering. Finally, the script-centric ego role formulates the final answer based on all relevant process information generated in the previous steps. Specifically, we instruct the LLM to synthesize the question q', key points K, script S, and script execution S' to generate the final result R. This process is formalized as:

$$R = \mathcal{M}_{Ego}(q', K, S, S'). \tag{7}$$

3.5 The Merge of Three Roles

To mitigate resource competition in multi-agent debate, we further explore effective management strategy for frequency of invoking LLMs. In accordance with the internal unity of three roles, they are merged into a whole through the utilization of open-source LLMs. As fine-tuning process performs as an adhesive, combining distinct roles, the modified method is called PSSD-SFT to differentiate it from PSSD. Subsequently, we discuss the construction of the training dataset, along with the training and reasoning processes involved in PSSD-SFT in greater detail.

3.5.1 Construction of Training Dataset. First of all, the fine-tuning process is essential a supervised learning procedure, thereby necessitating annotated samples as training labels. To construct required dataset, we use publicly available reasoning datasets as the data source.[3] For each question q_b^t in training set, we apply GPT-4 as the fundamental model of PSSD framework to generate reasoning records, which include the initial attempts, key points, script, script execution, and final answer.

To ensure accuracy, these reasoning records undergo a rigorous annotation process:

- Initial Annotation. Three graduate students initially annotate these reasoning records and carefully review and improve them based on the groundtruth.
- Review and Consensus. Each record is thoroughly reviewed by an additional annotator. Any discrepancies are thoroughly discussed until a consensus is reached on the final annotation.
- Structured Combination. Each annotated reasoning record is integrated into a predefined training sample template.

Finally, This process can be formally defined as follows:

$$d_b = Template[q_b^t \oplus \mathbf{A}_b \oplus K_b \oplus S_b \oplus S_b' \oplus R_b],$$

where q_b^t, $\mathbf{A}_b$, K_b, S_b, S_b' and R_b denote the given question, corresponding initial attempts, key points, script, script execution and final answer respectively. $\oplus$ denotes the operation of concatenation.

3.5.2 Fine-Tuning LLMs. To further conserve computational resources, we employ the LoRA [10] method for fine-tuning the LLMs. On one hand, considering the intuition-based id role purely relies on the capability of the fundamental model, its enhancement emphasizes the improvement of LLMs. Therefore, we fine-tune the first LoRA model using question-reasoning path pairs as training data. This process can be formulated as:

$$W_1 : \mathcal{L}_1 = -\frac{1}{L}\sum_{i=1}^{L} \log P(x_i | q_b^t, a_b^{<i}), \tag{8}$$

where W_1 represents weights of the first LoRA model, $\mathcal{L}_1$ represents its loss function, L is the token length of the sequence a_b, x_i is the currently predicted response token, q_b^t and $a_b^{<i}$ are the question and the response tokens before x_i.

On the other hand, acquired structured $\mathbf{D} = \{d_1, d_2, \ldots, d_{|D|}\}$ aims to provide supervised signals for training other more complicated roles (i.e., rule-driven superego and script-centric ego). Thus, it is employed to fine-tune another LoRA model focusing the generation of elements in d, formally,

$$W_2 : \mathcal{L}_2 = -\frac{1}{L}\sum_{i=1}^{L} \log P(x_i | q_b^t, \mathbf{A}_b, d_b^{<i}), \tag{9}$$

where $\mathbf{A}_b$ and $d_b^{<i}$ are the initial attempt with 5 reasoning paths and the response tokens before x_i.

This fine-tuning strategy strengthens LLMs not only in generating diverse reasoning paths during initial attempts but also in developing multi-role adaptive reasoning capabilities.

The parameters of these two LoRA models are ultimately merged into the original fundamental model, indicating that only one model supports the operation of PSSD-SFT, mathematically,

$$W = W_0 + W_1 + W_2 \tag{10}$$

where W and W_0 represent the weights of the merged model $\mathcal{M}_{LoRA}$ and the original weights of the LLM.

Specifically, the complete reasoning process in PSSD-SFT involves two main steps. For a given input question q', PSSD-SFT first generates l initial reasoning paths $\mathbf{A}$. Then, using both q' and $\mathbf{A}$ as inputs, it sequentially generates the key points K, script S, script execution S', and finally produces the final answer R, as below:

$$\mathcal{M}_{LoRA} : q' \rightarrow \mathbf{A} \rightarrow K, S, S', R. \tag{11}$$

where $\mathcal{M}_{LoRA}$ denotes the merged LoRA model with weight W.

4 Experiments

This section provides a detailed presentation of the experiments, along with an in-depth analysis of the results.

4.1 Experiment Setup

4.1.1 Datasets. We conduct experiments on both textual and mathematical reasoning tasks using four datasets: AdvHotpotQA, 2WikiMultiHopQA, GSM8K, and MATH.

