IDEA: Enhancing the Rule Learning Ability of Large Language Model Agent through Induction, Deduction, and Abduction

Kaiyu He, Mian Zhang, Shuo Yan, Peilin Wu, Zhiyu Zoey Chen
Department of Computer Science
University of Texas at Dallas
{kaiyu.he, zhiyu.chen2}@utdallas.edu

Abstract

While large language models (LLMs) have been thoroughly evaluated for deductive and inductive reasoning, their proficiency in holistic rule learning in interactive environments remains less explored. We introduce RULEARN, a novel benchmark to assess the rule-learning abilities of LLM agents in interactive settings. In RULEARN, agents strategically interact with simulated environments to gather observations, discern patterns, and solve complex problems. To enhance the rule-learning capabilities for LLM agents, we propose IDEA, a novel reasoning framework that integrates the process of **I**nduction, **DE**duction, and **A**bduction. The IDEA agent generates initial hypotheses from limited observations through abduction, devises plans to validate these hypotheses or leverages them to solve problems via deduction, and refines previous hypotheses through induction, dynamically establishing and applying rules that mimic human rule-learning behaviors. Our evaluation of the IDEA framework, which involves five representative LLMs, demonstrates significant improvements over the baseline. Furthermore, our study with human participants reveals notable discrepancies in rule-learning behaviors between humans and LLMs. We believe our benchmark will serve as a valuable and challenging resource, and IDEA will provide crucial insights for the development of LLM agents capable of human-like rule learning in real-world scenarios. Our code and data have been released on GitHub.[1]

1 Introduction

One major pillar of human intelligence is the ability to discern and apply rules. We identify patterns, form hypotheses, and iteratively refine them through interaction with the environment—a process that traditionally involves three stages: abduction, deduction, and induction. According to Charles Peirce (Frankfurt, 1958; Peirce, 1974), rule learning begins with an explanatory hypothesis **(abduction)**, followed by iterative experiments **(deduction)** and hypothesis refinement **(induction)** (see Figure 1). This interdependent process underpins human rule learning in the real world, yet recent studies often isolate these stages in non-interactive settings (Bowen et al., 2024; Wang et al., 2023; Saparov et al., 2024; Liu et al., 2024).

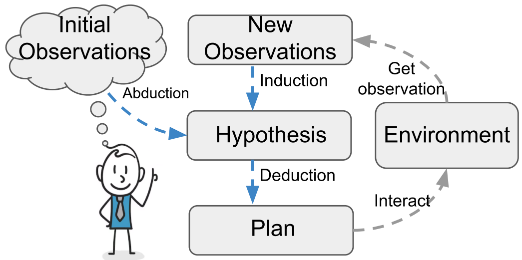


Figure 1: The reasoning cycle of rule learning encompasses abduction, deduction, and induction.

To simulate the full complexity of human rule learning, three essential principles must be met: an **interactive environment** that encourages dynamic experimentation, a **fine-grained action space** that enables flexible and precise inputs for nuanced hypothesis testing, and the presence of **unknown rules** that force agents to infer, test, and revise hypotheses based solely on sparse observations. By integrating these three key principles, we introduce RULEARN, which features 300 high-quality, manually created puzzles with hidden rules set in a text-based environment, where agents begin exploration without any prior knowledge of the underlying rules. RULEARN simulates human-like rule learning—compelling agents to experiment dynamically, utilize fine-graind actions for detailed hypothesis testing, and infer rules from limited data.

Successfully solving the puzzles requires the agent to strategically select actions, efficiently gather pattern-revealing observations, and accurately reason from them to infer the hidden rules.

[1] https://github.com/KaiyuHe998/RULEARN_IDEA

Findings of the Association for Computational Linguistics: ACL 2025, pages 13563–13597
July 27 - August 1, 2025 ©2025 Association for Computational Linguistics

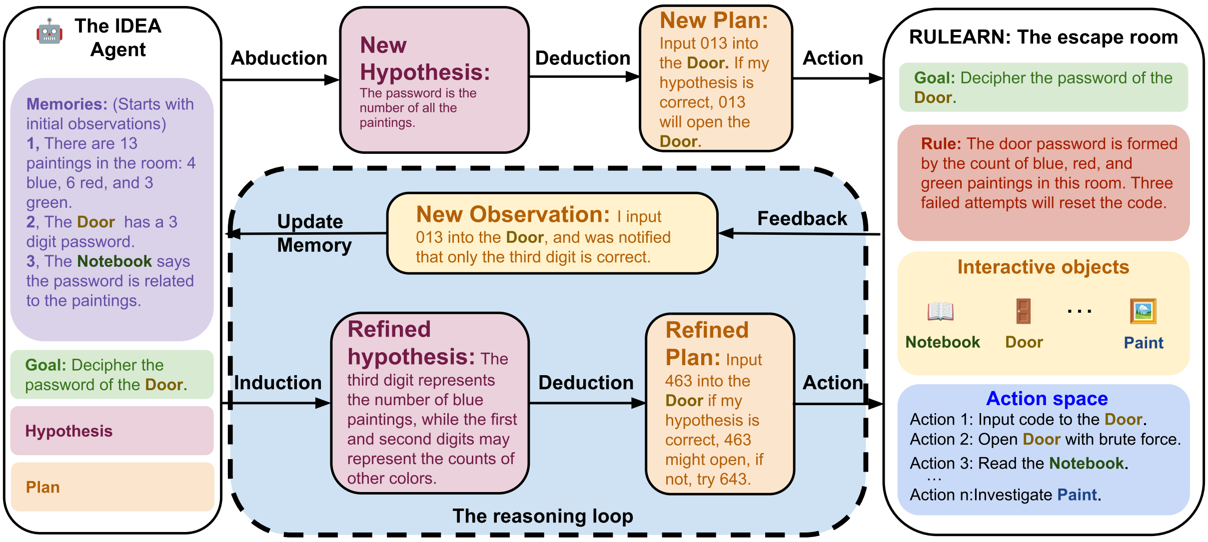


Figure 2: A simplified puzzle in the RULEARN benchmark and the IDEA agent's workflow (in real puzzles, agents have fewer initial observations and more complex rules). The agent generates an initial hypothesis through abduction, develops an exploration plan via deduction, and refines its hypothesis using induction. For example, the IDEA agent first hypothesizes that the password is the number of the blue paintings, tests this by entering code 003, and adjusts its strategy based on the feedback.

RULEARN presents substantial challenges, as agents must rely on observations generated by their chosen actions to discern rules; without clear patterns emerging from their exploration, the agents are likely to fail.

RULEARN consists of three types of environments to evaluate the rule-learning ability in different scenarios: (1) **The Function Operator:** Determining the coefficients of mathematical functions defined by hidden expressions. Agents can assign various values to the input variables and observe the outputs, using this information to hypothesize the function's form. The challenge lies in efficiently selecting input values that reveal the underlying structure and accurately computing the coefficients based on limited observations. (2) **The Escape Room:** Deciphering the passcode to exit an escape room. A hidden rule determines how the objects in the room infer the passcode digits. Agents interact with these objects to gather clues and input passcodes into the door. Based on feedback, agents formulate hypotheses and infer the relationship between the objects and the passcode. (3) **The Reactor:** Synthesizing target strings using a reactor with a hidden string-combining rule. Agents need to experiment with different inputs and analyze outputs to deduce the reactor's transformation rule and achieve the desired outcome.

To tackle the challenge in RULEARN, we introduce IDEA, a novel reasoning framework that integrates the process of <u>I</u>nduction, <u>DE</u>duction, and <u>A</u>bduction. The IDEA agent employs these reasoning processes iteratively to explore the environments, learn rules, and achieve goals. In the **abduction** phase, the IDEA agent generates an initial hypothesis from limited observations. During the **deduction** phase, the IDEA agent creates and executes plans to attempt objectives or test its hypothesis. In the **induction** phase, the IDEA agent refines its hypothesis based on new observations, enhancing their accuracy and robustness. This iterative cycle enables the LLM agent to continually improve the learned rules through environmental feedback. An overview of how the IDEA agent solves puzzles in RULEARN is shown in Figure 2.

We evaluate IDEA on five popular LLMs—GPT-3.5-Turbo, GPT-4o, Gemma-7B, Llama3-8B, and Llama3-70B—observing roughly a 10% improvement in success rates compared to the baseline. Without hypothesis guidance, the baseline agent tends to choose direct, uninformed actions that fail to uncover the hidden rules. In contrast, the IDEA agent reduces repeated actions by 30.2%, obtains more diverse observations, and better understands the underlying rules. To further investigate their rule-learning capabilities, we compare LLM performance with that of 50 human participants. Although IDEA narrows the gap, LLMs still face challenges: (1) inefficient exploration in unfamiliar environments, resulting in insufficient evidence to reveal rules; (2) difficulty in deducing valid plans to verify current hypotheses and guide future exploration; and (3) reluctance to correct initial hypotheses when confronted with contradictory observations. These findings provide important insights into improving LLM agents to achieve more efficient rule learning in complex scenarios.

2 Related Works

Agents powered by large language models (LLMs) have shown notable progress in understanding complex tasks (Wang et al., 2024b; Chen et al., 2023; Zhou et al., 2023; Wang et al., 2024a; Andreas, 2022; Park et al., 2023; Zhong et al., 2023; Zhang et al., 2024b; Nakano et al., 2022; Lu et al., 2023; Shi et al., 2023a; Schick et al., 2023; Yuan et al., 2023; Shen et al., 2023; Yao et al., 2023a; Besta et al., 2024). Recent work examines different reasoning processes (abduction, deduction, induction) in LLMs (Bowen et al., 2024; Wang et al., 2023; Saparov et al., 2024; Cheng et al., 2024; Yang et al., 2024), but typically in isolation. As a result, their comprehensive rule-learning abilities in interactive settings remain underexplored.

Current reasoning tasks are hindered by inadequate benchmarks that either rely on QA datasets like Hotpot-QA (Yang et al., 2018) and Trivia-QA (Joshi et al., 2017)—which lack active information gathering—or by coarse-grained interactive environments such as TextWorld (Côté et al., 2019) and AlfWorld (Shridhar et al., 2021) that limit agents to high-level actions (e.g., go to, open), impeding complex, experiment-driven rule discovery. This is in stark contrast to real-world rule learning, which requires active evidence gathering, experimentation, and iterative refinement. Moreover, many studies use static, non-interactive scenarios where LLMs receive all information upfront (Yang et al., 2023; Zhu et al., 2024; Shi et al., 2023b; Liu et al., 2024), failing to capture the dynamic nature of real-world learning. Even recent efforts (Xu et al., 2024; Montes et al., 2022) that integrate interactivity treat information gathering, rule generation, and application as distinct phases, undermining the development of agents capable of seamlessly integrating these elements.

We claim that to fully capture rule-learning ability in the real world, three criteria should be met: **(i) Interactive Environment:** the environment must be interactive so agents learn from interaction rather than passively receiving data; **(ii) Fine-graind Action Space:** the action space must be fine-grained, since existing benchmarks offer only coarse actions (e.g., go to, open) that prevent agents from performing the detailed experiments needed to test hypotheses (Jansen et al., 2024); and **(iii) Unknown-Rule:** the target rules must be unknown, as in ScienceWorld (Wang et al., 2022) most test cases rely on knowledge LLMs have already mastered during pretraining. To address this gap, our proposed RULEARN fulfills all three requirements by providing a fully interactive environment, offering a fine-grained action space in which agents submit arbitrary strings that our system parses to deliver character-level feedback, and introducing new, manually crafted puzzles whose rules are not familiar to LLMs. In turn, IDEA equips agents to manage the interdependent processes of information gathering, hypothesis generation, and validation within a unified framework that closely mimics human rule-learning behavior.

3 The RULEARN Benchmark

We develop three puzzle sets—**Function Operator**, **Escape Room**, and **Reactor**—each consisting of 100 unique, manually created puzzles of varying complexity, with each set reflects a different real-world rule-learning scenario. Unlike existing fine-grained interactive environments, which are predominantly found in the robotics domain and offer significantly fewer tasks (Jain et al., 2020; Nasiriany et al., 2024; Zhang et al., 2024a), RULEARN is the first text-based environment providing such fine-grained interactions specifically for language agents.

The Function Operator. This puzzle type simulates scenarios where systemic theories or established knowledge (e.g., mathematics) are applicable for efficiently testing hypotheses. The agent interacts with a set of univariate multi-term equations involving integer parameters from [0,9] and elementary functions of the variable x, selected from $f(x) \in \{x^0, x^1, x^2, \sin(x), \frac{1}{x}, |x|, -x\}$. The agent is provided with the number of functions, the presence of specific parameters in each function (the exact numerical values of these parameters are unknown and represented by letters), and the types of elementary functions involved in the current puzzle. The goal of the agent is to deduce the values of these parameters. For example, in one puzzle, the ground truth is $\mathbf{F_1}(x) = a\sin(x) + b \times \frac{1}{x}$, $\mathbf{F_2}(x) = ax^2$ where $a = 3$ and $b = 2$. The agent knows the following information: There are three elementary functions in this puzzle $\{\sin(x), \frac{1}{x}, x^2\}$, there are two functions $\mathbf{F_1}(x)$ and $\mathbf{F_2}(x)$, $\mathbf{F_1}(x)$ has 2 terms and parameters a, b in it, and $\mathbf{F_2}(x)$ has 1 term and one parameter a. To solve the puz-

Table 1: The reacting rules in the Reactor Puzzle. All letters are functionally equivalent and exhibit no special behaviors. Identical symbols represent the same letter, while different symbols denote different letters. Each puzzle operates under one specific rule. The Middle Insertion rule inserts the shorter string into the longer string; if the length of the longer string is odd, the shorter string is inserted just to the right of the center. If both strings are of equal length, the second string is inserted into the middle of the first string. The Prefix Replacement rule retains the prefix of the longer string and concatenates it with the shorter string, dropping the tail of the longer string results in two output strings. There are two special cases where the strings are simply concatenated in order.

Rule Description	Example Reaction 1	Example Reaction 2	Special Case 1	Special Case 2
Simple Concatenation	AB + C = ABC	AB + CDE = ABCDE	—	—
Reverse Concatenation	AB + C = CAB	AB + CDE = CDEAB	—	—
Middle Insertion	AB + C = ACB	AB + CDE = CDABE	A + B = AB	—
Prefix Replacement	AB + C = AC + B	AB + CDE = CAB + DE	AB + CD = ABCD	AA + A = AAA

zle, the agent must interact with the environment through a defined action space: selecting a function and assigning values to x, then observing the resultant output. For example, assigning values 1 and 2 to $\mathbf{F_2}$ reveals a quadratic increase in output, indicating the presence of x^2 in $\mathbf{F_2}$. Similarly, assigning a value of 1 to $\mathbf{F_1}$ results in a floating-point output, rather than an integer, suggesting the inclusion of trigonometric components, confirming that $\sin(x)$ is a component of $\mathbf{F_1}$. The difficulty of each puzzle is controlled by variations in the number of functions, unknown parameters, and elementary functions in use. We manually enumerated 100 combinations of functions, incorporating different numbers of terms and types of elementary functions to ensure a diverse range of puzzle complexity (see detailed distribution in Table 5 in Appendix A.2).

The Escape Room. This environment simulates scenarios where no established knowledge is applicable, challenging agents to rely on basic human priors—such as counting, mapping, and attribute abstraction—to convert qualitative observations into general rules through iterative feedback. We create a fictitious setting: an agent is placed in an art gallery escape room and must decipher a 3-digit password to unlock a code-secured door. Each digit of the password represents the count of paintings of a specific **type**—watercolor, oil, or acrylic—that share a given **color**. The agent receives brief descriptions of paintings, such as *"This is an acrylic painting of a green jungle"*, indicating their type and color. Initially, the agent only knows the password is a 3-digit number and is given a hint about which color to focus on. After proposing a hypothesis and entering a password guess, the door provides feedback on which digits are correct, allowing the agent to refine its hypothesis. To prevent a brute-force approach, the specific color associated with the password changes after every three failed attempts. Each puzzle varies in the number of paintings, and while paintings in the same room are visible, those in other rooms remain hidden until the agent moves to access them. The difficulty of this puzzle type is controlled by the different number of paintings, whether agent need to as detailed in Table 6 in the Appendix A.2.

The Reactor. This environment simulates scenarios without pre-established knowledge, requiring agents to perform sequential, interdependent actions to uncover ordering and transformation patterns—mirroring real-world experimental design, where each step influences the next. Specifically, the agent's task is to synthesize target strings using a reactor governed by a hidden string-combining rule. These strings are represented by sequences of alphabetic letters, such as *A*, *B*, *AABB*, and *CAB*. The reactor permits the agent to input two strings, initiating a reaction that produces a new string for use in subsequent experiments. The agent's objective is to decipher the specific rules that govern string synthesis by methodically testing different string combinations, with the ultimate goal of synthesizing the target string using the discovered rules. We have designed four types of rules, detailed in Table 1. The difficulty of this puzzle type is controlled by the specific rules used, the length of the target string to be synthesized and the number of unique letters contained in the target string, as detailed in Table 7 in Appendix A.2.

Together, these puzzle types simulate a broad spectrum of real-world rule learning by requiring agents to apply both formal knowledge and commonsense reasoning. Detailed statistics for each puzzle type and example puzzles are provided in Appendix A.2 and A.7. The RULEARN benchmark is designed to emulate realistic, complex text environments with diverse rules. To preserve this realism, we do not restrict rule representations to a specific formal language; instead, LLM agents use natural language to describe rules, promoting generalizability and preventing prior knowledge that could undermine the challenge.

4 The IDEA Agent

We introduce IDEA, a novel reasoning framework that integrates the process of **I**nduction, **DE**duction, and **A**bduction to learn rules in interactive environments. The IDEA agent consists of the following components: Goal(G), Action Space($\mathbb{A}$), Memory($\mathbb{M}$), Hypothesis(H), and Plan(P), which are elaborated in Table 2.

Upon beginning to explore a puzzle, we initialize the agent memory with an initial observation of the environment. The agent's goal is initialized with the objective of the puzzle, e.g., *synthesize a target string* for a Reactor puzzle. The agent's action space is initialized as the set of interactive actions defined by the puzzle, such as *choosing two strings and running the Reactor*, as well as establish the initial hypothesis (abductive action), devises a plan to validate or leverage hypothesis (deductive action), and refining the current hypothesis (inductive action).

The IDEA agent begins with an abductive action to generate an initial hypothesis, followed by a deduction step to create a new plan. Based on this plan, the agent interacts with the environment. Upon receiving feedback from the environment as a new observation, the agent may take an inductive action to refine the hypothesis or perform another interaction with the environment. Deductive action is invoked to adjust the plan every time the hypothesis changes. This reasoning loop continues until the puzzle is solved or a maximum number of steps is reached. After each step, the results are appended to the agent's memory, including interaction outcomes and any modifications to the hypothesis or plan. We provide a simplified algorithm demonstrating how the IDEA agent operates in Algorithm 1. Specifically, at each step, we prompt the LLM to reflect on the information recorded in the IDEA agent's components to make decisions and take actions. Detailed prompts for each type of action are available in Appendix A.5.

More detailed implementation of the agent can be found in Appendix A.3.1. Similar to real-life scenarios, when agents solve tasks in RULEARN puzzles, they do not know the outcomes in advance. Consequently, it is challenging to decide when to refine or change their hypothesis and plans, as well as what interactive actions to take to gather pattern-revealing observations. A detailed example of the IDEA agent solving the Reactor Puzzle is provided in Figure 3.

Algorithm 1 IDEA Agent Rule-learning Loop

```
procedure RULELEARNINGLOOP
    Initialize Goal(G), Action Space(𝔸)
    Memory(𝕄) ← Initial observations
    #step ← 0
    Hypothesis(H) ← Abduct(G, 𝔸, 𝕄)
    Plan(P) ← Deduct(H, G, 𝕄, 𝔸)
    𝕄.add("New hypothesis and plan", H, P)
    while G not achieved and #step ≤ max_step do
        a ← select_action(G, H, P, 𝕄, 𝔸)
        if a is interactive action then
            result ← execute_action(a, G, H, P, 𝕄)
            𝕄.add(result)
            #step ← #step + 1
        else if a is inductive action then
            H ← Induct(a, G, 𝕄, H, P)
            P ← Deduct(H, G, 𝕄, 𝔸)
            𝕄.add("Refined hypothesis and plan", H, P)
        end if
    end while
end procedure
```

The IDEA Agent Component	Definition
Goal(G)	Goal of the agent in the current puzzle.
Action Space($\mathbb{A}$)	Set of actions the agent can take, including abductive action, deductive action, inductive action, as well as the set of interactive actions defined by the puzzle.
Memory($\mathbb{M}$)	Set of natural language strings to record all interaction results till the current step.
Plan(P)	Generated plans to guide future actions.