For textual reasoning, we utilize the following datasets:

- AdvHotpotQA[4] [26] is a challenging subset derived from the multi-hop question answering dataset HotpotQA [24], where the correct and incorrect predictions are balanced;
- 2WikiMultiHopQA[5] [9] is a multi-hop question answering dataset that leverages the structured format of Wikidata and applies logical rules.

For mathematical reasoning, we use:

- GSM8K[6] [3], a benchmark dataset of grade-school-level math word problems, designed to evaluate mathematical reasoning. Each question is accompanied by a detailed step-by-step solution.
- MATH[7] [8] is a collection of challenging high school-level math problems, aimed at evaluating advanced mathematical reasoning and problem-solving skills.

4.1.2 Competitors. To put our results in perspective, we apply two classes of competitive baselines for demonstration.

In the first class, we select representative methods from leveraging-tools and multi-agent debate categories to evaluate PSSD, as these methods commonly employ closed-source LLMs (e.g., GPT-4) as the fundamental model: (1) Standard Prediction (Stand) directly predicts the answer for input using manually provided examples; (2) Original CoT (CoT-Ori) [22] generates a reasoning path before predicting the final answer; (3) CoT with Self-Consistency (CoT-SC) [21]samples five reasoning paths and selects the final answer based on consistency values; (4) ReAct [25] enhances CoT reasoning by integrating Wikipedia API to improve factual accuracy; (5) VE [30] is a post-editing framework that leverages external knowledge to increase the factual accuracy of predictions; (6) Self-Contrast [29] is a self-contrast framework that compares multiple solution perspectives by multiple agents to re-examine and eliminate mistakes.

In the second class, we compare PSSD-SFT with the following state-of-the-art fine-tuning methods based on the same open-source

[3]For textual reasoning, we select AdvHotpotQA and 2WikiMultiHopQA as data source, while for mathematical reasoning, we choose GSM8K and MATH as data source.

[4]https://github.com/xiye17/TextualExplInContext.
[5]https://github.com/Alab-NII/2wikimultihop.
[6]https://github.com/openai/grade-school-math.
[7]https://github.com/hendrycks/math.

Dataset	Method	EM	△ EM	PM	△ PM	RM	AvgT
AdvHotpotQA	Stand	36.36	-	-	-	-	1.81
	CoT-Ori	42.86	-	-	-	-	2.89
	CoT-SC	42.53	-	72.73	-	0.00	4.31
	ReAct	44.81	+2.28	59.46	-13.27	10.97	14.08
	VE	44.16	+1.63	60.81	-11.92	7.74	23.78
	Self-Contrast	45.13	+2.60	68.53	-4.20	9.84	34.64
	PSSD	**47.08**	**+4.55**	**74.32**	**+1.59**	**11.08**	19.80
2WikiMultiHopQA	Stand	34.82	-	-	-	-	1.80
	CoT-Ori	37.20	-	-	-	-	2.92
	CoT-SC	40.18	-	68.37	-	0.00	4.11
	ReAct	**45.24**	**+5.06**	65.84	-2.53	**14.32**	16.09
	VE	42.26	+2.08	62.16	-6.21	11.36	20.73
	Self-Contrast	41.37	+1.07	70.67	+2.30	9.64	29.45
	PSSD	41.96	+1.78	**72.43**	**+4.06**	9.48	19.14
GSM8K	Stand	88.00	-	-	-	-	2.03
	CoT-Ori	93.60	-	-	-	-	4.87
	CoT-SC	93.20	-	76.27	-	0.00	7.68
	Self-Contrast	95.40	+2.20	**86.71**	**+10.44**	10.00	49.58
	PSSD	**96.80**	**+3.60**	84.75	+8.48	**22.22**	34.14
MATH	Stand	65.20	-	-	-	-	2.38
	CoT-Ori	73.40	-	-	-	-	5.34
	CoT-SC	72.80	-	70.53	-	0.00	7.83
	Self-Contrast	76.00	+3.20	78.36	+7.83	8.92	53.92
	PSSD	**77.20**	**+4.40**	**80.18**	**+9.65**	**13.17**	36.41

Table 1: Overall results (%) of PSSD on GPT-4. △ denotes the improvement from CoT-SC baseline. The best results in each dataset are highlighted in bold. AvgT denotes the average time (in seconds) required for each individual reasoning process. - denotes the corresponding information does not exist. Considering LLMs make only a single attempt in both the Stand and CoT-Ori settings, the PM value is not applicable in these cases. Since external knowledge bases utilized by ReAct and VE cannot support math computation, they are not conducted in mathematical reasoning datasets. Details in Section 4.2.

LLMs (e.g., LLaMA): (1) CoT Fine-tuning fine-tunes the target model using the CoT reasoning paths from the target dataset; (2) Mistake Tuning [19] fine-tunes the model using both correct and incorrect reasoning paths to improve error correction; (3) AugGPT [5] generates training data using the specific sample as a seed, and then fine-tunes the target model on the generated data; (4) LEC [27] utilizes error-prone samples from the target model as seeds to generate training data, followed by fine-tuning on the data.

All experiments adhered to the default hyper-parameters reported in their papers. Fine-tuning experiments are conducted using the LoRA method to optimize computational efficiency.

4.1.3 Metrics. We evaluate the performance of PSSD via three metrics: (1) Exact Match (EM) measures the percentage of predictions that exactly match the ground truth, which evaluates the answering ability of models; (2) Potential Match (PM) measures the conditional probability of accurately selecting the correct answer from original attempts wherein the gold answer already exists, which evaluates the ability to distinguish between correct and incorrect results; (3) Rectified Match (RM) measures the conditional probability of original incorrect predictions that is finally corrected, which evaluates the capability to correct.

4.1.4 Implementations. To evaluate the effectiveness of PSSD and PSSD-SFT, we select both leading closed-source and powerful open-source LLMs as baselines. For closed-source LLMs, we use GPT-4-turbo [8] [14], accessed via its APIs. For open-source LLMs, select the powerful open-source LLMs Mistral-7B-Instruct-v0.2 [9] and LLaMA-3-8B-Instruct [10] as baseline models. In order to ensure fairness in our experiments, the same foundation is utilized to support competitors for the comparison. The experiments are performed with 2 * RTX 3090. During the training phase of PSSD-SFT, we utilize the AdamW optimizer [13] with $\beta_1 = 0.9$ and $\beta_2 = 0.999$. The learning rate is set to 1e-6, with a 0.1 ratio of warm-up steps and linear decay. We configure the maximum input length to 4,096 tokens and establish a training batch size of 4. The entire training process is completed within 4.5 hours, and we employ the final checkpoint for subsequent evaluations.

4.2 PSSD Results

As shown in Table 1, PSSD consistently achieves state-of-the-art results across nearly all tasks in all metrics, demonstrating the superiority and generalizability of our design. The methods, such as Stand, CoT-Ori, and CoT-SC, only employ LLMs without any additional architectural modifications. Thus, their average reasoning

[8]We use GPT-4-turbo-2024-04-09 as default GPT-4-turbo version
[9]https://huggingface.co/mistralai/Mistral-7B-Instruct-v0.2
[10]https://huggingface.co/meta-llama/Meta-Llama-3-8B-Instruct

Method	Model	Adv.	Wiki.	GSM8K	MATH
CoT Fine-Tuning	Mistral-7B-Instruct	30.84	30.06	58.00	41.80
AugGPT		**32.47**	29.46	54.40	**43.40**
LEC		29.87	28.27	54.00	41.20
Mistake Tuning		30.19	**30.95**	58.60	43.20
PSSD-SFT		32.14	30.65	**59.20**	**43.40**
CoT Fine-Tuning	LLaMA-3-8B-Instruct	31.50	**32.44**	79.50	42.20
AugGPT		32.14	30.65	77.80	39.20
LEC		30.19	28.86	80.20	38.40
Mistake Tuning		30.52	31.25	80.60	43.80
PSSD-SFT		**32.47**	32.14	**80.80**	**44.60**

Table 2: The results in EM (%) of PSSD-SFT. Adv. means AdvHotpotQA dataset and Wiki. means 2WikiMultiHopQA dataset. The best results in each dataset are highlighted in bold, while the second positions are underlined. Details in Section 4.3.

time exhibits a low value and should be considered solely for reference purposes. Interestingly, the introduction of CoT-SC does not significantly improve the EM value over CoT-Ori; in fact, it reduces the EM value on three datasets, except for 2WikiMultiHopQA. This outcome aligns with human intuition, as CoT-SC merely repeats the original inference multiple times without considering underlying mistakes. These facts demonstrate the advantage of distinct psychological roles of PSSD on enhancing accuracy of the reasoning, wherein it achieves 4.00% increase over CoT-Ori in EM on average (for simplicity we will omit the "average" in the follows if it refers to the mean value of all accessible datasets).

In both ReAct and VE, CoT-SC is applied to generate consistency values for each question, and then external knowledge base rectifies the filtered generations. The direct application of Wikipedia can account for the 4.17% and 0.3% higher EM performance of ReAct and VE compared to PSSD in 2WikiMultiHopQA. The comparison of ranking positions in AdvHotpotQA and 2WikiMultiHopQA further illustrates that the performance of methods in leveraging tools heavily relies on the quality of external facts. Once the supporting references cannot meet the requirement, their performance experiences a cliff decline, as results in AdvHotpotQA. Even so, PSSD outperforms ReAct by 10.75% and VE by 11.89% in PM, indicating its effectiveness in stimulating LLMs for mistake awareness. And the higher values in RM (except for 2WikiMultiHopQA) of PSSD demonstrate the correcting ability of our designs.