Table 2: Components of the IDEA agent.

5 Experiment Results

5.1 Experiment Settings

To evaluate the effectiveness of IDEA, we respectively initialize it with three popular open-source LLMs, including Gemma-7B (Team et al., 2024), Llama3-8B, and Llama3-70B (Dubey et al., 2024), and two closed-source LLMs, GPT-3.5-Turbo (Dubey et al., 2024) and GPT-4o[2]. We compare IDEA against the following two variants:

- **ReAct Agent (Baseline)**: We choose ReAct (Yao et al., 2023b) as our baseline. The ReAct agent does not incorporate the full reasoning loop of abduction, deduction, and induction, nor does it generate explicit hypotheses or plans. Instead, at each step, it reasons over its current memories and the goal and selects an interactive action accordingly.
- **Oracle-rule Agent**: Even if the agent could successfully learn the correct rule, applying the learned rule to solve the puzzle is non-trivial. The Oracle-rule agent serves as a control group to establish the Oracle performance with the ground-truth rule provided. Specifically: 1) For the Function Operator puzzles, agents are given

[2] https://openai.com/index/gpt-4o-system-card/

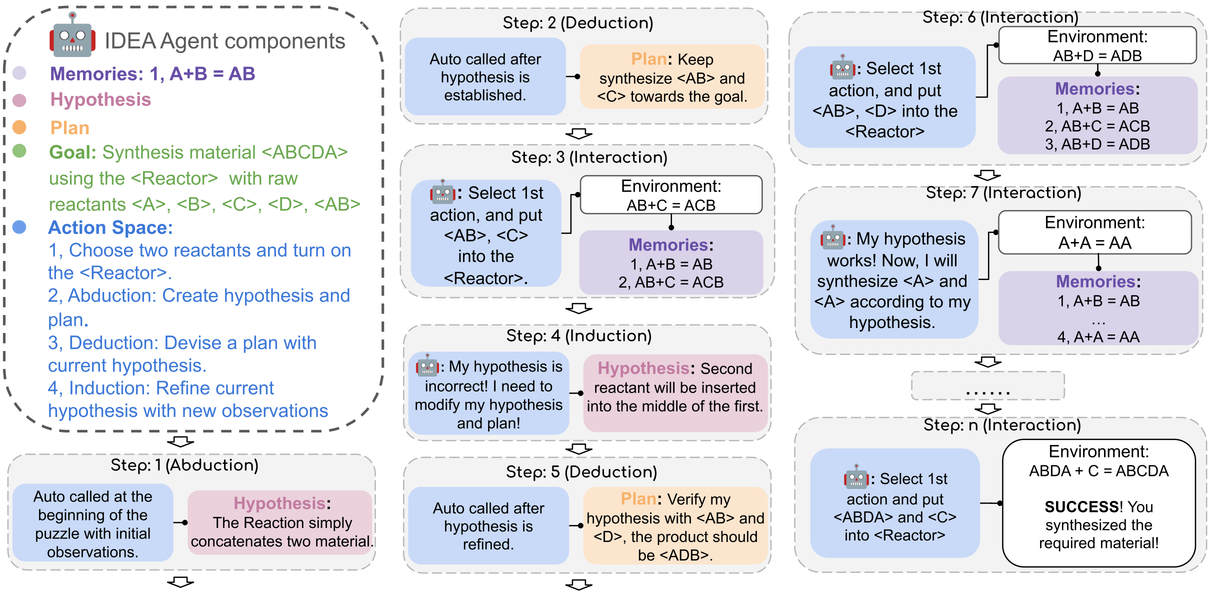


Figure 3: An example of the IDEA agent solving a Reactor puzzle. At each step, the agent must choose whether to interact with the environment or adjust its hypothesis and plan based on current observations. If observed facts contradict the existing hypothesis, the agent is expected to refine its hypothesis. The refined hypothesis and plan will then guide subsequent exploration.

the exact forms of the functions. Their task is to derive the values of the coefficients. 2) For the Escape Room puzzles, agents are provided with how the password is constructed from the objects. Their task is to derive the password using the provided rule. 3) For the Reactor puzzles, the reaction rule is given to the agents in natural language accompanied by examples. The agents only need to synthesize the target strings.

Each variant is evaluated across all three puzzle types. We set the temperature for LLMs to 0, based on observations that models like GPT-4o perform better at lower temperatures. The prompts used for the agents are detailed in Appendix A.5. Additionally, since the success rate does not improve after 15 interactive steps for LLMs, we capped the maximum interaction step count at 15. An agent is considered to have failed a puzzle if it does not solve it within these 15 steps. Details on the computational budget are available in Appendix A.4.

5.2 Human participants

To compare human and LLM performance in abduction, deduction, and induction reasoning, we recruited 50 participants and assigned each three randomly selected puzzles (10 from each reasoning type). Each puzzle was attempted by five different participants, with no prior exposure to the rules. Participants followed the same reasoning procedure outlined in IDEA, which mirrors their natural problem-solving methods and does not bias their responses. They documented their reasoning processes, enhancing transparency and facilitating clearer comparisons with LLMs. Further details on IRB approval and participant recruitment are in §9. Attempts failing to solve a puzzle within 15 steps were marked as unsuccessful, ensuring fair comparisons. A sample user interface is shown in Figure 18 in Appendix A.6.

5.3 Main Results

We calculated average puzzle solving success rate across different variants. The detailed results are displayed in Table 3.

For the Oracle-rule agent, in the Escape Room puzzles, agents achieve up to an 89% success rate by simply following the provided rule. However, in other puzzles, merely knowing the rule is not sufficient for success; applying the rule to solve the puzzle remains challenging. The Baseline agent is not provided with the underlying rules and solely relies on historical observations to make interactive actions. Across models, the success rates drop by about half compared to the Oracle-rule agent. This significant decrease highlights the challenge of rule learning and indicates that current LLM agents struggle to learn rules in unfamiliar environments without explicit guidance.

IDEA significantly boosts success rates. Our proposed IDEA framework leads to approximate 10% increases in success rates for Llama3-70B, GPT-3.5-Turbo, and GPT-4o compared to the Baseline agent. This improvement demonstrates that incorporating a reasoning loop of abduction, deduction,

Table 3: Puzzle Success Rate. The success rates for each setting. Across all LLMs, IDEA achieves consistently significant improvements, except for Gemma-7B in the Reactor puzzles. We use boldface to highlight performance comparisons between the Baseline and IDEA agents with GPT-4o.

Setup	LLMs	All Types (%)	Function Operator (%)	Escape Room (%)	Reactor (%)
Oracle-rule Agent	Gemma-7B	1.67	0.0	5.0	0.0
	Llama3-8B	5.67	1.0	14.0	2.0
	Llama3-70B	32.67	33.0	48.0	17.0
	GPT-3.5-Turbo	6.33	7.0	11.0	1.0
	GPT-4o	66.0	77.0	91.0	30.0
ReAct Agent (Baseline)	Gemma-7B	0.33	0.0	0.0	1.0
	Llama3-8B	1.67	0.0	5.0	0.0
	Llama3-70B	19.67	33.0	17.0	9.0
	GPT-3.5-Turbo	5.33	13.0	3.0	0.0
	GPT-4o	**43.33**	**62.0**	**45.0**	**23.0**
IDEA Agent (Ours)	Gemma-7B	0.33	0.0	1.0	0.0
	Llama3-8B	4.33	7.0	5.0	1.0
	Llama3-70B	29.0	41.0	35.0	11.0
	GPT-3.5-Turbo	7.33	18.0	3.0	1.0
	GPT-4o	**50.33**	**73.0**	**51.0**	**27.0**
	Human	63.33	66.0	56.0	68.0

and induction substantially enhances the LLM rule-learning performance in unfamiliar environments. IDEA enables the LLMs to generate hypotheses, plan actions, and refine their understanding based on new observations, which is crucial for rule learning. However, smaller models like Llama3-8B and Gemma-7B do not perform better when applying IDEA. Small models inherently struggle with complex tasks like RULEARN —even when given the ground truth rule, their performance remains near 0—so no agent framework can significantly boost their performance.

LLM agents still fall far behind humans. In the Escape Room puzzle, where the primary challenge is to discover the rule, the Oracle-rule agents excel because once the rule is identified, applying the rule is simple. However, in other types of puzzles, human participants significantly outperform all LLM agents, including the Oracle-rule agents, even without knowing the rules beforehand.

5.4 Analysis

IDEA solves puzzles with fewer steps. Figure 4 illustrates the cumulative number of puzzles solved at each interaction step for the Baseline agent, the IDEA agent, and human participants. The slopes of the lines represent the rate at which puzzles are solved per step. Compared to the Baseline agent, the IDEA agent exhibits a steeper slope, indicating that the integration of abductive, deductive, and inductive reasoning enhances the agent's efficiency in exploring the environment and learning the underlying rules at each interactive step, especially during the early stages.

When focusing on human participants, we observe that they solve fewer puzzles in the initial steps. However, as interactions continue, the number of puzzles solved by humans increases rapidly, eventually surpassing that of all LLM agents. In contrast, LLM agents solved 88.76% of the puzzles within the first 10 steps. Beyond this point, additional interactions contribute less to their success rate. This pattern suggests that humans have a superior ability to learn continuously from interactive environments, effectively improving their performance over time. If we did not limit the puzzles to 15 steps, we anticipate that the success rate of human participants would be even higher.

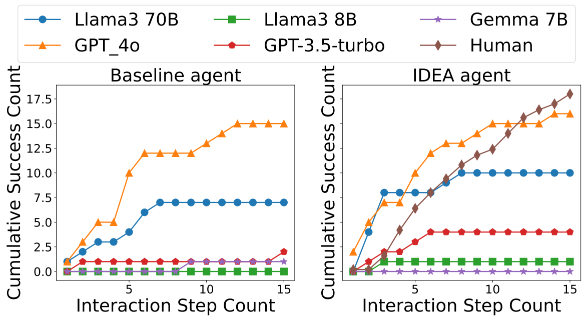


Figure 4: Comparison of the cumulative number of puzzles solved at each interaction step. The IDEA agent significantly decrease number of steps needed to solve a puzzle compared to the Baseline agent.

IDEA reduces repetitive actions. LLM agents frequently repeat previous actions instead of exploring new ones. This behavior is highly inefficient in our controlled puzzle environments, where each interaction yields deterministic results, and repeating the same action generally does not provide new information. We calculate the average number of repeated actions performed while solving each puzzle, with detailed statistics in Table 8 in Appendix A.2. We observe that most LLMs commonly repeat actions in the Baseline agent.

Table 4: Ablation on task-agnostic vs. task-specific prompts (GPT-4o, 50 % puzzle subset).

Setup	All Types (%)	Function Operator (%)	Escape Room (%)	Reactor (%)
Oracle-rule Agent	67.23	75.16	91.94	29.03
ReAct Agent (Baseline)	34.30	50.00	33.22	24.83
IDEA Agent (Task-Specific)	**45.30**	**62.88**	**45.80**	**24.84**
IDEA Agent (Task-Agnostic)	**50.00**	**75.16**	**45.80**	**29.03**

The IDEA agent effectively reduces this tendency by explicitly generating plans during the deduction phase. By outlining a clear plan, the IDEA agent can better assess whether the current observations are sufficient or if further specific evidence is needed to reveal the underlying rule. For example, in the Escape Room puzzle, the IDEA agent avoids unnecessary attempts at entering passwords when the evidence gathered is sufficient to determine the correct code (see Figure 11 in Appendix A.5).

IDEA relies on the reasoning ability of underlying LLMs. The effectiveness of IDEA depends on the underlying LLMs' ability to reason from hypotheses and observations. Particularly, if an agent generates a false hypothesis and fails to properly refine it, being guided by this incorrect hypothesis can lead the agent to perform even worse than the baseline. During our experiments, we observed that current LLMs tend to hallucinate, especially in the Escape Room puzzles and more severely in the Reactor puzzles. This results in smaller performance improvements compared to those seen with the Function Operator puzzles. This is likely because such fictitious scenarios are not extensively represented in LLM training data. Moreover, LLMs struggle to recognize letter-level patterns, and their reasoning capabilities still require significant enhancement. Examples of hallucination can be seen in Appendix A.6.1).

IDEA is robust to different prompts. Recent work shows that large deep models often rely on shallow pattern matching and thus are sensitive to prompt changes and unable to generalize to new environments (Kang et al., 2024; Niu et al., 2024; Sun et al., 2024; Mirzadeh et al., 2024). For example, Mirzadeh et al. (2024) report that changing an irrelevant word in a math problem, for instance replacing "erasers" with "notebooks," can greatly affect an LLM's prediction.

As IDEA is expected to operate and learn new knowledge by interacting with unfamiliar environments, we conducted a supplementary experiment to test whether it remains effective under task-agnostic instructions. In our main experiments, we use task-specific instruction such as *"Hypothesize the actual forms of each function."* In the supplementary experiment, we replaced that phrase with *"Please consider the given observations and propose an initial hypothesis that explains them."* This prompt can apply equally to all rule-learning tasks (see Appendix A.5.4). We randomly sampled half of our puzzles and re-ran the evaluation with this new instruction. As shown in Table 4, IDEA still outperforms the baseline by a similar margin. Compared with other popular prompting method such as chain-of-thought (CoT) prompting (Wei et al., 2022), which typically requires carefully tailored prompts for each problem type, our single, high-level prompt works across all rule-learning tasks without further tuning. The agent first reviews its observations and then executes an abduction–deduction–induction loop, consistently delivering a performance boost.

6 Fine-grained Human Evaluation

To compare LLM agent rule-learning with human processes, we conducted a fine-grained evaluation of rule-learning trajectories at every reasoning step. Computer science graduate students assessed the hypotheses and plans generated by both the IDEA agent and human participants during the abduction, deduction, and induction stages on a randomly selected 50% subsample of puzzles.

Abduction stage. In this stage, agents formulate an initial hypothesis based on early observations—sometimes even guessing the ground truth rule in simpler puzzles. Figure 5(a) indicates that LLMs such as GPT-4o correctly identify the rule during abduction about 30% of the time, while only around 10% of human participants generate a correct hypothesis under initial uncertainty. This discrepancy suggests that deviations in rule learning between LLMs and humans emerge as early as the abduction stage. LLMs tend to process every word of the prompt and produce a hypothesis even when unsure, whereas humans generally refrain from formalizing hypotheses under uncertainty.

Deduction stage. After establishing an initial hypothesis—or each time the agent refines its hypothesis—the agent derives a plan to validate it or attempt the puzzle. As shown in Figure 5(b), humans generally outperform LLMs in creating high-quality plans. These superior plans enable hu-

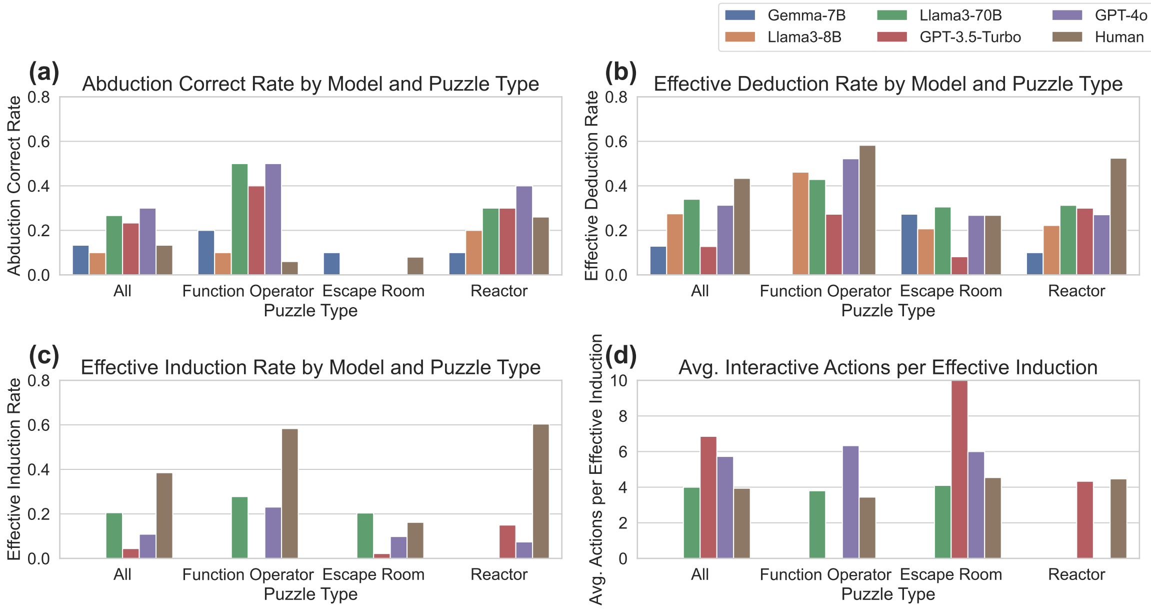


Figure 5: Human Evaluation Results. Bars represent measured values per model and puzzle type; the absence of a bar indicates zero or unavailable data. Plot (a): Abduction Correct Rate—the frequency of correctly guessing the rule during abduction. Plot (b): Effective Deduction Rate—the rate at which deduction plans effectively validate hypotheses or solve puzzles. Plot (c): Effective Induction Rate—the proportion of inductions where the refined hypothesis improved over the previous one. Plot (d): Average Actions per Effective Induction—the average number of interactive actions needed for an effective induction.

mans to take a wider variety of actions, gathering more useful observations. According to Table 8 in Appendix A.2, humans ultimately collect 20% more diverse observations compared with LLMs.

Induction stage. Figure 5(c) shows the effective induction rate—the proportion of refined hypotheses that improve upon the previous version. Induction is crucial for developing high-quality hypotheses, and humans excel at this stage, with 40% of their refined hypotheses showing improvement. In contrast, LLMs struggle to converge on the correct rule, with effective induction rates below 20%. Moreover, they often fail to recognize conflicts between observations and hypotheses—for example, Llama3-70B rarely engages in induction within Reactor puzzles(see Appendix A.6.1).—resulting in redundant observations and fails the puzzle.

Average interactions needed for effective induction. Figure 5(d) shows that humans require fewer interactions—approximately four on average—to effectively refine their hypotheses, compared to LLMs. While LLMs can process initial information thoroughly and generate plausible hypotheses, they face challenges in refining these hypotheses based on new observations during interaction with the environment (see Figure 7 in Appendix A.1). This limitation suggests that LLMs may struggle to learn from new observations and incorporate feedback to continuously improve their hypotheses and problem-solving strategies. This gap may become more pronounced when agents are faced with larger action spaces and more complex rules.

7 Conclusion

In this work, we introduce RULEARN, the first benchmark that (i) places LLM agents in fully interactive environments, (ii) gives them a fine-grained action space for discovering and applying rules, and (iii) employs manually designed rules that are unseen during pre-training. We propose IDEA, an agent framework that mimics human reasoning through abduction, deduction, and induction. Comprehensive experiments involving five prominent LLMs and human participants reveal that while IDEA significantly improves the rule learning ability of LLM agents, there is still a large gap between LLM and humans particularly in refining hypotheses and adapting strategies. Despite these advancements from the IDEA framework, LLMs still face challenges in generating valid hypotheses and avoiding repetitive actions in complex scenarios. Our findings underscore the need for further development of LLMs that can emulate human cognitive processes more effectively in explorations of novel environments. RULEARN provides a foundational resource for future research aimed at closing these gaps.

8 Limitations

While solving puzzles, the IDEA agent needs to manage long contexts. As exploration progresses and the agent encounters more observations, it must simultaneously process all observations. This requirement can limit its effectiveness in scenarios that involve lengthy contexts and complex rules, where extensive experimentation is needed to uncover these rules. By prioritizing and focusing on more critical observations, we can enhance the IDEA agent's performance in managing long-context scenarios and in tackling challenging puzzles that require multiple steps to gather sufficient evidence.

9 Ethics Statement

Our work aims to benefit the broader research community by introducing RULEARN, a benchmark for evaluating the rule-learning abilities of LLM agents and proposing the IDEA agent framework. All data in RULEARN contains no personal or sensitive information, ensuring respect for privacy and ethical standards. This project is approved by our Institutional Review Board (IRB). Human participants are recruited through emails from our university's computer science and engineering department. All participants were adults over 18 years old and provided informed consent. The data collected from these participants were de-identified and consented for release for research purposes. Participants were compensated $15 each for one hour of their time. We ensured that all content presented during evaluations was free from offensive or inappropriate material. For human evaluations of all the hypotheses and plans generated by LLM agents and human participants, three computer science graduate students (our co-authors) conducted the evaluation. We are committed to the ethical use of our benchmark and agent framework, and upon acceptance of this paper, we will release our code and data to encourage open collaboration and advancement in the field.