PSSD beats Self-Contrast by 1.32% in EM, which is in accordance with our anticipation, since this typical multi-agent debate merely assigns several roles to discuss and generate the answer. In comparison with PSSD, without consideration of the self-denial, it falls short on distinguishing correct generations from incorrect ones (a 1.85% drop in PM) and rectifying original total wrong attempts (a 4.39% drop in RM). When it comes to the mentioned resource competition, PSSD gets the reduction reasoning time by 14.7s below Self-Contrast, saving the time cost. The number of agent (3) and times of evoking LLMs (5) are also superior to those of Self-Contrast (4, 7.8), as listed in Table 5. These findings strongly supports the low-resource requirements of our three psychological tailored roles.

As aforementioned, the detailed analysis of the comparison with approaches in fine-tuning LLMs is provided in Section 4.3.

Id	Superego-R	Superego	Ego	Adv.	GSM8K
✓	✓	✓	✓	**47.08**	**96.80**
✓	✓	✓	×	45.45	95.60
✓	✓	×	×	44.81	95.00
✓	×	×	×	42.53	93.20

Table 3: Ablation studies on PSSD in EM (%) employing GPT-4. Adv. means AdvHotpotQA dataset. Id represents the intuition-based id role, selecting the final answer through self-consistency. Superego-R represents the rule-driven superego role without using summarized rules to generate key points. Superego represents the full superego role. Ego represents the script-based reasoning of the script-centric ego role. Details in Section 4.4

4.3 PSSD-SFT Results

PSSD-SFT achieved competitive performance on all benchmarks, particularly excelling in mathematical reasoning, where it delivered state-of-the-art results.

In comparison, the results of AugGPT and LEC indicate that additional LLM-generated training data does not always yield positive outcomes. For instance, on 2WikiMultiHopQA and GSM8K, this approach even underperformed compared to the dataset's built-in CoT training set. The result might be attributed to biases introduced by LLM-generated data, which can impair the fine-tuned LLMs' specific reasoning capabilities. Mistake Tuning shows promise by enhancing LLMs' ability to distinguish between correct and incorrect responses. However, the contrastive-style fails to approach LLMs' self-denial mechanism. Therefore, the model merely learns superficial features of mistakes.

In contrast, PSSD-SFT contains three psychological roles with their tailored fine-tuning methods (i.e., two stages), aiming to stimulate LLMs' self-denial. The merged LoRA model proves its significance through the superior results. The outperformance, meanwhile, indicates that the integration of roles from PSSD into an entity (smaller open-source LLMs) through fine-tuning is both a viable and effective strategy. Furthermore, the resource-friendliness feature of our design is also validated according to the numbers of agents, debate rounds, and times of invoking LLMs (ref. Table 5).

4.4 Ablation Study

To analyze effects of different roles in PSSD, we perform the ablation study and results is presented in Table 3. Due to the sequential feature of PSSD, in each setting, we instruct LLMs to derive a final answer based on the information generated by remaining roles.

In comparison with the complete model, the removal of specific roles results in performance drop. Notably, the whole rule-driven superego role provides the increase of 1.92% in AdvHotpotQA and 2.40% in GSM8K, which supports the significance of first identified challenge. Once the model is able to confidentially and convincingly supervise its generation, it starts to engage in self-denial. The fact that the introduction of summarized rules improve performance by 0.64% in AdvHotpotQA and 0.60% in GSM8K indicates a strongly positive correlation between persuasiveness of references and the accuracy of results. Moreover, the equipment of the script-centric ego role contributes to a 1.63% improvement in AdvHotpotQA and a 1.20% improvement in GSM8K, illustrating the value of handling

Method	AdvHotpotQA	2WikiMultiHopQA
ReAct	44.81	45.24
VE	44.16	42.26
PSSD	47.08	41.96
PSSD + ReAct	**48.05** (↑ 0.97)	**45.54** (↑ 3.58)
PSSD + VE	**48.05** (↑ 0.97)	42.86 (↑ 0.90)

Table 4: Compatibility Study of PSSD in EM (%). The best results in each dataset are highlighted in bold.↑ denotes the improvement from PSSD. Details in Section 4.5.

the second challenge identified. After discerning potential mistakes, the machine should further know targeted steps for rectifying initial attempts, which is effectively tackled by our script design.

Overall, the study demonstrates the indispensable nature of roles in PSSD, with each fulfilling a distinct and crucial function based on Freud's psychological theory. The contribution of each role during the self-denial process differentiates, and the combine efforts ensure that PSSD effectively enhances LLMs' reasoning capabilities.