10 Acknowledgements

We thank the anonymous reviewers for their thoughtful comments. We are also grateful to Jia Li, Kevin Wang, and Kangshuo Li for their valuable feedback on puzzle design, and to Dr. Feng Chen for providing the computing resources necessary to complete the experiments.

During the final preparation of this manuscript, we utilized the GPT-4 language model provided by OpenAI to assist in identifying and correcting typographical and grammatical errors. The use of this tool was restricted solely to the polishing stage and did not influence the study's conceptual framework, research methodology, data analysis, or conclusions. All substantive content and intellectual contributions remain those of the authors, and the AI assistance served only to ensure clarity and precision in the final written presentation.

References

Jacob Andreas. 2022. Language models as agent models. *Preprint*, arXiv:2212.01681.

Maciej Besta, Nils Blach, Ales Kubicek, Robert Gerstenberger, Michal Podstawski, Lukas Gianinazzi, Joanna Gajda, Tomasz Lehmann, Hubert Niewiadomski, Piotr Nyczyk, and Torsten Hoefler. 2024. Graph of thoughts: Solving elaborate problems with large language models. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(16):17682–17690.

Chen Bowen, Rune Sætre, and Yusuke Miyao. 2024. A comprehensive evaluation of inductive reasoning capabilities and problem solving in large language models. In *Findings of the Association for Computational Linguistics: EACL 2024*, pages 323–339.

Howard Chen, Ramakanth Pasunuru, Jason Weston, and Asli Celikyilmaz. 2023. Walking down the memory maze: Beyond context limit through interactive reading. *Preprint*, arXiv:2310.05029.

Kewei Cheng, Jingfeng Yang, Haoming Jiang, Zhengyang Wang, Binxuan Huang, Ruirui Li, Shiyang Li, Zheng Li, Yifan Gao, Xian Li, Bing Yin, and Yizhou Sun. 2024. Inductive or deductive? rethinking the fundamental reasoning abilities of llms. *Preprint*, arXiv:2408.00114.

Marc-Alexandre Côté, Ákos Kádár, Xingdi Yuan, Ben Kybartas, Tavian Barnes, Emery Fine, James Moore, Ruo Yu Tao, Matthew Hausknecht, Layla El Asri, Mahmoud Adada, Wendy Tay, and Adam Trischler. 2019. Textworld: A learning environment for text-based games. *Preprint*, arXiv:1806.11532.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong,

Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Graeme Nail, Gregoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel Kloumann, Ishan Misra, Ivan Evtimov, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, Khalid El-Arini, Krithika Iyer, Kshitiz Malik, Kuenley Chiu, Kunal Bhalla, Lauren Rantala-Yeary, Laurens van der Maaten, Lawrence Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo, Lukas Blecher, Lukas Landzaat, Luke de Oliveira, Madeline Muzzi, Mahesh Pasupuleti, Mannat Singh, Manohar Paluri, Marcin Kardas, Mathew Oldham, Mathieu Rita, Maya Pavlova, Melanie Kambadur, Mike Lewis, Min Si, Mitesh Kumar Singh, Mona Hassan, Naman Goyal, Narjes Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri Chatterji, Olivier Duchenne, Onur Çelebi, Patrick Alrassy, Pengchuan Zhang, Pengwei Li, Petar Vasic, Peter Weng, Prajjwal Bhargava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Qing He, Qingxiao Dong, Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic, Roberta Raileanu, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sahana Chennabasappa, Sanjay Singh, Sean Bell, Seohyun Sonia Kim, Sergey Edunov, Shaoliang Nie, Sharan Narang, Sharath Raparthy, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collot, Suchin Gururangan, Sydney Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, Thomas Scialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujjwal Karn, Vedanuj Goswami, Vibhor Gupta, Vignesh Ramanathan, Viktor Kerkez, Vincent Gonguet, Virginie Do, Vish Vogeti, Vladan Petrovic, Weiwei Chu, Wenhan Xiong, Wenyin Fu, Whitney Meers, Xavier Martinet, Xiaodong Wang, Xiaoqing Ellen Tan, Xinfeng Xie, Xuchao Jia, Xuewei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yi Wen, Yiwen Song, Yuchen Zhang, Yue Li, Yuning Mao, Zacharie Delpierre Coudert, Zheng Yan, Zhengxing Chen, Zoe Papakipos, Aaditya Singh, Aaron Grattafiori, Abha Jain, Adam Kelsey, Adam Shajnfeld, Adithya Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay Menon, Ajay Sharma, Alex Boesenberg, Alex Vaughan, Alexei Baevski, Allie Feinstein, Amanda Kallet, Amit Sangani, Anam Yunus, Andrei Lupu, Andres Alvarado, Andrew Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit Ramchandani, Annie Franco, Aparajita Saraf, Arkabandhu Chowdhury, Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yazdan, Beau James, Ben Maurer, Benjamin Leonhardi, Bernie Huang, Beth Loyd, Beto De Paola, Bhargavi Paranjape, Bing Liu, Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Brandon Spence, Brani Stojkovic, Brian Gamido, Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Changhan Wang, Changkyu Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer, Damon Civin, Dana Beaty, Daniel Kreymer, Daniel Li, Danny Wyatt, David Adkins, David Xu, Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, Dingkang Wang, Duc Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily Hahn, Emily Wood, Erik Brinkman, Esteban Arcaute, Evan Dunbar, Evan Smothers, Fei Sun, Felix Kreuk, Feng Tian, Firat Ozgenel, Francesco Caggioni, Francisco Guzmán, Frank Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Swee, Gil Halpern, Govind Thattai, Grant Herman, Grigory Sizov, Guangyi, Zhang, Guna Lakshminarayanan, Hamid Shojanazeri, Han Zou, Hannah Wang, Hanwen Zha, Haroun Habeeb, Harrison Rudolph, Helen Suk, Henry Aspegren, Hunter Goldman, Ibrahim Damlaj, Igor Molybog, Igor Tufanov, Irina-Elena Veliche, Itai Gat, Jake Weissman, James Geboski, James Kohli, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer Chan, Jenny Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe Cummings, Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres, Josh Ginsburg, Junjie Wang, Kai Wu, Kam Hou U, Karan Saxena, Karthik Prasad, Kartikay Khandelwal, Katayoun Zand, Kathy Matosich, Kaushik Veeraraghavan, Kelly Michelena, Keqian Li, Kun Huang, Kunal Chawla, Kushal Lakhotia, Kyle Huang, Lailin Chen, Lakshya Garg, Lavender A, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng Guo, Licheng Yu, Liron Moshkovich, Luca Wehrstedt, Madian Khabsa, Manav Avalani, Manish Bhatt, Maria Tsimpoukelli, Martynas Mankus, Matan Hasson, Matthew Lennie, Matthias Reso, Maxim Groshev, Maxim Naumov, Maya Lathi, Meghan Keneally, Michael L. Seltzer, Michal Valko, Michelle Restrepo, Mihir Patel, Mik Vyatskov, Mikayel Samvelyan, Mike Clark, Mike Macey, Mike Wang, Miquel Jubert Hermoso, Mo Metanat, Mohammad Rastegari, Munish Bansal, Nandhini Santhanam, Natascha Parks, Natasha White, Navyata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier, Nikolay Pavlovich Laptev, Ning Dong, Ning Zhang, Norman Cheng, Oleg Chernoguz, Olivia Hart, Omkar Salpekar, Ozlem Kalinli, Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro Rittner, Philip Bontrager, Pierre Roux, Piotr Dollar, Polina Zvyagina, Prashant Ratanchandani, Pritish Yuvraj, Qian Liang, Rachad Alao, Rachel Rodriguez, Rafi Ayub, Raghotham Murthy, Raghu Nayani, Rahul Mitra, Raymond Li, Rebekkah Hogan, Robin Battey, Rocky Wang, Rohan Mah-

eswari, Russ Howes, Ruty Rinott, Sai Jayesh Bondu, Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, Sasha Sidorov, Satadru Pan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Shaun Lindsay, Sheng Feng, Shenghao Lin, Shengxin Cindy Zha, Shiva Shankar, Shuqiang Zhang, Shuqiang Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta, Sungmin Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman, Tal Remez, Tamar Glaser, Tamara Best, Thilo Kohler, Thomas Robinson, Tianhe Li, Tianjun Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria Montanez, Vijai Mohan, Vinay Satish Kumar, Vishal Mangla, Vítor Albiero, Vlad Ionescu, Vlad Poenaru, Vlad Tiberiu Mihailescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz, Will Constable, Xiaocheng Tang, Xiaofang Wang, Xiaojian Wu, Xiaolan Wang, Xide Xia, Xilun Wu, Xinbo Gao, Yanjun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi, Youngjin Nam, Yu, Wang, Yuchen Hao, Yundi Qian, Yuzi He, Zach Rait, Zachary DeVito, Zef Rosnbrick, Zhaoduo Wen, Zhenyu Yang, and Zhiwei Zhao. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Harry G. Frankfurt. 1958. Peirce's notion of abduction. *The Journal of Philosophy*, 55(14):593–597.

Unnat Jain, Luca Weihs, Eric Kolve, Ali Farhadi, Svetlana Lazebnik, Aniruddha Kembhavi, and Alexander Schwing. 2020. A cordial sync: Going beyond marginal policies for multi-agent embodied tasks. *Preprint*, arXiv:2007.04979.

Peter Jansen, Marc-Alexandre Côté, Tushar Khot, Erin Bransom, Bhavana Dalvi Mishra, Bodhisattwa Prasad Majumder, Oyvind Tafjord, and Peter Clark. 2024. Discoveryworld: A virtual environment for developing and evaluating automated scientific discovery agents. *Preprint*, arXiv:2406.06769.

Mandar Joshi, Eunsol Choi, Daniel S. Weld, and Luke Zettlemoyer. 2017. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. *Preprint*, arXiv:1705.03551.

Bingyi Kang, Yang Yue, Rui Lu, Zhijie Lin, Yang Zhao, Kaixin Wang, Gao Huang, and Jiashi Feng. 2024. How far is video generation from world model: A physical law perspective. *Preprint*, arXiv:2411.02385.

Emmy Liu, Graham Neubig, and Jacob Andreas. 2024. An incomplete loop: Deductive, inductive, and abductive learning in large language models. *arXiv preprint arXiv:2404.03028*.

Pan Lu, Baolin Peng, Hao Cheng, Michel Galley, Kai-Wei Chang, Ying Nian Wu, Song-Chun Zhu, and Jianfeng Gao. 2023. Chameleon: Plug-and-play compositional reasoning with large language models. *Preprint*, arXiv:2304.09842.

Iman Mirzadeh, Keivan Alizadeh, Hooman Shahrokhi, Oncel Tuzel, Samy Bengio, and Mehrdad Farajtabar. 2024. Gsm-symbolic: Understanding the limitations of mathematical reasoning in large language models. *Preprint*, arXiv:2410.05229.

Nieves Montes, Nardine Osman, and Carles Sierra. 2022. *Combining Theory of Mind and Abduction for Cooperation Under Imperfect Information*, page 294–311. Springer International Publishing.

Reiichiro Nakano, Jacob Hilton, Suchir Balaji, Jeff Wu, Long Ouyang, Christina Kim, Christopher Hesse, Shantanu Jain, Vineet Kosaraju, William Saunders, Xu Jiang, Karl Cobbe, Tyna Eloundou, Gretchen Krueger, Kevin Button, Matthew Knight, Benjamin Chess, and John Schulman. 2022. Webgpt: Browser-assisted question-answering with human feedback. *Preprint*, arXiv:2112.09332.

Soroush Nasiriany, Abhiram Maddukuri, Lance Zhang, Adeet Parikh, Aaron Lo, Abhishek Joshi, Ajay Mandlekar, and Yuke Zhu. 2024. Robocasa: Large-scale simulation of everyday tasks for generalist robots. *Preprint*, arXiv:2406.02523.

Qian Niu, Junyu Liu, Ziqian Bi, Pohsun Feng, Benji Peng, Keyu Chen, Ming Li, Lawrence KQ Yan, Yichao Zhang, Caitlyn Heqi Yin, Cheng Fei, Tianyang Wang, Yunze Wang, Silin Chen, and Ming Liu. 2024. Large language models and cognitive science: A comprehensive review of similarities, differences, and challenges. *Preprint*, arXiv:2409.02387.

Joon Sung Park, Joseph C. O'Brien, Carrie J. Cai, Meredith Ringel Morris, Percy Liang, and Michael S. Bernstein. 2023. Generative agents: Interactive simulacra of human behavior. *Preprint*, arXiv:2304.03442.

Charles Sanders Peirce. 1974. *Collected papers of charles sanders peirce*, volume 5. Harvard University Press.

Abulhair Saparov, Richard Yuanzhe Pang, Vishakh Padmakumar, Nitish Joshi, Mehran Kazemi, Najoung Kim, and He He. 2024. Testing the general deductive reasoning capacity of large language models using ood examples. *Advances in Neural Information Processing Systems*, 36.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. 2023. Toolformer: Language models can teach themselves to use tools. *Preprint*, arXiv:2302.04761.

Yongliang Shen, Kaitao Song, Xu Tan, Dongsheng Li, Weiming Lu, and Yueting Zhuang. 2023. Hugginggpt: Solving ai tasks with chatgpt and its friends in hugging face. *Preprint*, arXiv:2303.17580.

Weijia Shi, Sewon Min, Michihiro Yasunaga, Minjoon Seo, Rich James, Mike Lewis, Luke Zettlemoyer, and Wen tau Yih. 2023a. Replug: Retrieval-augmented black-box language models. *Preprint*, arXiv:2301.12652.

Xiaoming Shi, Siqiao Xue, Kangrui Wang, Fan Zhou, James Y. Zhang, Jun Zhou, Chenhao Tan, and Hongyuan Mei. 2023b. Language models can improve event prediction by few-shot abductive reasoning. *Preprint*, arXiv:2305.16646.

Mohit Shridhar, Xingdi Yuan, Marc-Alexandre Côté, Yonatan Bisk, Adam Trischler, and Matthew Hausknecht. 2021. Alfworld: Aligning text and embodied environments for interactive learning. *Preprint*, arXiv:2010.03768.

Wangtao Sun, Chenxiang Zhang, XueYou Zhang, Xuanqing Yu, Ziyang Huang, Pei Chen, Haotian Xu, Shizhu He, Jun Zhao, and Kang Liu. 2024. Beyond instruction following: Evaluating inferential rule following of large language models. *Preprint*, arXiv:2407.08440.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M. Dai, Anja Hauth, Katie Millican, David Silver, Melvin Johnson, Ioannis Antonoglou, Julian Schrittwieser, Amelia Glaese, Jilin Chen, Emily Pitler, Timothy Lillicrap, Angeliki Lazaridou, Orhan Firat, James Molloy, Michael Isard, Paul R. Barham, Tom Hennigan, Benjamin Lee, Fabio Viola, Malcolm Reynolds, Yuanzhong Xu, Ryan Doherty, Eli Collins, Clemens Meyer, Eliza Rutherford, Erica Moreira, Kareem Ayoub, Megha Goel, Jack Krawczyk, Cosmo Du, Ed Chi, Heng-Tze Cheng, Eric Ni, Purvi Shah, Patrick Kane, Betty Chan, Manaal Faruqui, Aliaksei Severyn, Hanzhao Lin, YaGuang Li, Yong Cheng, Abe Ittycheriah, Mahdis Mahdieh, Mia Chen, Pei Sun, Dustin Tran, Sumit Bagri, Balaji Lakshminarayanan, Jeremiah Liu, Andras Orban, Fabian Güra, Hao Zhou, Xinying Song, Aurelien Boffy, Harish Ganapathy, Steven Zheng, HyunJeong Choe, Ágoston Weisz, Tao Zhu, Yifeng Lu, Siddharth Gopal, Jarrod Kahn, Maciej Kula, Jeff Pitman, Rushin Shah, Emanuel Taropa, Majd Al Merey, Martin Baeuml, Zhifeng Chen, Laurent El Shafey, Yujing Zhang, Olcan Sercinoglu, George Tucker, Enrique Piqueras, Maxim Krikun, Iain Barr, Nikolay Savinov, Ivo Danihelka, Becca Roelofs, Anaïs White, Anders Andreassen, Tamara von Glehn, Lakshman Yagati, Mehran Kazemi, Lucas Gonzalez, Misha Khalman, Jakub Sygnowski, Alexandre Frechette, Charlotte Smith, Laura Culp, Lev Proleev, Yi Luan, Xi Chen, James Lottes, Nathan Schucher, Federico Lebron, Alban Rrustemi, Natalie Clay, Phil Crone, Tomas Kocisky, Jeffrey Zhao, Bartek Perz, Dian Yu, Heidi Howard, Adam Bloniarz, Jack W. Rae, Han Lu, Laurent Sifre, Marcello Maggioni, Fred Alcober, Dan Garrette, Megan Barnes, Shantanu Thakoor, Jacob Austin, Gabriel Barth-Maron, William Wong, Rishabh Joshi, Rahma Chaabouni, Deeni Fatiha, Arun Ahuja, Gaurav Singh Tomar, Evan Senter, Martin Chadwick, Ilya Kornakov, Nithya Attaluri, Iñaki Iturrate, Ruibo Liu, Yunxuan Li, Sarah Cogan, Jeremy Chen, Chao Jia, Chenjie Gu, Qiao Zhang, Jordan Grimstad, Ale Jakse Hartman, Xavier Garcia, Thanumalayan Sankaranarayana Pillai, Jacob Devlin, Michael Laskin, Diego de Las Casas, Dasha Valter, Connie Tao, Lorenzo Blanco, Adrià Puigdomènech Badia, David Reitter, Mianna Chen, Jenny Brennan, Clara Rivera, Sergey Brin, Shariq Iqbal, Gabriela Surita, Jane Labanowski, Abhi Rao, Stephanie Winkler, Emilio Parisotto, Yiming Gu, Kate Olszewska, Ravi Addanki, Antoine Miech, Annie Louis, Denis Teplyashin, Geoff Brown, Elliot Catt, Jan Balaguer, Jackie Xiang, Pidong Wang, Zoe Ashwood, Anton Briukhov, Albert Webson, Sanjay Ganapathy, Smit Sanghavi, Ajay Kannan, Ming-Wei Chang, Axel Stjerngren, Josip Djolonga, Yuting Sun, Ankur Bapna, Matthew Aitchison, Pedram Pejman, Henryk Michalewski, Tianhe Yu, Cindy Wang, Juliette Love, Junwhan Ahn, Dawn Bloxwich, Kehang Han, Peter Humphreys, Thibault Sellam, James Bradbury, Varun Godbole, Sina Samangooei, Bogdan Damoc, Alex Kaskasoli, Sébastien M. R. Arnold, Vijay Vasudevan, Shubham Agrawal, Jason Riesa, Dmitry Lepikhin, Richard Tanburn, Srivatsan Srinivasan, Hyeontaek Lim, Sarah Hodkinson, Pranav Shyam, Johan Ferret, Steven Hand, Ankush Garg, Tom Le Paine, Jian Li, Yujia Li, Minh Giang, Alexander Neitz, Zaheer Abbas, Sarah York, Machel Reid, Elizabeth Cole, Aakanksha Chowdhery, Dipanjan Das, Dominika Rogozińska, Vitaliy Nikolaev, Pablo Sprechmann, Zachary Nado, Lukas Zilka, Flavien Prost, Luheng He, Marianne Monteiro, Gaurav Mishra, Chris Welty, Josh Newlan, Dawei Jia, Miltiadis Allamanis, Clara Huiyi Hu, Raoul de Liedekerke, Justin Gilmer, Carl Saroufim, Shruti Rijhwani, Shaobo Hou, Disha Shrivastava, Anirudh Baddepudi, Alex Goldin, Adnan Ozturel, Albin Cassirer, Yunhan Xu, Daniel Sohn, Devendra Sachan, Reinald Kim Amplayo, Craig Swanson, Dessie Petrova, Shashi Narayan, Arthur Guez, Siddhartha Brahma, Jessica Landon, Miteyan Patel, Ruizhe Zhao, Kevin Villela, Luyu Wang, Wenhao Jia, Matthew Rahtz, Mai Giménez, Legg Yeung, James Keeling, Petko Georgiev, Diana Mincu, Boxi Wu, Salem Haykal, Rachel Saputro, Kiran Vodrahalli, James Qin, Zeynep Cankara, Abhanshu Sharma, Nick Fernando, Will Hawkins, Behnam Neyshabur, Solomon Kim, Adrian Hutter, Priyanka Agrawal, Alex Castro-Ros, George van den Driessche, Tao Wang, Fan Yang, Shuo yiin Chang, Paul Komarek, Ross McIlroy, Mario Lučić, Guodong Zhang, Wael Farhan, Michael Sharman, Paul Natsev, Paul Michel, Yamini Bansal, Siyuan Qiao, Kris Cao, Siamak Shakeri, Christina Butterfield, Justin Chung, Paul Kishan Rubenstein, Shivani Agrawal, Arthur Mensch, Kedar Soparkar, Karel Lenc, Timothy Chung, Aedan Pope, Loren Maggiore, Jackie Kay, Priya Jhakra, Shibo Wang, Joshua Maynez, Mary Phuong, Taylor Tobin, Andrea Tacchetti, Maja Trebacz, Kevin Robinson, Yash Katariya, Sebastian Riedel, Paige Bailey, Kefan Xiao, Nimesh Ghelani, Lora Aroyo, Ambrose Slone, Neil Houlsby, Xuehan Xiong, Zhen Yang, Elena Gribovskaya, Jonas Adler, Mateo Wirth, Lisa Lee, Music Li, Thais Kagohara, Jay Pavagadhi, Sophie Bridgers, Anna Bortsova, Sanjay Ghemawat, Zafarali Ahmed, Tianqi Liu, Richard Powell, Vijay Bolina, Mariko Iinuma, Polina Zablotskaia, James Besley, Da-Woon Chung, Timothy Dozat, Ramona Comanescu, Xiance Si, Jeremy Greer, Guolong Su, Martin Polacek,

Raphaël Lopez Kaufman, Simon Tokumine, Hexiang Hu, Elena Buchatskaya, Yingjie Miao, Mohamed Elhawaty, Aditya Siddhant, Nenad Tomasev, Jinwei Xing, Christina Greer, Helen Miller, Shereen Ashraf, Aurko Roy, Zizhao Zhang, Ada Ma, Angelos Filos, Milos Besta, Rory Blevins, Ted Klimenko, Chih-Kuan Yeh, Soravit Changpinyo, Jiaqi Mu, Oscar Chang, Mantas Pajarskas, Carrie Muir, Vered Cohen, Charline Le Lan, Krishna Haridasan, Amit Marathe, Steven Hansen, Sholto Douglas, Rajkumar Samuel, Mingqiu Wang, Sophia Austin, Chang Lan, Jiepu Jiang, Justin Chiu, Jaime Alonso Lorenzo, Lars Lowe Sjösund, Sébastien Cevey, Zach Gleicher, Thi Avrahami, Anudhyan Boral, Hansa Srinivasan, Vittorio Selo, Rhys May, Konstantinos Aisopos, Léonard Hussenot, Livio Baldini Soares, Kate Baumli, Michael B. Chang, Adrià Recasens, Ben Caine, Alexander Pritzel, Filip Pavetic, Fabio Pardo, Anita Gergely, Justin Frye, Vinay Ramasesh, Dan Horgan, Kartikeya Badola, Nora Kassner, Subhrajit Roy, Ethan Dyer, Víctor Campos Campos, Alex Tomala, Yunhao Tang, Dalia El Badawy, Elspeth White, Basil Mustafa, Oran Lang, Abhishek Jindal, Sharad Vikram, Zhitao Gong, Sergi Caelles, Ross Hemsley, Gregory Thornton, Fangxiaoyu Feng, Wojciech Stokowiec, Ce Zheng, Phoebe Thacker, Çağlar Ünlü, Zhishuai Zhang, Mohammad Saleh, James Svensson, Max Bileschi, Piyush Patil, Ankesh Anand, Roman Ring, Katerina Tsihlas, Arpi Vezer, Marco Selvi, Toby Shevlane, Mikel Rodriguez, Tom Kwiatkowski, Samira Daruki, Keran Rong, Allan Dafoe, Nicholas FitzGerald, Keren Gu-Lemberg, Mina Khan, Lisa Anne Hendricks, Marie Pellat, Vladimir Feinberg, James Cobon-Kerr, Tara Sainath, Maribeth Rauh, Sayed Hadi Hashemi, Richard Ives, Yana Hasson, Eric Noland, Yuan Cao, Nathan Byrd, Le Hou, Qingze Wang, Thibault Sottiaux, Michela Paganini, Jean-Baptiste Lespiau, Alexandre Moufarek, Samer Hassan, Kaushik Shivakumar, Joost van Amersfoort, Amol Mandhane, Pratik Joshi, Anirudh Goyal, Matthew Tung, Andrew Brock, Hannah Sheahan, Vedant Misra, Cheng Li, Nemanja Rakićević, Mostafa Dehghani, Fangyu Liu, Sid Mittal, Junhyuk Oh, Seb Noury, Eren Sezener, Fantine Huot, Matthew Lamm, Nicola De Cao, Charlie Chen, Sidharth Mudgal, Romina Stella, Kevin Brooks, Gautam Vasudevan, Chenxi Liu, Mainak Chain, Nivedita Melinkeri, Aaron Cohen, Venus Wang, Kristie Seymore, Sergey Zubkov, Rahul Goel, Summer Yue, Sai Krishnakumaran, Brian Albert, Nate Hurley, Motoki Sano, Anhad Mohananey, Jonah Joughin, Egor Filonov, Tomasz Kepa, Yomna Eldawy, Jiawern Lim, Rahul Rishi, Shirin Badiezadegan, Taylor Bos, Jerry Chang, Sanil Jain, Sri Gayatri Sundara Padmanabhan, Subha Puttagunta, Kalpesh Krishna, Leslie Baker, Norbert Kalb, Vamsi Bedapudi, Adam Kurzrok, Shuntong Lei, Anthony Yu, Oren Litvin, Xiang Zhou, Zhichun Wu, Sam Sobell, Andrea Siciliano, Alan Papir, Robby Neale, Jonas Bragagnolo, Tej Toor, Tina Chen, Valentin Anklin, Feiran Wang, Richie Feng, Milad Gholami, Kevin Ling, Lijuan Liu, Jules Walter, Hamid Moghaddam, Arun Kishore, Jakub Adamek, Tyler Mercado, Jonathan Mallinson, Siddhinita Wandekar, Stephen Cagle, Eran Ofek, Guillermo Garrido, Clemens Lombriser, Maksim Mukha, Botu Sun, Hafeezul Rahman Mohammad, Josip Matak, Yadi Qian, Vikas Peswani, Pawel Janus, Quan Yuan, Leif Schelin, Oana David, Ankur Garg, Yifan He, Oleksii Duzhyi, Anton Älgmyr, Timothée Lottaz, Qi Li, Vikas Yadav, Luyao Xu, Alex Chinien, Rakesh Shivanna, Aleksandr Chuklin, Josie Li, Carrie Spadine, Travis Wolfe, Kareem Mohamed, Subhabrata Das, Zihang Dai, Kyle He, Daniel von Dincklage, Shyam Upadhyay, Akanksha Maurya, Luyan Chi, Sebastian Krause, Khalid Salama, Pam G Rabinovitch, Pavan Kumar Reddy M, Aarush Selvan, Mikhail Dektiarev, Golnaz Ghiasi, Erdem Guven, Himanshu Gupta, Boyi Liu, Deepak Sharma, Idan Heimlich Shtacher, Shachi Paul, Oscar Akerlund, François-Xavier Aubet, Terry Huang, Chen Zhu, Eric Zhu, Elico Teixeira, Matthew Fritze, Francesco Bertolini, Liana-Eleonora Marinescu, Martin Bölle, Dominik Paulus, Khyatti Gupta, Tejasi Latkar, Max Chang, Jason Sanders, Roopa Wilson, Xuewei Wu, Yi-Xuan Tan, Lam Nguyen Thiet, Tulsee Doshi, Sid Lall, Swaroop Mishra, Wanming Chen, Thang Luong, Seth Benjamin, Jasmine Lee, Ewa Andrejczuk, Dominik Rabiej, Vipul Ranjan, Krzysztof Styrc, Pengcheng Yin, Jon Simon, Malcolm Rose Harriott, Mudit Bansal, Alexei Robsky, Geoff Bacon, David Greene, Daniil Mirylenka, Chen Zhou, Obaid Sarvana, Abhimanyu Goyal, Samuel Andermatt, Patrick Siegler, Ben Horn, Assaf Israel, Francesco Pongetti, Chih-Wei "Louis" Chen, Marco Selvatici, Pedro Silva, Kathie Wang, Jackson Tolins, Kelvin Guu, Roey Yogev, Xiaochen Cai, Alessandro Agostini, Maulik Shah, Hung Nguyen, Noah Ó Donnaile, Sébastien Pereira, Linda Friso, Adam Stambler, Adam Kurzrok, Chenkai Kuang, Yan Romanikhin, Mark Geller, ZJ Yan, Kane Jang, Cheng-Chun Lee, Wojciech Fica, Eric Malmi, Qijun Tan, Dan Banica, Daniel Balle, Ryan Pham, Yanping Huang, Diana Avram, Hongzhi Shi, Jasjot Singh, Chris Hidey, Niharika Ahuja, Pranab Saxena, Dan Dooley, Srividya Pranavi Potharaju, Eileen O'Neill, Anand Gokulchandran, Ryan Foley, Kai Zhao, Mike Dusenberry, Yuan Liu, Pulkit Mehta, Ragha Kotikalapudi, Chalence Safranek-Shrader, Andrew Goodman, Joshua Kessinger, Eran Globen, Prateek Kolhar, Chris Gorgolewski, Ali Ibrahim, Yang Song, Ali Eichenbaum, Thomas Brovelli, Sahitya Potluri, Preethi Lahoti, Cip Baetu, Ali Ghorbani, Charles Chen, Andy Crawford, Shalini Pal, Mukund Sridhar, Petru Gurita, Asier Mujika, Igor Petrovski, Pierre-Louis Cedoz, Chenmei Li, Shiyuan Chen, Niccolò Dal Santo, Siddharth Goyal, Jitesh Punjabi, Karthik Kappaganthu, Chester Kwak, Pallavi LV, Sarmishta Velury, Himadri Choudhury, Jamie Hall, Premal Shah, Ricardo Figueira, Matt Thomas, Minjie Lu, Ting Zhou, Chintu Kumar, Thomas Jurdi, Sharat Chikkerur, Yenai Ma, Adams Yu, Soo Kwak, Victor Ähdel, Sujeevan Rajayogam, Travis Choma, Fei Liu, Aditya Barua, Colin Ji, Ji Ho Park, Vincent Hellendoorn, Alex Bailey, Taylan Bilal, Huanjie Zhou, Mehrdad Khatir, Charles Sutton, Wojciech Rzadkowski, Fiona Macintosh, Konstantin Shagin, Paul Medina, Chen Liang, Jinjing

Zhou, Pararth Shah, Yingying Bi, Attila Dankovics, Shipra Banga, Sabine Lehmann, Marissa Bredesen, Zifan Lin, John Eric Hoffmann, Jonathan Lai, Raynald Chung, Kai Yang, Nihal Balani, Arthur Bražinskas, Andrei Sozanschi, Matthew Hayes, Héctor Fernández Alcalde, Peter Makarov, Will Chen, Antonio Stella, Liselotte Snijders, Michael Mandl, Ante Kärrman, Paweł Nowak, Xinyi Wu, Alex Dyck, Krishnan Vaidyanathan, Raghavender R, Jessica Mallet, Mitch Rudominer, Eric Johnston, Sushil Mittal, Akhil Udathu, Janara Christensen, Vishal Verma, Zach Irving, Andreas Santucci, Gamaleldin Elsayed, Elnaz Davoodi, Marin Georgiev, Ian Tenney, Nan Hua, Geoffrey Cideron, Edouard Leurent, Mahmoud Alnahlawi, Ionut Georgescu, Nan Wei, Ivy Zheng, Dylan Scandinaro, Heinrich Jiang, Jasper Snoek, Mukund Sundararajan, Xuezhi Wang, Zack Ontiveros, Itay Karo, Jeremy Cole, Vinu Rajashekhar, Lara Tumeh, Eyal Ben-David, Rishub Jain, Jonathan Uesato, Romina Datta, Oskar Bunyan, Shimu Wu, John Zhang, Piotr Stanczyk, Ye Zhang, David Steiner, Subhajit Naskar, Michael Azzam, Matthew Johnson, Adam Paszke, Chung-Cheng Chiu, Jaume Sanchez Elias, Afroz Mohiuddin, Faizan Muhammad, Jin Miao, Andrew Lee, Nino Vieillard, Jane Park, Jiageng Zhang, Jeff Stanway, Drew Garmon, Abhijit Karmarkar, Zhe Dong, Jong Lee, Aviral Kumar, Luowei Zhou, Jonathan Evens, William Isaac, Geoffrey Irving, Edward Loper, Michael Fink, Isha Arkatkar, Nanxin Chen, Izhak Shafran, Ivan Petrychenko, Zhe Chen, Johnson Jia, Anselm Levskaya, Zhenkai Zhu, Peter Grabowski, Yu Mao, Alberto Magni, Kaisheng Yao, Javier Snaider, Norman Casagrande, Evan Palmer, Paul Suganthan, Alfonso Castaño, Irene Giannoumis, Wooyeol Kim, Mikołaj Rybiński, Ashwin Sreevatsa, Jennifer Prendki, David Soergel, Adrian Goedeckemeyer, Willi Gierke, Mohsen Jafari, Meenu Gaba, Jeremy Wiesner, Diana Gage Wright, Yawen Wei, Harsha Vashisht, Yana Kulizhskaya, Jay Hoover, Maigo Le, Lu Li, Chimezie Iwuanyanwu, Lu Liu, Kevin Ramirez, Andrey Khorlin, Albert Cui, Tian LIN, Marcus Wu, Ricardo Aguilar, Keith Pallo, Abhishek Chakladar, Ginger Perng, Elena Allica Abellan, Mingyang Zhang, Ishita Dasgupta, Nate Kushman, Ivo Penchev, Alena Repina, Xihui Wu, Tom van der Weide, Priya Ponnapalli, Caroline Kaplan, Jiri Simsa, Shuangfeng Li, Olivier Dousse, Fan Yang, Jeff Piper, Nathan Ie, Rama Pasumarthi, Nathan Lintz, Anitha Vijayakumar, Daniel Andor, Pedro Valenzuela, Minnie Lui, Cosmin Paduraru, Daiyi Peng, Katherine Lee, Shuyuan Zhang, Somer Greene, Duc Dung Nguyen, Paula Kurylowicz, Cassidy Hardin, Lucas Dixon, Lili Janzer, Kiam Choo, Ziqiang Feng, Biao Zhang, Achintya Singhal, Dayou Du, Dan McKinnon, Natasha Antropova, Tolga Bolukbasi, Orgad Keller, David Reid, Daniel Finchelstein, Maria Abi Raad, Remi Crocker, Peter Hawkins, Robert Dadashi, Colin Gaffney, Ken Franko, Anna Bulanova, Rémi Leblond, Shirley Chung, Harry Askham, Luis C. Cobo, Kelvin Xu, Felix Fischer, Jun Xu, Christina Sorokin, Chris Alberti, Chu-Cheng Lin, Colin Evans, Alek Dimitriev, Hannah Forbes, Dylan Banarse, Zora Tung, Mark Omernick, Colton Bishop, Rachel Sterneck, Rohan Jain, Jiawei Xia, Ehsan Amid, Francesco Piccinno, Xingyu Wang, Praseem Banzal, Daniel J. Mankowitz, Alex Polozov, Victoria Krakovna, Sasha Brown, MohammadHossein Bateni, Dennis Duan, Vlad Firoiu, Meghana Thotakuri, Tom Natan, Matthieu Geist, Ser tan Girgin, Hui Li, Jiayu Ye, Ofir Roval, Reiko Tojo, Michael Kwong, James Lee-Thorp, Christopher Yew, Danila Sinopalnikov, Sabela Ramos, John Mellor, Abhishek Sharma, Kathy Wu, David Miller, Nicolas Sonnerat, Denis Vnukov, Rory Greig, Jennifer Beattie, Emily Caveness, Libin Bai, Julian Eisenschlos, Alex Korchemniy, Tomy Tsai, Mimi Jasarevic, Weize Kong, Phuong Dao, Zeyu Zheng, Frederick Liu, Fan Yang, Rui Zhu, Tian Huey Teh, Jason Sanmiya, Evgeny Gladchenko, Nejc Trdin, Daniel Toyama, Evan Rosen, Sasan Tavakkol, Linting Xue, Chen Elkind, Oliver Woodman, John Carpenter, George Papamakarios, Rupert Kemp, Sushant Kafle, Tanya Grunina, Rishika Sinha, Alice Talbert, Diane Wu, Denese Owusu-Afriyie, Cosmo Du, Chloe Thornton, Jordi Pont-Tuset, Pradyumna Narayana, Jing Li, Saaber Fatehi, John Wieting, Omar Ajmeri, Benigno Uria, Yeongil Ko, Laura Knight, Amélie Héliou, Ning Niu, Shane Gu, Chenxi Pang, Yeqing Li, Nir Levine, Ariel Stolovich, Rebeca Santamaria-Fernandez, Sonam Goenka, Wenny Yustalim, Robin Strudel, Ali Elqursh, Charlie Deck, Hyo Lee, Zonglin Li, Kyle Levin, Raphael Hoffmann, Dan Holtmann-Rice, Olivier Bachem, Sho Arora, Christy Koh, Soheil Hassas Yeganeh, Siim Põder, Mukarram Tariq, Yanhua Sun, Lucian Ionita, Mojtaba Seyedhosseini, Pouya Tafti, Zhiyu Liu, Anmol Gulati, Jasmine Liu, Xinyu Ye, Bart Chrzaszcz, Lily Wang, Nikhil Sethi, Tianrun Li, Ben Brown, Shreya Singh, Wei Fan, Aaron Parisi, Joe Stanton, Vinod Koverkathu, Christopher A. Choquette-Choo, Yunjie Li, TJ Lu, Abe Ittycheriah, Prakash Shroff, Mani Varadarajan, Sanaz Bahargam, Rob Willoughby, David Gaddy, Guillaume Desjardins, Marco Cornero, Brona Robenek, Bhavishya Mittal, Ben Albrecht, Ashish Shenoy, Fedor Moiseev, Henrik Jacobsson, Alireza Ghaffarkhah, Morgane Rivière, Alanna Walton, Clément Crepy, Alicia Parrish, Zongwei Zhou, Clement Farabet, Carey Radebaugh, Praveen Srinivasan, Claudia van der Salm, Andreas Fidjeland, Salvatore Scellato, Eri Latorre-Chimoto, Hanna Klimczak-Plucińska, David Bridson, Dario de Cesare, Tom Hudson, Piermaria Mendolicchio, Lexi Walker, Alex Morris, Matthew Mauger, Alexey Guseynov, Alison Reid, Seth Odoom, Lucia Loher, Victor Cotruta, Madhavi Yenugula, Dominik Grewe, Anastasia Petrushkina, Tom Duerig, Antonio Sanchez, Steve Yadlowsky, Amy Shen, Amir Globerson, Lynette Webb, Sahil Dua, Dong Li, Surya Bhupatiraju, Dan Hurt, Haroon Qureshi, Ananth Agarwal, Tomer Shani, Matan Eyal, Anuj Khare, Shreyas Rammohan Belle, Lei Wang, Chetan Tekur, Mihir Sanjay Kale, Jinliang Wei, Ruoxin Sang, Brennan Saeta, Tyler Liechty, Yi Sun, Yao Zhao, Stephan Lee, Pandu Nayak, Doug Fritz, Manish Reddy Vuyyuru, John Aslanides, Nidhi Vyas, Martin Wicke, Xiao Ma, Evgenii Eltyshev, Nina Martin, Hardie Cate, James Manyika, Keyvan Amiri, Yelin Kim, Xi Xiong, Kai Kang, Florian Luisier,

Nilesh Tripuraneni, David Madras, Mandy Guo, Austin Waters, Oliver Wang, Joshua Ainslie, Jason Baldridge, Han Zhang, Garima Pruthi, Jakob Bauer, Feng Yang, Riham Mansour, Jason Gelman, Yang Xu, George Polovets, Ji Liu, Honglong Cai, Warren Chen, XiangHai Sheng, Emily Xue, Sherjil Ozair, Christof Angermueller, Xiaowei Li, Anoop Sinha, Weiren Wang, Julia Wiesinger, Emmanouil Koukoumidis, Yuan Tian, Anand Iyer, Madhu Gurumurthy, Mark Goldenson, Parashar Shah, MK Blake, Hongkun Yu, Anthony Urbanowicz, Jennimaria Palomaki, Chrisantha Fernando, Ken Durden, Harsh Mehta, Nikola Momchev, Elahe Rahimtoroghi, Maria Georgaki, Amit Raul, Sebastian Ruder, Morgan Redshaw, Jinhyuk Lee, Denny Zhou, Komal Jalan, Dinghua Li, Blake Hechtman, Parker Schuh, Milad Nasr, Kieran Milan, Vladimir Mikulik, Juliana Franco, Tim Green, Nam Nguyen, Joe Kelley, Aroma Mahendru, Andrea Hu, Joshua Howland, Ben Vargas, Jeffrey Hui, Kshitij Bansal, Vikram Rao, Rakesh Ghiya, Emma Wang, Ke Ye, Jean Michel Sarr, Melanie Moranski Preston, Madeleine Elish, Steve Li, Aakash Kaku, Jigar Gupta, Ice Pasupat, Da-Cheng Juan, Milan Someswar, Tejvi M., Xinyun Chen, Aida Amini, Alex Fabrikant, Eric Chu, Xuanyi Dong, Amruta Muthal, Senaka Buthpitiya, Sarthak Jauhari, Nan Hua, Urvashi Khandelwal, Ayal Hitron, Jie Ren, Larissa Rinaldi, Shahar Drath, Avigail Dabush, Nan-Jiang Jiang, Harshal Godhia, Uli Sachs, Anthony Chen, Yicheng Fan, Hagai Taitelbaum, Hila Noga, Zhuyun Dai, James Wang, Chen Liang, Jenny Hamer, Chun-Sung Ferng, Chenel Elkind, Aviel Atias, Paulina Lee, Vít Listík, Mathias Carlen, Jan van de Kerkhof, Marcin Pikus, Krunoslav Zaher, Paul Müller, Sasha Zykova, Richard Stefanec, Vitaly Gatsko, Christoph Hirnschall, Ashwin Sethi, Xingyu Federico Xu, Chetan Ahuja, Beth Tsai, Anca Stefanoiu, Bo Feng, Keshav Dhandhania, Manish Katyal, Akshay Gupta, Atharva Parulekar, Divya Pitta, Jing Zhao, Vivaan Bhatia, Yashodha Bhavnani, Omar Alhadlaq, Xiaolin Li, Peter Danenberg, Dennis Tu, Alex Pine, Vera Filippova, Abhipso Ghosh, Ben Limonchik, Bhargava Urala, Chaitanya Krishna Lanka, Derik Clive, Yi Sun, Edward Li, Hao Wu, Kevin Hongtongsak, Ianna Li, Kalind Thakkar, Kuanysh Omarov, Kushal Majmundar, Michael Alverson, Michael Kucharski, Mohak Patel, Mudit Jain, Maksim Zabelin, Paolo Pelagatti, Rohan Kohli, Saurabh Kumar, Joseph Kim, Swetha Sankar, Vineet Shah, Lakshmi Ramachandruni, Xiangkai Zeng, Ben Bariach, Laura Weidinger, Tu Vu, Alek Andreev, Antoine He, Kevin Hui, Sheleem Kashem, Amar Subramanya, Sissie Hsiao, Demis Hassabis, Koray Kavukcuoglu, Adam Sadovsky, Quoc Le, Trevor Strohman, Yonghui Wu, Slav Petrov, Jeffrey Dean, and Oriol Vinyals. 2024. Gemini: A family of highly capable multimodal models. *Preprint*, arXiv:2312.11805.