4.5 Compatibility Study

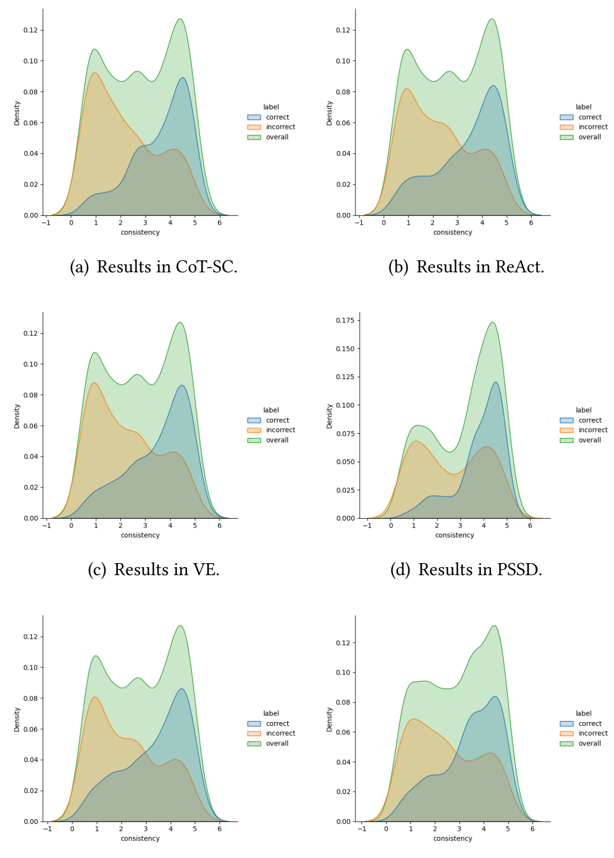
To demonstrate that PSSD is seamlessly orthogonal to other category, we evaluate it by incorporating leveraging-tools methods, such as VE and ReAct. During the final answer generation step in PSSD, we set the number of returned candidate texts to 5 and apply above methods to assist in answering questions when its consistency values fall below 3. The result is provided in Table 4.

The results demonstrate that when combined with VE or ReAct, PSSD can significantly enhance the reasoning capabilities of LLMs. This illustrates strong compatibility or PSSD with external retrieval systems, thereby highlighting its adaptability and effectiveness in leveraging additional knowledge. The fact suggests a future direction for further optimizing the proposed methodology. Besides, the observed increase (i.e., 3.58%) in 2WikiMultiHopQA combined with ReAct and the highest result substantiate our hypothesis that the leverage of Wikipedia determines the best result of ReAct in 2WikiMultiHopQA (ref. Section 4.2).

4.6 Consistency Analysis

To explore the confidence of PSSD in its determination, we use kernel density estimation [21] to analyze the confidence distribution of different methods in generating outputs. The results, visualized in Figure 2, show distinct distribution patterns for each method.

From a global perspective, CoT-SC, ReAct, and VE present a bimodal distribution, while PSSD displays a right-skewed distribution. The distribution illustrates that PSSD instills greater confidence in its determinations, which might be derived from the rule-driven superego role and the script-centric ego role. More specifically, PSSD exhibits a predominantly right-skewed distribution, with the highest peak value in correct samples compared to others. In incorrect samples, however, the distribution presents a bimodal feature, indicating that PSSD still make wrong prediction with higher confidence. The situation is mitigated when integrated with ReAct and VE, and we leave the further improvement in this direction as the future work.



(a) Results in CoT-SC.
(b) Results in ReAct.
(c) Results in VE.
(d) Results in PSSD.
(e) Results in PSSD +ReAct.
(f) Results in PSSD +VE.

Figure 2: Sketch of the consistency distribution in different methods. Consistency denotes the models confidence on predictions, and density denotes the probability density. We present the results in AdvHotpotQA for illustration. Details in Section 4.6.

5 Conclusion

In this paper, we propose PSSD a novel and comprehensive approach to enhance reasoning abilities of LLMs via acquiring self-denial. By identifying the challenges, our method builds on the idea of human psyche structure and introduces three tailored roles. The intuition-based id role tries to provide initial attempts based on LLMs. Subsequently, the rule-driven superego role aims to increase the precision of judgement via summarized rules. The key points of a specific question are generated as clues for the next phase. Finally, the script-centric ego role focuses on executable scripts to guide the final refinement, wherein it synthesizes attempts and key points to complete detailed execution. Besides, we merge three roles into an integration by proposing two-stage fine-tuning strategy to evaluate the resource-friendliness of PSSD. Comprehensive experiments demonstrate that PSSD and PSSD-SFT outperform competing models in all primary category, and can be fused to retrieve systems for further enhancement in reasoning accuracy.

References

[1] Shengnan An, Zexiong Ma, Zeqi Lin, Nanning Zheng, Jian-Guang Lou, and Weizhu Chen. 2023. Learning From Mistakes Makes LLM Better Reasoner. *CoRR* abs/2310.20689 (2023).

[2] Kai Chen, Chunwei Wang, Kuo Yang, Jianhua Han, Lanqing Hong, Fei Mi, Hang Xu, Zhengying Liu, Wenyong Huang, Zhenguo Li, Dit-Yan Yeung, Lifeng Shang, Xin Jiang, and Qun Liu. 2023. Gaining Wisdom from Setbacks: Aligning Large Language Models via Mistake Analysis. *CoRR* abs/2310.10477 (2023).