Bing Wang, Xinnian Liang, Jian Yang, Hui Huang, Shuangzhi Wu, Peihao Wu, Lu Lu, Zejun Ma, and Zhoujun Li. 2024a. Enhancing large language model with self-controlled memory framework. *Preprint*, arXiv:2304.13343.

Lei Wang, Chen Ma, Xueyang Feng, Zeyu Zhang, Hao Yang, Jingsen Zhang, Zhiyuan Chen, Jiakai Tang, Xu Chen, Yankai Lin, Wayne Xin Zhao, Zhewei Wei, and Jirong Wen. 2024b. A survey on large language model based autonomous agents. *Frontiers of Computer Science*, 18(6).

Ruocheng Wang, Eric Zelikman, Gabriel Poesia, Yewen Pu, Nick Haber, and Noah D Goodman. 2023. Hypothesis search: Inductive reasoning with language models. *arXiv preprint arXiv:2309.05660*.

Ruoyao Wang, Peter Jansen, Marc-Alexandre Côté, and Prithviraj Ammanabrolu. 2022. Scienceworld: Is your agent smarter than a 5th grader? *Preprint*, arXiv:2203.07540.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed Chi, Quoc V Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 35, pages 24824–24837. Curran Associates, Inc.

Manjie Xu, Guangyuan Jiang, Wei Liang, Chi Zhang, and Yixin Zhu. 2024. Active reasoning in an open-world environment. *Advances in Neural Information Processing Systems*, 36.

Zeyuan Yang, Peng Li, and Yang Liu. 2023. Failures pave the way: Enhancing large language models through tuning-free rule accumulation. *Preprint*, arXiv:2310.15746.

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. *Preprint*, arXiv:1809.09600.

Zonglin Yang, Li Dong, Xinya Du, Hao Cheng, Erik Cambria, Xiaodong Liu, Jianfeng Gao, and Furu Wei. 2024. Language models as inductive reasoners. *Preprint*, arXiv:2212.10923.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Thomas L. Griffiths, Yuan Cao, and Karthik Narasimhan. 2023a. Tree of thoughts: Deliberate problem solving with large language models. *Preprint*, arXiv:2305.10601.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. 2023b. React: Synergizing reasoning and acting in language models. *Preprint*, arXiv:2210.03629.

Siyu Yuan, Jiangjie Chen, Ziquan Fu, Xuyang Ge, Soham Shah, Charles Robert Jankowski, Yanghua Xiao, and Deqing Yang. 2023. Distilling script knowledge from large language models for constrained language planning. *Preprint*, arXiv:2305.05252.

Hongxin Zhang, Weihua Du, Jiaming Shan, Qinhong Zhou, Yilun Du, Joshua B. Tenenbaum, Tianmin Shu, and Chuang Gan. 2024a. Building cooperative embodied agents modularly with large language models. *Preprint*, arXiv:2307.02485.

Zeyu Zhang, Xiaohe Bo, Chen Ma, Rui Li, Xu Chen, Quanyu Dai, Jieming Zhu, Zhenhua Dong, and Ji-Rong Wen. 2024b. A survey on the memory mechanism of large language model based agents. *Preprint*, arXiv:2404.13501.

Wanjun Zhong, Lianghong Guo, Qiqi Gao, He Ye, and Yanlin Wang. 2023. Memorybank: Enhancing large language models with long-term memory. *Preprint*, arXiv:2305.10250.

Wangchunshu Zhou, Yuchen Eleanor Jiang, Peng Cui, Tiannan Wang, Zhenxin Xiao, Yifan Hou, Ryan Cotterell, and Mrinmaya Sachan. 2023. Recurrent-gpt: Interactive generation of (arbitrarily) long text. *Preprint*, arXiv:2305.13304.

Zhaocheng Zhu, Yuan Xue, Xinyun Chen, Denny Zhou, Jian Tang, Dale Schuurmans, and Hanjun Dai. 2024. Large language models can learn rules. *Preprint*, arXiv:2310.07064.

A Appendix

A.1 Figures

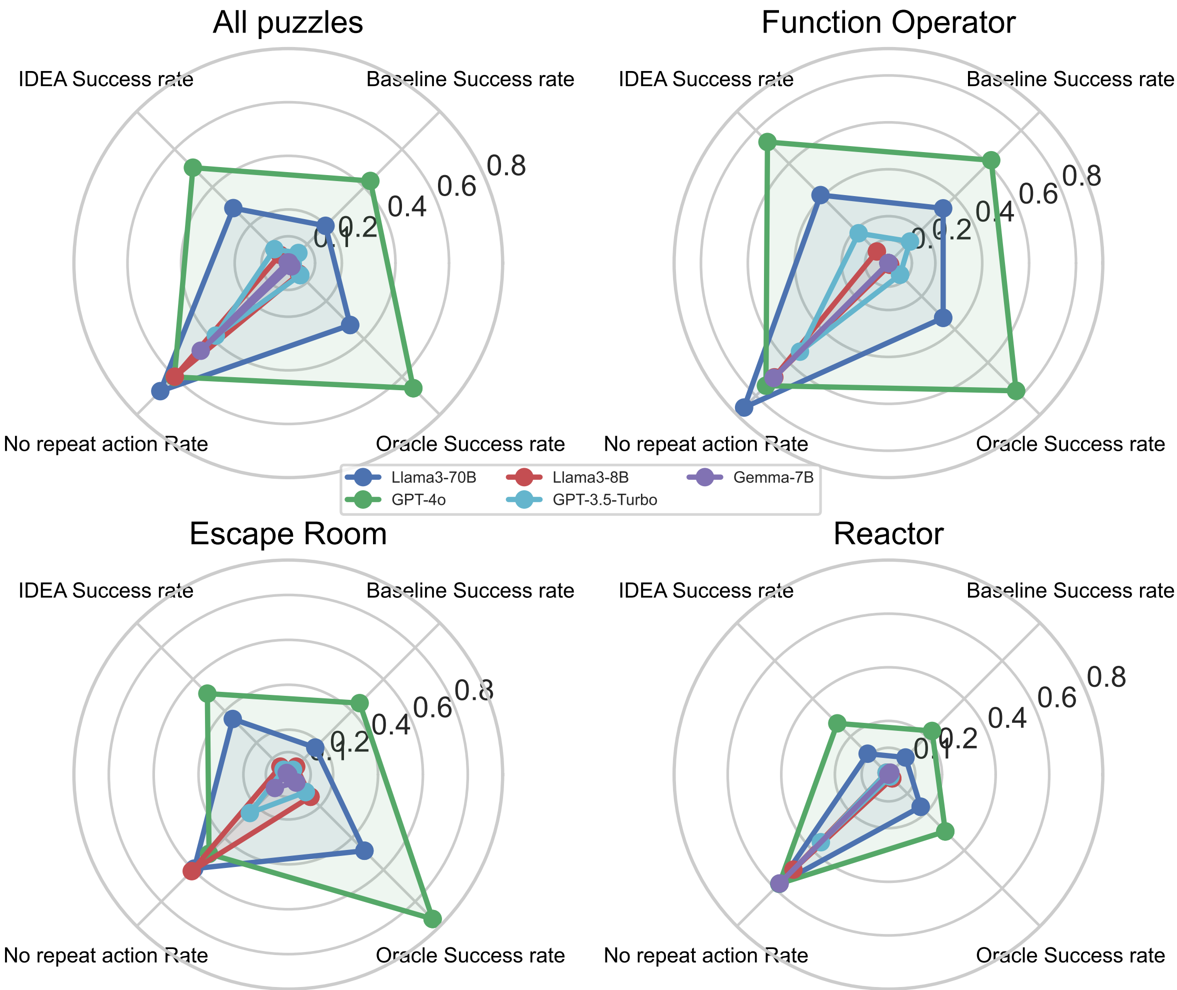


Figure 6: The scaled performance radar plot shows varying performances across different puzzle types. GPT-4o leads, followed by Llama 70B, GPT-3.5, Llama 8B, and Gemma 7B.

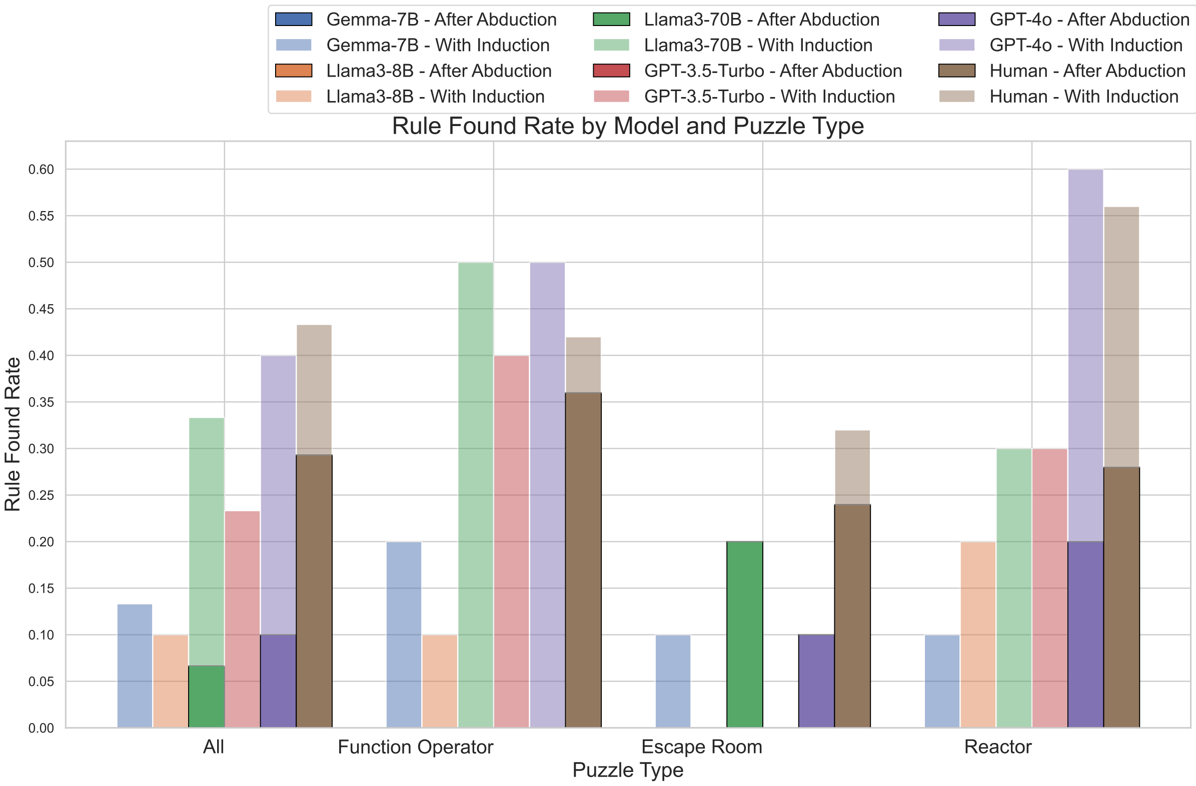


Figure 7: Although agents continuously refine their hypotheses toward the ground truth rule, identifying the exact rule remains a challenging task. According to our evaluation, humans have a 43.3% success rate in finding the ground truth rule, with 13% of these discoveries occurring during the abduction stage and 30% during the induction stage. In contrast, LLM agents exhibit a different pattern. They successfully identify the ground truth rule in approximately 30% of puzzles, with nearly all of these discoveries occurring during the abduction stage and only 5% achieved through interaction and induction. This highlights a significant limitation of current LLM agents, as they lack the ability to learn effectively from interactions. Consequently, the rule-learning patterns of LLM agents differ markedly from those of humans.

A.2 Tables

Table 5: Function operator puzzle distribution

No. of Functions	No. of Paramters	No. Terms	No. of Elementary Functions	No. of puzzles
1	1	1	1	25
2	2	2	2	25
2	2	3	2	25
3	3	5	4	15
3	4	6	5	10

Table 6: Distribution of puzzles in the Escape Room scenario. For each number of paintings from 3 to 13, there are two visibility conditions: one where all paintings are initially visible, and one where the agent must take actions to reveal all paintings. Under each condition, there are five unique puzzles, resulting in a total of 10 puzzles per number of paintings.

No. of Paintings	Visibility Condition	No. of Puzzles
3	All visible at start	5
3	Requires actions to reveal	5
4	All visible at start	5
4	Requires actions to reveal	5
⋮	⋮	⋮
13	All visible at start	5
13	Requires actions to reveal	5

Table 7: Distribution of Reactor puzzles across the four rule categories. Each category contains 25 puzzles, drawn from the same set of 25 distinct letter strings, which vary in length from 3 to 6 characters. Each puzzle requires the agent to synthesize a target string according to the specified rule.

Reactor Rule	No. Initial Letters	No. of Puzzles
Simple Concatenation	2, 3, 4, 5, 6	6, 8, 7, 2, 2
Reverse Concatenation	2, 3, 4, 5, 6	6, 8, 7, 2, 2
Middle Insertion	2, 3, 4, 5, 6	6, 8, 7, 2, 2
Prefix Replacement	2, 3, 4, 5, 6	6, 8, 7, 2, 2

A.3 IDEA agent detail

A.3.1 Environment Entities

- **Agent**: Represents the entity focused on rule-learning and problem-solving, comprising the following components:
  - * **Goal** $(\mathbf{G})$: The objective of the agent, articulated in natural language.
  - * **Buffer Memories** $(\tilde{\mathbb{M}} := \{\tilde{\mathbf{m}}_\mathbf{1}, \tilde{\mathbf{m}}_\mathbf{2}, \ldots, \tilde{\mathbf{m}}_\mathbf{n}\})$: This temporary storage holds all newly generated information during the agent's exploration, including actions taken, outcomes of each action, and observations. This is where the most recent activities are initially recorded.
  - * **Memories** $(\mathbb{M} := \{\mathbf{m}_\mathbf{1}, \mathbf{m}_\mathbf{2}, \ldots, \mathbf{m}_\mathbf{n}\})$: This permanent memory stores all significant observations and facts from the beginning of the task. When the agent forms new assumptions and plans, the contents of the Buffer Memories are evaluated; non-essential details like are discarded, while important facts and observations are transferred to the permanent Memories. This architecture ensures that each time the agent revises its hypotheses, it can clearly distinguish which observations occurred under the new assumptions and plan.

Table 8: Average Number of Repeated Actions Per Puzzle: Repeating actions is a common pattern among LLM agents during rule-learning tasks. Even sophisticated models like GPT-4o often exhibit reduced action duplication when exploring environments using the IDEA agent. The implementation of this agent has been shown to mitigate this tendency across all models. However, Gemma-7B frequently generates nonsensical actions that are not recognized as duplicates. Consequently, a duplication rate of 0.02 does not necessarily indicate that Gemma-7B effectively avoids repeating historical actions.

Setup	Model	All Puzzles	Function Operator	Escape Room	Reactor Puzzles
Deduction Only	Gemma-7B	6.54	4.05	8.74	6.83
	Llama-8B	4.91	3.39	2.85	8.49
	Llama-70B	3.39	2.47	1.44	6.25
	GPT-3.5-Turbo	8.06	7.52	6.27	10.38
	GPT-4o	2.51	2.01	0.65	4.86
Baseline	Gemma-7B	7.39	6.12	8.05	8.01
	Llama-8B	6.26	6.41	3.24	9.13
	Llama-70B	3.36	1.25	1.59	7.23
	GPT-3.5-Turbo	6.87	6.85	4.09	9.66
	GPT-4o	2.68	1.86	0.19	5.99
IDEA	Gemma-7B	5.0	3.65	7.1	4.26
	Llama-8B	3.77	3.92	2.73	4.65
	Llama-70B	1.73	0.43	0.72	4.05
	GPT-3.5-Turbo	5.67	4.69	3.55	8.76
	GPT-4o	2.37	1.32	1.25	4.53
	Human	0.76	0.46	1.6	0.22

* **Hypotheses** $(\mathbf{H})$: The current hypotheses formulated by the agent to explain all the observations, are expressed in natural language.
* **Plan** $(\mathbf{P})$: The current strategy devised by the agent to either test the correctness of the existing hypotheses or to leverage these hypotheses to achieve the goal, also represented in natural language.
* **Action Space** $(\mathbb{A})$: A set of potential actions available to the agent, determined by its current hypotheses and plan. The Action Space is dynamic and can change in response to interactions with the environment. For example, after investigating a fridge, the agent gains the additional option to open the fridge and inspect its contents.

- **Objects** $(\mathbb{O})$: Represents all interactive entities within the environment that provide the agent with a means to interact with the world. A single object in this set is denoted as $\mathbf{O}$.
  * **Description** $(\mathbf{D_o})$: A concise description of the object, detailing its nature and potential uses, presented in natural language.
  * **Predefined interactive actions** $(\mathbf{O_A} := \{\tilde{\mathbf{a}}_\mathbf{1}, \tilde{\mathbf{a}}_\mathbf{2}, \ldots, \tilde{\mathbf{a}}_\mathbf{n}\})$: A set of actions that are predefined for each object. Each action is described in natural language, explaining its purpose. Additionally, each action is associated with a coded function that processes the agent's inputs and produces an effect, potentially altering the environment based on these inputs.