[3] 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. 2021. Training Verifiers to Solve Math Word Problems. *CoRR* abs/2110.14168 (2021).

[4] Roi Cohen, May Hamri, Mor Geva, and Amir Globerson. 2023. LM vs LM: Detecting Factual Errors via Cross Examination. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*. Association for Computational Linguistics, 12621–12640.

[5] Haixing Dai, Zhengliang Liu, Wenxiong Liao, Xiaoke Huang, Zihao Wu, Lin Zhao, Wei Liu, Ninghao Liu, Sheng Li, Dajiang Zhu, Hongmin Cai, Quanzheng Li, Dinggang Shen, Tianming Liu, and Xiang Li. 2023. ChatAug: Leveraging ChatGPT for Text Data Augmentation. *CoRR* abs/2302.13007 (2023).

[6] Yilun Du, Shuang Li, Antonio Torralba, Joshua B. Tenenbaum, and Igor Mordatch. 2023. Improving Factuality and Reasoning in Language Models through Multiagent Debate. *CoRR* abs/2305.14325 (2023).

[7] Sigmund Freud. 1923. *Das Ich und das Es*. Gesammelte Werke, Vol. XIX. Internationaler Psychoanalytischer Verlag, Leipzig, Vienna and Zurich.

[8] Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021. Measuring Mathematical Problem Solving With the MATH Dataset. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks 1, NeurIPS Datasets and Benchmarks 2021, December 2021, virtual*, Joaquin Vanschoren and Sai-Kit Yeung (Eds.).

[9] Xanh Ho, Anh-Khoa Duong Nguyen, Saku Sugawara, and Akiko Aizawa. 2020. Constructing A Multi-hop QA Dataset for Comprehensive Evaluation of Reasoning Steps. In *Proceedings of the 28th International Conference on Computational Linguistics, COLING 2020, Barcelona, Spain (Online), December 8-13, 2020*. International Committee on Computational Linguistics, 6609–6625.

[10] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-Rank Adaptation of Large Language Models. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.

[11] Tiffany A Ito, Jeff T. Larsen, N. Kyle Smith, and John T. Cacioppo. 1998. Negative information weighs more heavily on the brain: the negativity bias in evaluative categorizations. *Journal of personality and social psychology* 75 4 (1998), 887–900.

[12] Tian Liang, Zhiwei He, Wenxiang Jiao, Xing Wang, Yan Wang, Rui Wang, Yujiu Yang, Zhaopeng Tu, and Shuming Shi. 2023. Encouraging Divergent Thinking in Large Language Models through Multi-Agent Debate. *CoRR* abs/2305.19118 (2023).

[13] Ilya Loshchilov and Frank Hutter. 2019. Decoupled Weight Decay Regularization. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

[14] OpenAI. 2023. GPT-4 Technical Report. *CoRR* abs/2303.08774 (2023).

[15] Arkil Patel, Satwik Bhattamishra, and Navin Goyal. 2021. Are NLP Models really able to Solve Simple Math Word Problems?. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021, Online, June 6-11, 2021*. Association for Computational Linguistics, 2080–2094.

[16] Freda Shi, Xinyun Chen, Kanishka Misra, Nathan Scales, David Dohan, Ed H. Chi, Nathanael Schärli, and Denny Zhou. 2023. Large Language Models Can Be Easily Distracted by Irrelevant Context. In *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA (Proceedings of Machine Learning Research, Vol. 202)*. PMLR, 31210–31227.

[17] Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. 2023. 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*.

[18] Alon Talmor, Ori Yoran, Ronan Le Bras, Chandra Bhagavatula, Yoav Goldberg, Yejin Choi, and Jonathan Berant. 2021. CommonsenseQA 2.0: Exposing the Limits of AI through Gamification. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks 1, NeurIPS Datasets and Benchmarks 2021, December 2021, virtual*.

[19] Yongqi Tong, Dawei Li, Sizhe Wang, Yujia Wang, Fei Teng, and Jingbo Shang. 2024. Can LLMs Learn from Previous Mistakes? Investigating LLMs' Errors to Boost for Reasoning. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, Lun-Wei Ku, Andre Martins, and Vivek Srikumar (Eds.). Association for Computational Linguistics, 3065–3080.

[20] Danqing Wang and Lei Li. 2023. Learning from Mistakes via Cooperative Study Assistant for Large Language Models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*. Association for Computational Linguistics, 10667–10685.

[21] Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V. Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2023. Self-Consistency Improves Chain of Thought Reasoning in Language Models. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

[22] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. 2022. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. In *NeurIPS*.

[23] Zeyuan Yang, Peng Li, and Yang Liu. 2023. Failures Pave the Way: Enhancing Large Language Models through Tuning-free Rule Accumulation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*. Association for Computational Linguistics, 1751–1777.

[24] Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W. Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. 2018. HotpotQA: A Dataset for Diverse, Explainable Multi-hop Question Answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*. Association for Computational Linguistics, 2369–2380.

[25] Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R. Narasimhan, and Yuan Cao. 2023. ReAct: Synergizing Reasoning and Acting in Language Models. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

[26] Xi Ye and Greg Durrett. 2022. The Unreliability of Explanations in Few-shot Prompting for Textual Reasoning. In *NeurIPS*.

[27] Jiahao Ying, Mingbao Lin, Yixin Cao, Wei Tang, Bo Wang, Qianru Sun, Xuanjing Huang, and Shuicheng Yan. 2024. LLMs-as-Instructors: Learning from Errors Toward Automating Model Improvement. *CoRR* abs/2407.00497 (2024).

[28] Eric Zelikman, Yuhuai Wu, Jesse Mu, and Noah D. Goodman. 2022. STaR: Bootstrapping Reasoning With Reasoning. 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*.

[29] Wenqi Zhang, Yongliang Shen, Linjuan Wu, Qiuying Peng, Jun Wang, Yueting Zhuang, and Weiming Lu. 2024. Self-Contrast: Better Reflection Through Inconsistent Solving Perspectives. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*. Association for Computational Linguistics, 3602–3622.

[30] Ruochen Zhao, Xingxuan Li, Shafiq Joty, Chengwei Qin, and Lidong Bing. 2023. Verify-and-Edit: A Knowledge-Enhanced Chain-of-Thought Framework. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*. Association for Computational Linguistics, 5823–5840.

[31] Denny Zhou, Nathanael Schärli, Le Hou, Jason Wei, Nathan Scales, Xuezhi Wang, Dale Schuurmans, Claire Cui, Olivier Bousquet, Quoc V. Le, and Ed H. Chi. 2023. Least-to-Most Prompting Enables Complex Reasoning in Large Language Models. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

[32] Andy Zou, Zifan Wang, J. Zico Kolter, and Matt Fredrikson. 2023. Universal and Transferable Adversarial Attacks on Aligned Language Models. *CoRR* abs/2307.15043 (2023).

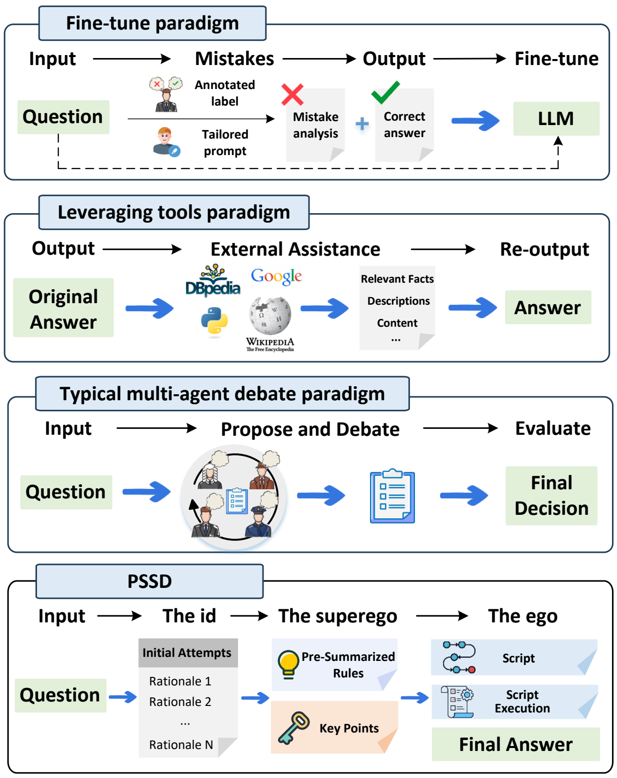


Figure 3: The comparison of different paradigms.

A Complementary Experiments

A.1 Resource Analysis

Method	# Agent Number	# Debate Round	# Call
Self-Contrast	4	1.0	7.8
LM vs LM	2	2.9	8.9
MAD	3	1.2	4.6
Multiagent Debate	3	2.0	6.0
PSSD	3	1.0	5.0
PSSD-SFT	1	1.0	2.0

Table 5: The configuration and the average number of API/LLM calls of multi-agent debate methods.

A.2 PSSD vs. Self-Contrast

PSSD inspires LLMs to engage in human psyche by adopting a multi-agent debate paradigm that is based on the distinct roles of intuition-based id, rule-driven superego, and script-centric ego, rather than simply stacking numerous roles or iterating. The underlying assumption is that PSSD surpasses Self-Contrast in terms of both accuracy and stability. To validate this assumption, we conduct an experiment using 200 samples from the AdvHotpotQA dataset, comparing the results of PSSD and Self-Contrast on each sample.