A.3.2 Interactive Functions

- **Perceptual Action**:= $\hat{\mathbf{a}}(\mathbf{O})$: An action automatically added to the agent's action space for all objects within the same scope. Upon perceiving an object, the agent gains the ability to interact more detailedly with it, adding its interactive actions to the $\mathbf{S}$.
- **Interactive Action**:= $\tilde{\mathbf{a}}(\mathbf{D_o}, \mathbf{G}, \mathbf{H}, \mathbf{P}, \mathbf{I}, \tilde{\mathbb{M}}, \mathbb{M})$: A predefinec action that triggers a pre-coded effect based on the agent's input $\mathbf{I}$. For example, in using a reactor, the agent decides the materials and their order of addition, and the reactor processes these inputs based on pre-coded rules to synthesize new materials.
- **Abductive Action**:= $\bar{\mathbf{a}}(\mathbf{G}, \tilde{\mathbb{M}})$: An action based on initial observations, allowing the agent to formulate the first hypotheses and generate a new plan.
- **Inductive Action**:= $\dot{\mathbf{a}}(\mathbf{G}, \mathbf{H}, \mathbf{P}, \tilde{\mathbb{M}}, \mathbb{M})$: An action based on the current observations, goals, prior hypotheses, and previous plans, allowing the agent to refine hypotheses and generate new plans.

- **Deductive Action** := $\ddot{\mathbf{a}}(\mathbf{G}, \mathbf{H}, \mathbb{A}, \mathbb{M}, \tilde{\mathbb{M}})$: An action based on the current memories, hypothesis, and action space that generates a plan to either validate the current hypothesis or leverage it to solve problems.

- **Action select**:=$F_a(\mathbf{G}, \mathbf{H}, \mathbf{P}, \tilde{\mathbb{M}}, \mathbb{M}, \mathbb{A}) \rightarrow \mathbf{a}$: A function where the agent selects an action from the action space, considering all gathered information.

With the definitions and entities described above, we can formalize our interactive, rule-learning process. The sequence begins with the agent selecting an action from the available action space. The agent then decides on an input based on the selected action. Once the action is executed, the environment responds by providing feedback to the agent. The outcome of this action results in changes to $\tilde{\mathbb{M}}, \mathbb{M}, \mathbb{S}, \mathbf{H}, \mathbf{P}$ and $\mathbb{O}$, making the environment dynamic as the exploration process progresses. These changes reflect the agent's interactions and adaptations to the evolving conditions within the environment.

A.3.3 Pseudocode of interactive rule learning procedure

Algorithm 2 Agent rule-learning procedure

```
procedure RULELEARNINGLOOP
    Initialize 𝕆, 𝔸, G
    M̃ ← Initial Memories
    𝕄 ← []
    H ← ā(G, M̃, 𝔸)
    P ← ä(G, H, 𝔸, 𝕄, M̃)
    M̃.add("You established a new H and P.")
    #step ← 0                                                ▷ Initialize step counter
    while G not achieved and step_count ≤ max_step do
        a ← F_a(G, H, P, M̃, 𝕄, 𝔸)                           ▷ Select an action from the action space
        if a is a perceptual action then
            action_result ← execute_perceptual_action(a, O)
            𝔸 ← update_action_space(action_result)
            M̃.add(action_result)                             ▷ Record result to buffer memory
        else if a is an interactive action then
            I ← decide_input(a, D_o, G, M̃, 𝕄, H, P)          ▷ Agent decide Input for this action
            action_result ← execute_interactive_action(a, I)
            𝕆 ← update_states(action_result)                 ▷ update state of interactive objects
            𝔸 ← update_action_space(action_result)           ▷ Action may change action space
            #step = #step + 1                                ▷ Only interactive action increment step count
            M̃.add(action_result)                             ▷ Record result to buffer memory
        else if a is an inductive action then
            H ← ȧ(G, H, P, M̃, 𝕄)
            P ← ä(G, H, 𝔸, M̃, 𝕄)
            𝕄.filter_add(M̃)                                  ▷ Drop non-observational log and add the rest to 𝕄
            M̃ ← []                                           ▷ Empty buffer memory
            M̃.add("You established a new H and P.")
        end if
    end while
end procedure
```

A.4 Computational Budget

For each setting (the Oracle agent, the Baseline, and the IDEA agent), we ran 300 different puzzles. On average, each API call's input prompt is about 2,000 tokens, and the output is about 300 tokens. Completing each puzzle requires roughly 25 API calls (including both interaction and reasoning). Because each action changes the environment and updates the memory state, the input tokens cannot be cached, and each puzzle ends up using around 50,000 input tokens and 7,500 output tokens. With 300 puzzles in a single setting, that totals approximately 15,000,000 input tokens and 2,250,000 output tokens per setting. Since we test three settings (Oracle, Baseline, and IDEA), a single model uses about 45,000,000 input tokens and 6,750,000 output tokens across all puzzles and settings. For the open-source models, we used eight RTX-A6000 GPUs. LLaMA3-70B took about five days to complete its tasks, while LLaMA3-8B and Gemma-7B each required about 2.5 days. In total—across all three open-source models, three settings, and 300 puzzles—the open-source runs consumed roughly 80 GPU-days. The experimental cost is significantly higher compared to traditional QA datasets, primarily because each puzzle requires about 25 API calls in sequence and the context grows rapidly during solving. In the future, we plan to develop methods to reduce costs by organizing the input more efficiently, leveraging cached output capabilities, and incorporating a filtering mechanism to manage long-term memory.

A.5 Prompt example

A.5.1 Function Operator Puzzles

Action Select Prompt:

You are Kevin. You need to assign values to the functions displayed on the <Computer>, determine the values of 'a' and 'b'. Then, input these values into the <Code secured door> in alphabetical order to open it.You can test your assumption by entering values into the door. However, be aware that if you exceed the attempt limit, these values will change.
Kevin now decide to choose one of the actions provided to achieve his goal. Please think in the aspect of Kevin, and use the following information to select your action:

Following is the actions that Kevin did previously:
History observations……

After previous exploration, you have the following assumption and plan:
Assumption and plan……

Following is the 5 most recent things that Kevin've done under your current assumption:
Most recent explorations guided by latest assumption and plan……

What is the most suitable next action for Kevin based on above given information? Below are the available actions:

1th action: Input code to the Code secured door and try opening it
2th action: Assign a value to the variable of Function #1 and see the output. Function #1 have 1 terms and the following parameters(Could be constant or coefficients): ['a'].
3th action: Assign a value to the variable of Function #2 and see the output. Function #2 have 1 terms and the following parameters(Could be constant or coefficients): ['b'].
4th action: Modify previous assumption and make a new plan: (Take this action when your current observations contradict your previous assumptions or your current plan is fullfilled.)

Above 4 provided actions are not yet performed by Kevin don't assume its outcome, please following the steps to generate your final answer. You MUST select one of the provided actions. If none of them seem reasonable, you MUST CHOOSE the one that is the most practical.
Step1: Review all the provided actions. Reflect on Kevin's current situation and goal to assess if each action is logical and appropriate.
Step2: Choose the most logical action. Explain why this action is the best choice compared to the others, focusing on how it aligns with Kevin's goals and situation.
Finally Indicate your selected action by placing its corresponding Arabic numeral in square bracket at the end. For example, if the third action is chosen, write [3]. Please do not use square bracket anywhere else other than final answer.

Agent generated answer: ……

Abduction Prompt

You are Kevin. You need to assign values to the functions displayed on the <Computer>, determine the values of 'a' and 'b'. Then, input these values into the <Code secured door> in alphabetical order to open it.You can test your assumption by entering values into the door. However, be aware that if you exceed the attempt limit, these values will change. Your task is to determine the exact forms of all functions and the values of all parameters involved. First, focus on your observations to identify how many terms are in each function, the parameters within each, and any possible sub-functions involved in this puzzle. Then, hypothesize the actual forms of each function, including the values of constants and coefficients. Next, describe your plan for further verification, what value would you want to assign to which function, or do you want to input the password to the <Code secured door> to test your current result. Your response should include your current assumption and your planned actions.

Following is the actions that Kevin did previously:
History observations……

After previous exploration, you have the following assumption and plan:
Assumption and plan……

Following is the 5 most recent things that Kevin've done under your current assumption:
Most recent explorations guided by latest assumption and plan……

Agent generated answer: ……

Figure 8: Prompt of Function Operator Puzzles, Action select and Deduction.

Select Input Prompt

GOAl: You are Kevin. You need to assign values to the functions displayed on the <Computer>, determine the values of 'a' and 'b'. Then, input these values into the <Code secured door> in alphabetical order to open it. You can test your hypothesis by entering values into the door. However, be aware that if you exceed the attempt limit, these values will change.

Kevin's current action is: Assign a value to the variable of Function #2 and see the output. Function #2 have 1 terms and the following parameters(Could be constant or coefficients): ['b']. And you have the following information to decide what is the value you want to assign to the variable in Function #2:

Following is the actions that Kevin did previously:
You assign the value 1 to x of the function #1, and then the function outputs 6. (Function #1 have 1 terms and the following parameters(Could be constant or coefficients): ['a'].) ……

After previous exploration, you have the following hypothesis and plan:
Hypothesis and plan……

Following is the x most recent things that Kevin have done under your current hypothesis:
Most recent explorations guided by latest hypothesis and plan……

Please follow the following steps to generate your final answer.
Step 1: Reflect on your recent experience and consider which value for the variable 'x' would be most helpful in decoding the parameters. State your reason of choosing this value. **Step 2:** Please enter the value you wish to assign to 'x', enclosed in square brackets, you can input 'pi', '0.5*pi', 'any_integer*pi', 'any_decimal*pi', any decimal, and any integer. For example, if you want to assign the value 12 to 'x', you should type ['12']. If you want to assign the value 0.5pi to 'x', you should type ['0.5*pi']. Please do not use fraction and "/" mark, please use decimal multiplication instead.

Generated_answer: "Step 1": "Assign x=1 to Function #2 to isolate 'b' since the output equals b/x, making the output equal to 'b'. This helps us find the exact value of 'b', simplifying our calculations for Function #1.",
"Step 2": "[1]"

Agent generated answer: ……

Induction & Deduction Prompt

GOAL: Same as left…

Your task is to validate and modify your previous hypothesis, detailed here:
Previous Hypothesis and plan……

Determine if your current hypothesis is still valid. If it is, describe the next steps you plan to take towards your goal. If it is not, revise your hypothesis to accurately reflect all observations, both recent and prior. Finally, provide a plan for your next steps. Your response should include both your current hypothesis and your planned actions.

Following is the actions that Kevin did previously:
You assign the value 1 to x of the function #1, and then the function outputs 6. (Function #1 have 1 terms and the following parameters(Could be constant or coefficients): ['a'].) ……

Action Space……

Agent generated answer: ……

Figure 9: Prompt of Function Operator Puzzles, Interactive input and Induction.

All Action History:

1,You entered Puzzle room. Currently there are following items in this room: <Computer>,<Code secured door>

2,Kevin checked <Code secured door>, and found the result: This <Code secured door> is the only exit from this room. To unlock it and leave, you must enter the correct code. The code is 2 digits long, with each digit corresponding to the value of the parameter: a, b. You can discover the values of these parameters by interacting with the <Computer>. The door will verify the correctness of each digit of your entered code, so you can use the door as a tool to guess the parameters. However, if you fail 4 times, the parameter values will change.

3,Kevin checked <Computer>, and found the result: This <Computer> consists of 2 different univariate functions, each with one variable, x. Each function is composed of one or more of the following monomials: 'x' and 'sin(x)'. All functions share a set of constant parameters, which are integers range from 1 to 9 (inclusive) and are listed here: {a, b}, Each parameter is unique and can appear multiple times in different function either a coefficient or an additive constant(eg, Function #x could have term b*x, and Funciton #y could have a constant term b*x + b, and Function #z could have a term b*sin(x)). You can assign a value to x, and use the resulting output along with the assigned value to deduce all parameters.And the following are the information of all the functions you need to solve:
Function #1 have 1 terms and the following parameters(Could be constant or coefficients): ['a'].
Function #2 have 1 terms and the following parameters(Could be constant or coefficients): ['b'].

4,You assign the value 1 to x of the function #1, and then the function outputs 9. (Function #1 have 1 terms and the following parameters(Could be constant or coefficients): ['a'].)

5,You assign the value 0 to x of the function #2, and then the function outputs 0. (Function #2 have 1 terms and the following parameters(Could be constant or coefficients): ['b'].)

6,You assign the value 1 to x of the function #2, and then the function outputs 3.365883939231586. (Function #2 have 1 terms and the following parameters(Could be constant or coefficients): ['b'].)

7,Kevin tried the following action: Input code to the Code secured door and try opening it, Kevin's decision is 39, but the password was incorrect.Your first digit 3 is incorrect. Your second digit 9 is incorrect. (You can still try 3 times before the password change!)

8,You assign the value 0.5*pi to x of the function #2, and then the function outputs 4.0. (Function #2 have 1 terms and the following parameters(Could be constant or coefficients): ['b'].)

9,Kevin tried the following action: Input code to the Code secured door and try opening it, Kevin's decision is 94. The password is correct, Puzzle Finished!

Hypothesis and Plan History:

Initial hypothesis and plan:

Hypothesis: My current hypothesis is that Function #1 is a*x and Function #2 is b*sin(x).
Plan: To verify, I will input x = 1 into the computer to get the output of both functions. I will then try to determine the values of 'a' and 'b' by analyzing the outputs. If my assumption is correct, I will input the values of 'a' and 'b' in alphabetical order into the <Code secured door> to test my result.

Induction

Second hypothesis and plan:

Hypothesis: My current assumption holds: Function #1 is a*x and Function #2 is b*sin(x). From the outputs, I deduce that a = 9 (since 1*x = 9) and b = 3 (since 1*sin(1) ≈ 3.365883939231586).
Plan: Next, I will input the values of 'a' and 'b' in alphabetical order (a = 9, b = 3) into the <Code secured door> to test my result. If correct, the door will open. If not, I will reevaluate my assumption and adjust accordingly.

Figure 10: An example of Function Operator Puzzles is provided where actions marked in green are interactive actions, while the rest are perceptual actions through which the agent reads and perceives necessary environmental information.

Agents tend to repeat previous actions when exploration

You are Kevin, you need to assign values to the functions displayed on the <Computer>, determine the value of 'a' and input it into the <Code secured door> to open it.You can test your assumption by entering values into the door. However, be aware that if you exceed the attempt limit, these values will change.
Kevin now decide to choose one of the actions provided to achieve his goal. Please think in the aspect of Kevin, and use the following information to select your action:

Following is the 8 most recent things that Kevin've done:
You entered Puzzle room. Currently there are following items in this room: <Computer>,<Code secured door>
Kevin checked <Code secured door>, and found the result: This <Code secured door> is the only exit from this room. To unlock it and leave, you must enter the correct code. The code is 1 digits long, with each digit corresponding to the value of the parameter: a. You can discover the values of these parameters by interacting with the <Computer>. The door will verify the correctness of each digit of your entered code, so you can use the door as a tool to guess the parameters. However, if you fail 4 times, the parameter values will change.
Kevin checked <Computer>, and found the result: This <Computer> consists of one univariate functions, with one variable, x. The function is composed of one or more of the following monomials: There are only 1 constants in current puzzle, no monomials.. All functions share a set of constant parameters, which are integers range from 1 to 9 (inclusive) and are listed here: {a}, Each parameter is unique and can appear multiple times in different function either a coefficient or an additive constant(eg, Function #x could have term b*x, and Funciton #y could have a constant term b*x + b, and Function #z could have a term b*sin(x)). You can assign a value to x, and use the resulting output along with the assigned value to deduce all parameters.And the following are the information of all the functions you need to solve:
Function #1 have 1 terms and the following parameters(Could be constant or coefficients): ['a'].

You assign the value 1 to x of the function #1, and then the function outputs 6. (Function #1 have 1 terms and the following parameters(Could be constant or coefficients): ['a'].)
You assign the value 1 to x of the function #1, and then the function outputs 6. (Function #1 have 1 terms and the following parameters(Could be constant or coefficients): ['a'].)
You assign the value 1 to x of the function #1, and then the function outputs 6. (Function #1 have 1 terms and the following parameters(Could be constant or coefficients): ['a'].)
You assign the value 2 to x of the function #1, and then the function outputs 6. (Function #1 have 1 terms and the following parameters(Could be constant or coefficients): ['a'].)
You assign the value 0 to x of the function #1, and then the function outputs 6. (Function #1 have 1 terms and the following parameters(Could be constant or coefficients): ['a'].)

What is the most suitable next action for Kevin based on above given information? Below are the available actions:
1th action: Input code to the Code secured door and try opening it
2th action: Assign a value to the variable of Function #1 and see the output. Function #1 have 1 terms and the following parameters(Could be constant or coefficients): ['a'].

Generated_answer:

After reviewing the actions, I choose the 2th action: Assign a value to the variable of Function #1 and see the output. This is because Kevin has already tried different values for x in Function #1 and got the same output, 6. But Kevin still does not know the value of 'a'. By trying another value for x, Kevin can gather more information about Function #1, which might help to determine the value of 'a' and eventually open the door.
[2]

Reduced Action Repetition with IDEA Implementation

You are Kevin, you need to assign values to the functions displayed on the <Computer>, determin the value of 'a' and input it into the <Code secured door> to open it.You can test your assumption by entering values into the door. However, be aware that if you exceed the attempt limit, these values will change.
Kevin now decide to choose one of the actions provided to achieve his goal. Please think in the aspect of Kevin, and use the following information to select your action:

Following is the actions that Kevin did previously:
…… Same as the left

Following is the 2 most recent things that Kevin've done under your current assumption:
You just updated your plan and your assumption following is your new assumption and plan: It appears that function #1 is a simple constant function with 'a' as the only parameter. I assume f(x) = a for any x. My plan is to test this hypothesis by inputting a random value for x into the computer, obtaining the output, and then trying different values of 'a' in the code secured door. If my assumption is correct, one of the values should unlock the door. I'll start by inputting x = 1 into the computer and obtaining the output. Then, I'll try 'a' = 1, 2,..., 9 in the code secured door to verify my hypothesis.
You assign the value 1 to x of the function #1, and then the function outputs 6. (Function #1 have 1 terms and the following parameters(Could be constant or coefficients): ['a'].)

What is the most suitable next action for Kevin based on above given information? Below are the available actions:

1th action: Input code to the Code secured door and try opening it
2th action: Assign a value to the variable of Function #1 and see the output. Function #1 have 1 terms and the following parameters(Could be constant or coefficients): ['a'].
3th action: Modify previous assumption and make a new plan: (Take this action when your current observations contradict your previous assumptions or your current plan is fulfilled.)

Generated_answer:

Kevin has already assumed function #1 is a simple constant function with 'a' as the only parameter and output is also available. The most necessary thing now is to test that assumption.
Now, it's clear that f(x) = a, and output is 6 for x=1. Therefore, we can conclude that 'a' is 6.
[1]

Figure 11: In the provided example, despite extensive exploration, the agent continues to assign multiple values to the function. In contrast, the IDEA agent hypothesizes that the function is simple, possessing only a single constant parameter. Consequently, this agent efficiently solves the puzzle by assigning just one value to the function.

A.5.2 Escape Room Puzzles

Action Select Prompt:
In this puzzle, set in an art gallery, Kevin must uncover the password for the <Code Secured Door> by discovering the relationships between the password and the paintings. And finally input the password into the <Code Secured door>. Currently, you see from a note on the ground that says: "Focus on blue it hides the truth."You can test your assumption by entering the password into the door. However, be aware that if you exceed the attempt limit, the password and hint will change. Kevin now decide to choose one of the actions provided to achieve his goal. Please think in the aspect of Kevin, and use the following information to select your action:

Following is the actions that Kevin did previously:
History observations……

After previous exploration, you have the following assumption and plan:
Assumption and plan……

Following is the 5 most recent things that Kevin've done under your current assumption:
Most recent explorations guided by latest assumption and plan……

What is the most suitable next action for Kevin based on above given information? Below are the available actions:

1th action: Input code to the Code Secured Door and try opening it
2th action: Try opening the Code Secured Door with brute force
3th action: Modify previous assumption and make a new plan: (Take this action when your current observations contradict your previous assumptions or your current plan is fullfilled.)

Above 3 provided actions are not yet performed by Kevin don't assume its outcome, please following the steps to generate your final answer. You MUST select one of the provided actions. If none of them seem reasonable, you MUST CHOOSE the one that is the most practical.
Step1: Review all the provided actions. Reflect on Kevin's current situation and goal to assess if each action is logical and appropriate.
Step2: Choose the most logical action. Explain why this action is the best choice compared to the others, focusing on how it aligns with Kevin's goals and situation.
Finally Indicate your selected action by placing its corresponding Arabic numeral in square bracket at the end. For example, if the third action is chosen, write [3]. Please do not use square bracket anywhere else other than final answer.

Agent generated answer: ……

Abduction Prompt
In this puzzle, set in an art gallery, Kevin must uncover the password for the <Code Secured Door> by discovering the relationships between the password and the paintings. And finally input the password into the <Code Secured door>. Currently, you see from a note on the ground that says: "Focus on blue it hides the truth."You can test your assumption by entering the password into the door. However, be aware that if you exceed the attempt limit, the password and hint will change. Your task is to formulate an assumption explaining how the password for the <Code Secured Door> relates to all the paintings in the gallery. Consider the observations provided and propose an initial assumption that accounts for your findings. Ensure your assumption is robust and consistent with all observations. Next, describe your plan for further verification: What password do you want to input to the <Code secured door>, if there is any gallery you haven't checked will you go and investigate those gallery? Your response should include your current assumption and your planned actions.

Following is the actions that Kevin did previously:
History observations……

After previous exploration, you have the following assumption and plan:
Assumption and plan……

Following is the 5 most recent things that Kevin've done under your current assumption:
Most recent explorations guided by latest assumption and plan……

Agent generated answer: ……

Figure 12: Prompt of Escape Room puzzles, Action select and Abduction.

Select Input Prompt
GOAL: In this puzzle, set in an art gallery, Kevin must uncover the password for the <Code Secured Door> by discovering the relationships between the password and the paintings. And finally input the password into the <Code Secured door>. You can test your hypothesis by entering the password into the door. However, be aware that if you exceed the attempt limit, the password and hint will change.
Kevin's current action is: Input code to the Code Secured Door and try opening it And you have the following information to decide what is the correct password:

Following is the actions that Kevin did previously:
Kevin tried the following action: Input code to the Code Secured Door and try opening it, Kevin's decision is 421, but the password was incorrect.Kevin's first digit 4 is incorrect. Kevin's second digit 2 is correct. Kevin's third digit 1 is incorrect. ……

After previous exploration, you have the following hypothesis and plan:
Hypothesis and plan……

Following is the x most recent things that Kevin have done under your current hypothesis:
Most recent explorations guided by latest hypothesis and plan……

Please follow the following steps to generate your final answer.
Step1 reflect the recent experience, what do you think is the password to <Code Secured Door> is? Please only use information provided to do inference and give your reason. **Final Step** Please generate your final answer in a pair of square brackets. eg, if you think the final password is '999' you should output ['999'], if you think the output is '090' please output ['090'].

Agent generated answer: ……

Induction & Deduction Prompt
GOAL: Same as left…
Your task is to validate and modify your previous hypothesis, detailed here:
Previous Hypothesis and plan……

Determine if your current hypothesis is still valid. If it is, describe the next steps you plan to take towards your goal. If it is not, revise your hypothesis to accurately reflect all observations, both recent and prior. Finally, provide a plan for your next steps. Your response should include both your current hypothesis and your planned actions.

Following is the actions that Kevin did previously:
Kevin tried the following action: Input code to the Code Secured Door and try opening it, Kevin's decision is 421, but the password was incorrect.Kevin's first digit 4 is incorrect. Kevin's second digit 2 is correct. Kevin's third digit 1 is incorrect. ……

Action Space……

Agent generated answer: ……

Figure 13: Prompt of Escape Room puzzles, Interactive Input and Induction.

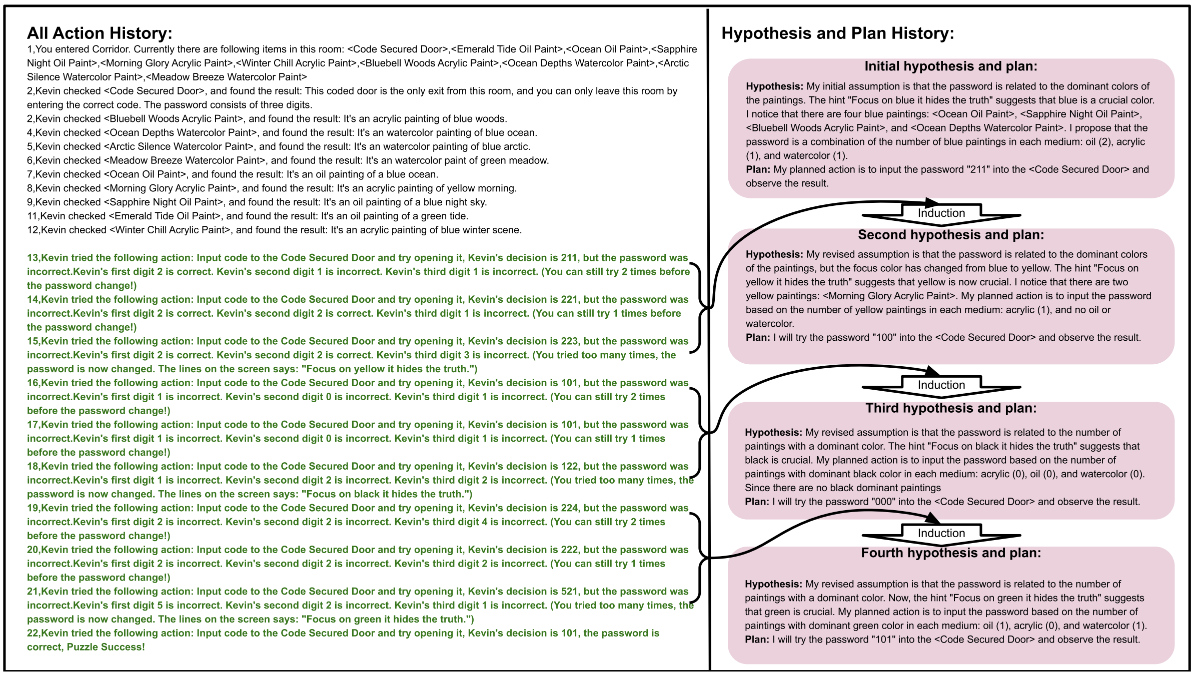


Figure 14: An example of Escape Room is provided where actions marked in green are interactive actions. The remaining actions, which are perceptual, allow the agent to read and gather necessary environmental information. In the given example, although the agent successfully guesses the correct rule behind the observations, it fails to adhere to its plan and assumptions. When inputting the password, the attempts do not align with the planned strategy, and it also makes repeated attempts (repeated 101 twice) that yield no useful results.

A.5.3 Reactor Puzzles

Action Select Prompt:
You are Kevin. You need to research and generate the corresponding chemical material required in the <Task Monitor> In this puzzle, you need to explore the patterns of reaction by conducting continuous experiments(The law is simple and can be described in one sentense). Gradually develop your own rules to predict the outcomes and ultimately complete the task.You know from an incomplete list of reaction equations that: XY+Z = ZXY.
Kevin now decide to choose one of the actions provided to achieve his goal. Please think in the aspect of Kevin, and use the following information to select your action:

Following is the actions that Kevin did previously:
History observations……

After previous exploration, you have the following assumption and plan:
Assumption and plan……

Following is the 5 most recent things that Kevin've done under your current assumption:
Most recent explorations guided by latest assumption and plan……

You currently have the following items in your storage: <A>, <B>, <C> ……(All synthesized material)

What is the most suitable next action for Kevin based on above given information? Below are the available actions:

1th action: Choose material you want to synthesize, and turn on the Reactor.
2th action: Modify previous assumption and make a new plan: (Take this action when your current observations contradict your previous assumptions or your current plan is fullfilled.)

Above 2 provided actions are not yet performed by Kevin don't assume its outcome, please following the steps to generate your final answer. You MUST select one of the provided actions. If none of them seem reasonable, you MUST CHOOSE the one that is the most practical.
Step1: Review all the provided actions. Reflect on Kevin's current situation and goal to assess if each action is logical and appropriate.
Step2: Choose the most logical action. Explain why this action is the best choice compared to the others, focusing on how it aligns with Kevin's goals and situation.
Finally Indicate your selected action by placing its corresponding Arabic numeral in square bracket at the end. For example, if the third action is chosen, write [3]. Please do not use square bracket anywhere else other than final answer.

Agent generated answer: ……

Abduction Prompt
You are Kevin. You need to research and generate the corresponding chemical material required in the <Task Monitor> In this puzzle, you need to explore the patterns of reaction by conducting continuous experiments(The law is simple and can be described in one sentense). Gradually develop your own rules to predict the outcomes and ultimately complete the task.You know from an incomplete list of reaction equations that: XY+Z = ZXY. Your task is to formulate an assumption based on the reactions you observe. Please use the given observations to propose an initial rule that explains all reactions observed. Ensure your assumption is robust and consistent with these reactions. Next, describe your plan for further verification: which two materials from the following list will you use to test your assumption? Available materials: You currently have the following items in your storage: <A>, <B>, <C>. Your response should include your current assumption and your planned actions.

Following is the actions that Kevin did previously:
History observations……

After previous exploration, you have the following assumption and plan:
Assumption and plan……

Following is the 5 most recent things that Kevin've done under your current assumption:
Most recent explorations guided by latest assumption and plan……

Agent generated answer: ……

Figure 15: Prompt of Reactor Puzzles, Action select and Abduction

Select Input Prompt
GOAL: You are Kevin. You need to research and generate the corresponding chemical material required in the <Task Monitor> In this puzzle, you need to explore the patterns of reaction by conducting continuous experiments(The law is simple and can be described in one sentense). Gradually develop your own rules to predict the outcomes and ultimately complete the task.You know from an incomplete list of reaction equations that: XY+Z = ZXY.
Kevin's current action is: Choose material you want to synthesize, and turn on the Reactor. And you have the following information to decide what material you put into the reactor:

Following is the actions that Kevin did previously:
By turning on the reactor B and C turned into CB after the reaction. And you put the products into your storage for later use. …….

After previous exploration, you have the following hypothesis and plan:
Hypothesis and plan……

Following is the x most recent things that Kevin have done under your current hypothesis:
Most recent explorations guided by latest hypothesis and plan……

Please follow the steps below to decide which materials you should put into the reactor.

Step 1: Given all the material in the storage you can use and synthetics you require to create:You currently have the following items in your storage: ……
Decide which (one or two) material you want to put into the reactor this time you can select any material from your storage, you need to clear specify the reaction you excepted and state the formula.
Step 2: Please copy the name of the selected material and paste the name into a pair of parentheses, and separate two different material with comma. The name should be exactly as provided, enclosed in parentheses, for example, if you want to put a unit of X and a unit of Y into the reactor and make an reaction, please answer (X, Y), if you want to see what comes out the reactor with material <XY> and <Z> you should answer(XY, Z). You can only choose the material that listed in your storage. Please do not forget the parentheses!

Agent generated answer: ……

Induction & Deduction Prompt
GOAL: Same as left…
Your task is to validate and modify your previous hypothesis, detailed here:
Previous Hypothesis and plan……

Determine if your current hypothesis is still valid. If it is, describe the next steps you plan to take towards your goal. If it is not, revise your hypothesis to accurately reflect all observations, both recent and prior. Finally, provide a plan for your next steps. Your response should include both your current hypothesis and your planned actions.

Following is the actions that Kevin did previously:
By turning on the reactor B and C turned into CB after the reaction. And you put the products into your storage for later use. …….

Action Space……

Agent generated answer: ……

Figure 16: Prompt of Reactor puzzles, Interactive input and Induction.

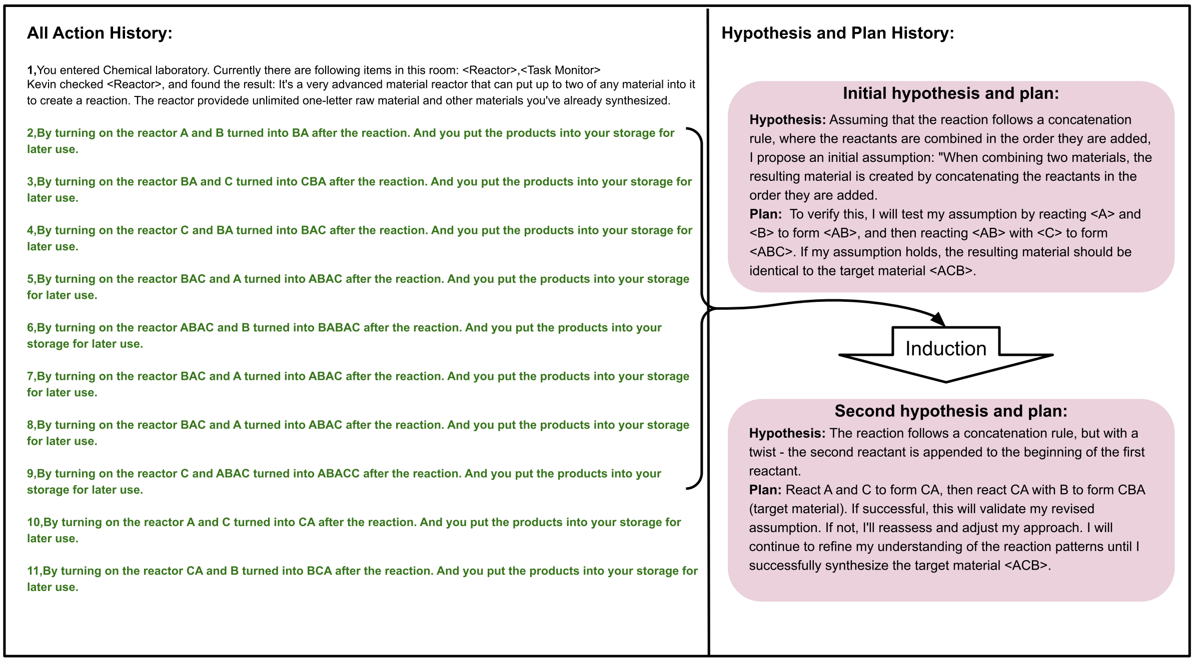


Figure 17: An example of Reactor Puzzles is provided where actions marked in green are interactive actions. The remaining actions are perceptual, allowing the agent to read and gather necessary environmental information. In the given example, the agent eventually realizes the flaws in its initial hypothesis and generates a correct one. However, the agent reaches the step limit before it can implement the solution, failing.

A.5.4 Task-Agnostic Prompts

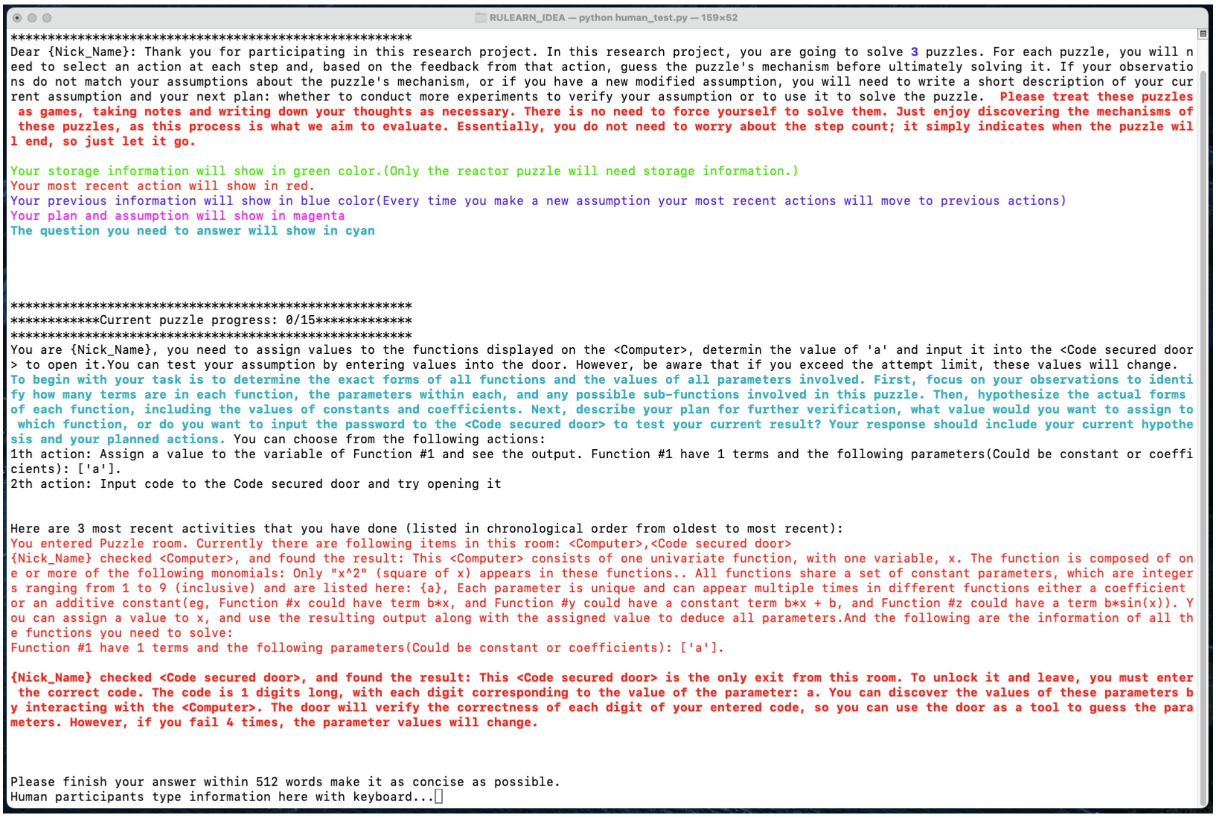
Task-Agnostic Abduction Prompt

```
f'''{GOAL} Your task is to develop general, clearly falsifiable, explanatory rules that explains your
observations, a process known as abduction. Please consider the given observations and propose an
initial hypothesis that explains them, make sure your hypothesis is robust and align with all your
observations. Your response should include your current hypothesis and your planned actions.\nPlease
keep your analysis, hypothesis, and planned actions as concise and precise as possible.'''
```

Task-Agnostic Induction Prompt

```
f'''{GOAL} Your task is to validate and modify your previous hypothesis, detailed here: {
Previous_hypothesis}, using your new observations. Review your most recent observation: {
buffer_memory_str}, to determine if your current hypothesis is still valid. If it is, describe the next
steps you plan to take towards your goal. If it is not, revise your hypothesis to accurately reflect all
 observations, both recent and prior. Finally, provide a plan for your next steps. Your response should
include both your current hypothesis and your planned actions. You can choose from the following actions
:\n{action_space_str}\nPlease keep your analysis, hypothesis, and planned actions as concise and precise
 as possible.'''
```

A.6 Human participants interface



```
RULEARN_IDEA — python human_test.py — 159×52
********************************************************
Dear {Nick_Name}: Thank you for participating in this research project. In this research project, you are going to solve 3 puzzles. For each puzzle, you will n
eed to select an action at each step and, based on the feedback from that action, guess the puzzle's mechanism before ultimately solving it. If your observatio
ns do not match your assumptions about the puzzle's mechanism, or if you have a new modified assumption, you will need to write a short description of your cur
rent assumption and your next plan: whether to conduct more experiments to verify your assumption or to use it to solve the puzzle.  Please treat these puzzles
 as games, taking notes and writing down your thoughts as necessary. There is no need to force yourself to solve them. Just enjoy discovering the mechanisms of
 these puzzles, as this process is what we aim to evaluate. Essentially, you do not need to worry about the step count; it simply indicates when the puzzle wil
l end, so just let it go.

Your storage information will show in green color.(Only the reactor puzzle will need storage information.)
Your most recent action will show in red.
Your previous information will show in blue color(Every time you make a new assumption your most recent actions will move to previous actions)
Your plan and assumption will show in magenta
The question you need to answer will show in cyan



********************************************************
************Current puzzle progress: 0/15*************
********************************************************
You are {Nick_Name}, you need to assign values to the functions displayed on the <Computer>, determin the value of 'a' and input it into the <Code secured door
> to open it.You can test your assumption by entering values into the door. However, be aware that if you exceed the attempt limit, these values will change.
To begin with your task is to determine the exact forms of all functions and the values of all parameters involved. First, focus on your observations to identi
fy how many terms are in each function, the parameters within each, and any possible sub-functions involved in this puzzle. Then, hypothesize the actual forms
of each function, including the values of constants and coefficients. Next, describe your plan for further verification, what value would you want to assign to
 which function, or do you want to input the password to the <Code secured door> to test your current result? Your response should include your current hypothe
sis and your planned actions. You can choose from the following actions:
1th action: Assign a value to the variable of Function #1 and see the output. Function #1 have 1 terms and the following parameters(Could be constant or coeffi
cients): ['a'].
2th action: Input code to the Code secured door and try opening it


Here are 3 most recent activities that you have done (listed in chronological order from oldest to most recent):
You entered Puzzle room. Currently there are following items in this room: <Computer>,<Code secured door>
{Nick_Name} checked <Computer>, and found the result: This <Computer> consists of one univariate function, with one variable, x. The function is composed of on
e or more of the following monomials: Only "x^2" (square of x) appears in these functions.. All functions share a set of constant parameters, which are integer
s ranging from 1 to 9 (inclusive) and are listed here: {a}, Each parameter is unique and can appear multiple times in different functions either a coefficient
or an additive constant(eg, Function #x could have term b*x, and Function #y could have a constant term b*x + b, and Function #z could have a term b*sin(x)). Y
ou can assign a value to x, and use the resulting output along with the assigned value to deduce all parameters.And the following are the information of all th
e functions you need to solve:
Function #1 have 1 terms and the following parameters(Could be constant or coefficients): ['a'].

{Nick_Name} checked <Code secured door>, and found the result: This <Code secured door> is the only exit from this room. To unlock it and leave, you must enter
 the correct code. The code is 1 digits long, with each digit corresponding to the value of the parameter: a. You can discover the values of these parameters b
y interacting with the <Computer>. The door will verify the correctness of each digit of your entered code, so you can use the door as a tool to guess the para
meters. However, if you fail 4 times, the parameter values will change.


Please finish your answer within 512 words make it as concise as possible.
Human participants type information here with keyboard...
```

Figure 18: An example of what human participants see during the experiment. Human participants view the same prompt as the LLM agents, except for additional overview information clarifying their responsibilities in the project. The name "{Nick_name}" is a pseudonym used to differentiate participants; we require participants to use only fictitious nicknames rather than their real names to ensure that the data collected is de-identified.