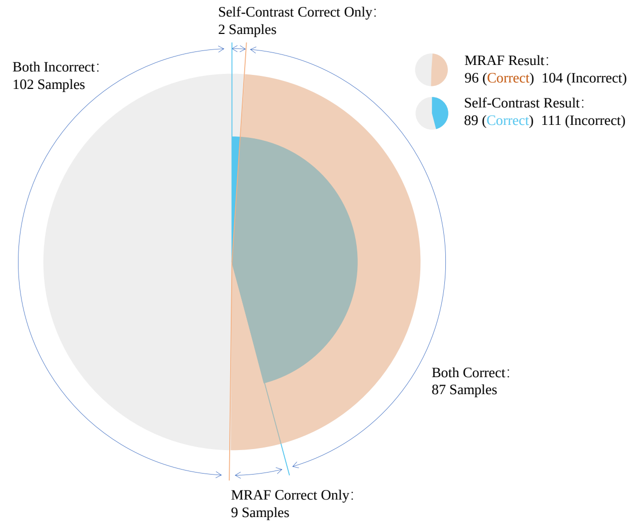


Figure 4: We compare the results of the PSSD and Self-Contrast using two pie charts. It shows PSSD is more accurate and stable than direct Self-Contrast.

As illustrated in Figure 4, PSSD achieves higher accuracy with 96 correct answers, compared to 89 correct answers for Self-Contrast.

We further categorize the results into four distinct cases: (1) Both methods correctly answer; (2) PSSD correctly answers, while Self-Contrast fails; (3) PSSD fails, while Self-Contrast correctly answers; (4) Both methods fail. The results show that when Self-Contrast arrives at a correct solution, PSSD typically achieves the same outcome, except for two instances where Self-Contrast succeeded while PSSD did not. However, in 9 instances, PSSD succeeded where Self-Contrast failed, further highlighting PSSD's reliability. These results underscore that PSSD not only improves accuracy but also enhances stability, making it a more reliable framework for LLMs reasoning compared to Self-Contrast.

A.3 Answering Ability Analysis

The focus lies in harnessing the potential of LLMs through learning from mistakes, while it remains crucial to accurately discern the original questions that can be correctly answered. Hence, extensive experiments are conducted to closely observe the situation, as depicted in Figure 5.

Evidently, PSSD consistently achieves accurate predictions across all answer types and maintains a high level of consistency to the original correct determination (as shown in the right bar). For the questions in missing answer type, PSSD provides more correct predictions than the incorrect ones while minimizing the number of unresolved cases. This indicates that the proposed method enables LLMs to reflect on their generations and carefully select the most satisfactory one. And we leave the efforts on increasing correct hits to the future work. Additionally, the presence of three distinct types of outcomes within the original pure incorrect answer type demonstrates that PSSD effectively enhances LLMs by encouraging more advantageous attempts.

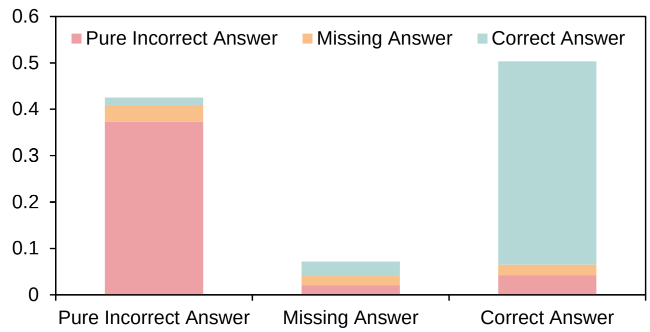


Figure 5: Sketch of the answering type distribution between CoT-SC and PSSD in AdvHotpotQA. The presence of the missing answer indicates incorrect results including exact answers, thereby contributing to the overall count of incorrect answers. The height of each bar means the corresponding answer type in CoT-SC, and the inside detailed types come from the results of PSSD. Details in Section A.3.

B Detailed Statistics for Data

Dataset	Adv.	Wiki.	GSM8K	MATH
Size	2,620	2,554	3,087	3,092
- *Training*	2,312	2,218	2,587	2,592
- *Test*	308	336	500	500
# Key Points	7,781	7,636	9,631	10,389
- *Avg.*	2.97	2.99	3.12	3.36
# Scripts	8,124	7,946	12,903	13,488
- *Avg.*	3.10	3.11	4.18	4.36
# Script Executions	8,118	7,932	10,681	11,251
- *Avg.*	3.10	3.11	3.46	3.64

Table 6: Detailed statistics for the training and test splits of the datasets we used. Adv. means AdvHotpotQA dataset and Wiki. means 2WikiMultiHopQA dataset. # Key Points represent the total number of key points contained in each question, while the Average represents the average number of key points per question (calculated in the same way for both Scripts and Script Executions).

Received 20 February 2007; revised 12 March 2009; accepted 5 June 2009