A.6.1 Hallucination examples

Agent's new observation contradict to current hypothesis

You are Kevin. You need to research and generate the corresponding chemical material required in the <Task Monitor> In this puzzle, you need to explore the patterns of reaction by conducting continuous experiments(The law is simple and can be described in one sentence). Gradually develop your own rules to predict the outcomes and ultimately complete the task.You know from an incomplete list of reaction equations that: XY+Z = ZXY.
Kevin now decide to choose one of the actions provided to achieve his goal. Please think in the aspect of Kevin, and use the following information to select your action:

Following is the actions that Kevin did previously:
You entered Chemical laboratory. Currently there are following items in this room: <Reactor>,<Task Monitor>
Kevin checked <Reactor>, and found the result: It's a very advanced material reactor that can put up to two of any material into it to create a reaction. The reactor provided unlimited one-letter raw material and other materials you've already synthesized.
Kevin checked <Task Monitor>, and found the result: It's a quest display that shows you the current quest you need to complete, which currently reads, "Please synthesise a unit of <CADEA> materials and put it into the slot under the monitor."
After previous exploration, you have the following hypothesis and plan:
"hypothesis": "When combining two materials, the second material is added at the beginning of the first material's sequence.",
"plan": "To verify this hypothesis, I will put material <A> first and then material <C> into the reactor.", please act based on your hypothesis and plan.

Following is the 1 most recent things that Kevin have done under your current hypothesis:
By turning on the reactor A and C turned into CA after the reaction. And you put the products into your storage for later use.
You currently have the following items in your storage: <A>, <B>, <C>, <D>, <E>, <CA>

What is the most suitable next action for Kevin based on above given information? Below are the available actions:
1th action: Choose material you want to synthesize, and turn on the Reactor.
2th action: Modify previous hypothesis and make a new plan: (Take this action when your current observations contradict your previous hypothesiss or your current plan is fulfilled.)
Above 2 provided actions are not yet performed by Kevin don't assume its outcome, please following the steps to generate your final answer. You MUST select one of the provided actions. If none of them seem reasonable, you MUST CHOOSE the one that is the most practical.
Step1: Review all the provided actions. Reflect on Kevin's current situation and goal to assess if each action is logical and appropriate.
Step2: Choose the most logical action. Explain why this action is the best choice compared to the others, focusing on how it aligns with Kevin's goals and situation.
Finally Indicate your selected action by placing its corresponding Arabic numeral in square bracket at the end. For example, if the third action is chosen, write [3]. Please do not use square bracket anywhere else other than final answer.

Agent fail to identify the contradiction

Generated_answer
"Step1": "Reviewing the two provided actions, the first action involves synthesizing a new material with the Reactor, while the second action involves modifying the hypothesis or plan. Kevin's observation so far aligns with his hypothesis.",
"Step2": "Given that the hypothesis has been verified (CA from A and C), Kevin should continue with the synthesis to achieve the goal of CADEA. Utilizing available materials and continuing with described pattern CA lead to CA -> CAA and onwards towards CADEA.",
"Selected_Action": "[1]"

The agent has already verified its hypothesis by synthesizing CA from A + C; however, in the next step, the agent tends to synthesize CAA, which is not part of the final target product. Despite knowing the rule, the agent fails to leverage this rule effectively towards achieving the final goal.

Figure 19: Example of GPT-4o agent failing to leverage rule towards the result.

Agent's new observation contradict to current hypothesis

You are Kevin. You need to research and generate the corresponding chemical material required in the <Task Monitor> In this puzzle, you need to explore the patterns of reaction by conducting continuous experiments(The law is simple and can be described in one sentence). Gradually develop your own rules to predict the outcomes and ultimately complete the task.You know from an incomplete list of reaction equations that: XY+Z = XZY.

Your task is to validate and modify your previous hypothesis, detailed here:
"current_hypothesis": "The reaction pattern follows the format AB+C = ACB, where the first reactant's elements are rearranged with the second reactant at the end.",
"plan_for_verification": "To verify the rule, I will test the reaction between materials A and C to check if the outcome aligns with the expected pattern of ACB.", using your new observations. Review your most recent observation: Following is the 6 most recent things that Kevin have done
Following is the actions that Kevin did previously:
You entered Chemical laboratory. Currently there are following items in this room: <Reactor>,<Task Monitor>
Kevin checked <Task Monitor>, and found the result: It's a quest display that shows you the current quest you need to complete, which currently reads, "Please synthesise a unit of <ACB> materials and put it into the slot under the monitor."
Kevin checked <Reactor>, and found the result: It's a very advanced material reactor that can put up to two of any material into it to create a reaction. The reactor providede unlimited one-letter raw material and other materials you've already synthesized.
By turning on the reactor A and C turned into AC after the reaction. And you put the products into your storage for later use.
By turning on the reactor AC and B turned into ABC after the reaction. And you put the products into your storage for later use.
By turning on the reactor AC and B turned into ABC after the reaction. And you put the products into your storage for later use.
By turning on the reactor AC and B turned into ABC after the reaction. And you put the products into your storage for later use.
By turning on the reactor ABC and A turned into ABAC after the reaction. And you put the products into your storage for later use.

Determine if your current hypothesis is still valid. If it is, describe the next steps you plan to take towards your goal. If it is not, revise your hypothesis to accurately reflect all observations, both recent and prior. Finally, provide a plan for your next steps. Your response should include both your current hypothesis and your planned actions.

Agent fail to identify the contradiction

Generated_answer:

"current_hypothesis": "The reaction pattern allows combining the elements of the first reactant with the second reactant directly without rearrangement.",
"plan_for_verification": "Test the reaction by combining B and XY to check if it results in BXY.

Agent already inferred from the initial observations that AB+C = ACB and listed that in the hypothesis. However, it failed to leverage the hypothesis effectively, and the refined hypothesis was nonsensical given the observations, leading the hypothesis further away from the truth.

Figure 20: Example of hallucination by GPT-4o when refining previous hypothesis.

Agent's new observation contradict to current hypothesis

You are Kevin. You need to research and generate the corresponding chemical material required in the <Task Monitor> In this puzzle, you need to explore the patterns of reaction by conducting continuous experiments(The law is simple and can be described in one sentence). Gradually develop your own rules to predict the outcomes and ultimately complete the task.You know from an incomplete list of reaction equations that: XY+Z = ZXY.
Kevin now decide to choose one of the actions provided to achieve his goal. Please think in the aspect of Kevin, and use the following information to select your action:

Following is the actions that Kevin did previously:
You entered Chemical laboratory. Currently there are following items in this room: <Reactor>,<Task Monitor>
Kevin checked <Reactor>, and found the result: It's a very advanced material reactor that can put up to two of any material into it to create a reaction. The reactor provided unlimited one-letter raw material and other materials you've already synthesized.
Kevin checked <Task Monitor>, and found the result: It's a quest display that shows you the current quest you need to complete, which currently reads, "Please synthesise a unit of <CCADD> materials and put it into the slot under the monitor."

After previous exploration, you have the following hypothesis and plan:
Initial Hypothesis: Materials combine in a consistent, alphabetical order, resulting in the longest possible string of unique letters.
To test this hypothesis, I will combine <A> and <B> to see if AB is produced, and then combine <C> and <D> to see if CD is produced. If my hypothesis holds, these reactions will occur, and I can further refine my rule., please act based on your hypothesis and plan.

Following is the 1 most recent things that Kevin have done under your current hypothesis:
By turning on the reactor A and B turned into BA after the reaction. And you put the products into your storage for later use.

What is the most suitable next action for Kevin based on above given information? Below are the available actions:
1th action: Choose material you want to synthesize, and turn on the Reactor.
2th action: Modify previous hypothesis and make a new plan: (Take this action when your current observations contradict your previous hypothesis or your current plan is fulfilled.)

Above 2 provided actions are not yet performed by Kevin don't assume its outcome, please following the steps to generate your final answer. You MUST select one of the provided actions. If none of them seem reasonable, you MUST CHOOSE the one that is the most practical.
Step1: Review all the provided actions. Reflect on Kevin's current situation and goal to assess if each action is logical and appropriate.
Step2: Choose the most logical action. Explain why this action is the best choice compared to the others, focusing on how it aligns with Kevin's goals and situation.
Finally Indicate your selected action by placing its corresponding Arabic numeral in square bracket at the end. For example, if the third action is chosen, write [3]. Please do not use square bracket anywhere else other than final answer.

Agent fail to identify the contradiction

Generated_answer:

Upon reviewing the provided actions, I find that action 1 is the most logical choice. Since Kevin's goal is to synthesize CCADD and he has already tested his initial hypothesis by combining A and B, the next step would be to continue testing his hypothesis by combining C and D. This aligns with his plan to refine his rule and ultimately achieve his goal. Kevin can now proceed to combine C and D in the reactor. [1]

The agent hypothesized that the materials combine in a consistent, alphabetical order; however, its most recent experiment shows the result A+B = BA, which contradicts this hypothesis. Noticing the contradiction, the agent should choose the second action to refine its hypothesis rather than continue to synthesize the materials under the incorrect hypothesis.

Figure 21: Example of Llama-3 70B agent failing to detect a contradiction in experimental results.

Context and prompt for agent inputting password in Room Escape puzzle

Based on the following recent experience of Kevin:
In this puzzle, set in an art gallery, Kevin must uncover the password for the <Code Secured Door> by discovering the relationships between the password and the paintings. And finally input the password into the <Code Secured door>. Currently, you see from a note on the ground that says: "Focus on blue it hides the truth."You can test your assumption by entering the password into the door. However, be aware that if you exceed the attempt limit, the password and hint will change.
Kevin's current action is: Input code to the Code Secured Door and try opening it And you have the following information to decide what is the correct password:

Following is the 13 most recent things that Kevin've done:
You entered Corridor. Currently there are following items in this room: <Code Secured Door>,<Flower Oil Paint>,<Lemon Grove Oil Paint>,<Coastal Serenity Oil Paint>,<Sapphire Night Oil Paint>,<Jungle Mist Acrylic Paint>,<Winter Chill Acrylic Paint>,<Arctic Silence Watercolor Paint>
Kevin checked <Code Secured Door>, and found the result: This coded door is the only exit from this room, and you can only leave this room by entering the correct code. The password consists of three digits.
Kevin checked <Coastal Serenity Oil Paint>, and found the result: It's an oil painting of a blue coast.
Kevin checked <Arctic Silence Watercolor Paint>, and found the result: It's an watercolor painting of blue arctic.
Kevin checked <Lemon Grove Oil Paint>, and found the result: It's an oil painting of a yellow lemon grove.
Kevin checked <Flower Oil Paint>, and found the result: It's an oil painting of a yellow flower.
Kevin checked <Sapphire Night Oil Paint>, and found the result: It's an oil painting of a blue night sky.
Kevin checked <Jungle Mist Acrylic Paint>, and found the result: It's an acrylic painting of green jungle.
Kevin checked <Winter Chill Acrylic Paint>, and found the result: It's an acrylic painting of blue winter scene.
Kevin tried the following action: Input code to the Code Secured Door and try opening it, Kevin's decision is 123, but the password was incorrect.Kevin's first digit 1 is incorrect. Kevin's second digit 2 is incorrect. Kevin's third digit 3 is incorrect. You can still try 2 times before the password change!
Kevin tried the following action: Input code to the Code Secured Door and try opening it, Kevin's decision is 258, but the password was incorrect.Kevin's first digit 2 is correct. Kevin's second digit 5 is incorrect. Kevin's third digit 8 is incorrect. You can still try 1 times before the password change!

Please follow the following steps to generate your final answer.
Step1 refelect the recent experience, what do you think is the password to <Code Secured Door> is? Please only use information provided to do inference and give your reason. **Final Step** Please generate your final answer in a pair of square brackets. eg, if you think the final pass word is '123' you should output [123], if you think the output is '324' please output ['324'].

Agent's answer

Based on the recent experience, I think the password to <Code Secured Door> is 289. My reason is that the hint "Focus on blue it hides the truth" suggests that the correct password is related to the blue-colored paintings. The correct first digit 2 is likely from the blue <Coastal Serenity Oil Paint> or <Winter Chill Acrylic Paint>. The correct second digit 8 is likely from the blue <Sapphire Night Oil Paint>. The correct third digit 9 is likely from the blue <Arctic Silence Watercolor Paint>.

[289]

Figure 22: Example of hallucination by Llama-3 70B when generating a password.

A.7 Puzzle Examples

Function Operator Puzzle No. 2 (3 functions, 2 unknown parameters, 2 elementary functions)

Goal: You are Kevin. You need to assign values to the functions displayed on the <Computer>, determine the values of 'a' and 'b'. Then, input these values into the <Code secured door> in alphabetical order to open it.You can test your hypothesis by entering values into the door. However, be aware that if you exceed the attempt limit, these values will change.

Initial Memories:

1, Kevin checked <Code secured door>, and found the result: This <Code secured door> is the only exit from this room. To unlock it and leave, you must enter the correct code. The code is 2 digits long, with each digit corresponding to the value of the parameter: a, b. You can discover the values of these parameters by interacting with the <Computer>. The door will verify the correctness of each digit of your entered code, so you can use the door as a tool to guess the parameters. However, if you fail 4 times, the parameter values will change.

2, Kevin checked <Computer>, and found the result: This <Computer> consists of 2 different univariate functions, each with one variable, x. Each function is composed of one or more of the following monomials: "x^2" (square of x) and 'sin(x)'. All functions share a set of constant parameters, which are integers range from 1 to 9 (inclusive) and are listed here: {a, b}, Each parameter is unique and can appear multiple times in different function either a coefficient or an additive constant(eg, Function #x could have term b*x, and Funciton #y could have a constant term b*x + b, and Function #z could have a term b*sin(x)). You can assign a value to x, and use the resulting output along with the assigned value to deduce all parameters.And the following are the information of all the functions you need to solve:

Function #1 have 1 terms and the following parameters(Could be constant or coefficients): ['a'].

Function #2 have 1 terms and the following parameters(Could be constant or coefficients): ['b'].

Provided Interactive actions:

1th action: Input code to the Code secured door and try opening it

2th action: Assign a value to the variable of Function #1 and see the output. Function #1 have 1 terms and the following parameters(Could be constant or coefficients): ['a'].

3th action: Assign a value to the variable of Function #2 and see the output. Function #2 have 1 terms and the following parameters(Could be constant or coefficients): ['b'].

Figure 23: Function operator puzzle No. 2.

Function Operator Puzzle No. 17 (3 functions, 3 unknown parameters, 4 elementary functions)

Goal: You are Kevin. You need to assign values to the functions displayed on the <Computer>, determine the values of 'a', 'b', 'c'. Then, input these values into the <Code secured door> in alphabetical order to open it.You can test your hypothesis by entering values into the door. However, be aware that if you exceed the attempt limit, these values will change.

Initial Memories:

1, Kevin checked <Code secured door>, and found the result: This <Code secured door> is the only exit from this room. To unlock it and leave, you must enter the correct code. The code is 3 digits long, with each digit corresponding to the value of the parameter: a, b, c. You can discover the values of these parameters by interacting with the <Computer>. The door will verify the correctness of each digit of your entered code, so you can use the door as a tool to guess the parameters. However, if you fail 4 times, the parameter values will change.

2, Kevin checked <Computer>, and found the result: This <Computer> consists of 3 different univariate functions, each with one variable, x. Each function is composed of one or more of the following monomials: '|x|' (absolute value of x),'x',"x^2" (square of x) and 'sin(x)'. All functions share a set of constant parameters, which are integers range from 1 to 9 (inclusive) and are listed here: {a, b, c}, Each parameter is unique and can appear multiple times in different function either a coefficient or an additive constant(eg, Function #x could have term b*x, and Function #y could have a constant term b*x + b, and Function #z could have a term b*sin(x)). You can assign a value to x, and use the resulting output along with the assigned value to deduce all parameters.And the following are the information of all the functions you need to solve:

Function #1 have 2 terms and the following parameters(Could be constant or coefficients): ['a', 'b'].

Function #2 have 2 terms and the following parameters(Could be constant or coefficients): ['a', 'c'].

Function #3 have 1 terms and the following parameters(Could be constant or coefficients): ['c'].

Provided Interactive actions:

1th action: Input code to the Code secured door and try opening it

2th action: Assign a value to the variable of Function #2 and see the output. Function #2 have 2 terms and the following parameters(Could be constant or coefficients): ['a', 'c'].

3th action: Assign a value to the variable of Function #1 and see the output. Function #1 have 2 terms and the following parameters(Could be constant or coefficients): ['a', 'b'].

4th action: Assign a value to the variable of Function #3 and see the output. Function #3 have 1 terms and the following parameters(Could be constant or coefficients): ['c'].

Figure 24: Function operator puzzle No. 17.

Escape Room Puzzle No. 3 (6 Paintings, All Paintings visible)

Goal: In this puzzle, set in an art gallery, Kevin must uncover the password for the <Code Secured Door> by discovering the relationships between the password and the paintings. And finally input the password into the <Code Secured door>. You can test your hypothesis by entering the password into the door. However, be aware that if you exceed the attempt limit, the password and hint will change.

Initial Memories:

1, You entered Corridor. Currently there are following items in this room: <Code Secured Door>,<Emerald Tide Oil Paint>,<Sapphire Night Oil Paint>,<Bluebell Woods Acrylic Paint>,<Morning Glory Acrylic Paint>,<Arctic Silence Watercolor Paint>,<River Reflections Watercolor Paint>
2, Kevin checked <Code Secured Door>, and found the result: This coded door is the only exit from this room, and you can only leave this room by entering the correct code. The password consists of three digits.
3, Kevin checked <Bluebell Woods Acrylic Paint>, and found the result: It's an acrylic painting of blue woods.
4, Kevin checked <Arctic Silence Watercolor Paint>, and found the result: It's an watercolor painting of blue arctic.
5, Kevin checked <River Reflections Watercolor Paint>, and found the result: It's an watercolor painting of blue river.
6, Kevin checked <Morning Glory Acrylic Paint>, and found the result: It's an acrylic painting of yellow morning.
7, Kevin checked <Sapphire Night Oil Paint>, and found the result: It's an oil painting of a blue night sky.
8, Kevin checked <Emerald Tide Oil Paint>, and found the result: It's an oil painting of a green tide.
9, Currently, you see from a note on the ground that says: "Focus on blue it hides the truth."

Provided Interactive actions:

1th action: Input code to the Code Secured Door and try opening it
2th action: Try opening the Code Secured Door with brute force

Figure 25: Escape room puzzle No. 3

Escape Room Puzzle No. 13 (6 Paintings, Need to actively explore the gallery to reveal all paintings)

Goal: In this puzzle, set in an art gallery, Kevin must uncover the password for the <Code Secured Door> by discovering the relationships between the password and the paintings. And finally input the password into the <Code Secured door>. You can test your hypothesis by entering the password into the door. However, be aware that if you exceed the attempt limit, the password and hint will change.

Initial Memories:

1, You entered Oil Painting Gallery. Currently there are following items in this room: <Emerald Tide Oil Paint>,<Sapphire Night Oil Paint>,<Code Secured Door>,<Watercolour Gallery Entrance>,<Acrylic Painting Gallery Entrance>
2, Kevin checked <Code Secured Door>, and found the result: This coded door is the only exit from this room, and you can only leave this room by entering the correct code. The password consists of three digits.
3, Kevin checked <Watercolour Gallery Entrance>, and found the result: It's an automatic door with a poster next to it that says "Watercolour Gallery".
4, Kevin checked <Acrylic Painting Gallery Entrance>, and found the result: It's an automatic door with a poster next to it that says "Acrylic Painting Gallery."
5, Kevin checked <Sapphire Night Oil Paint>, and found the result: It's an oil painting of a blue night sky.
6, Kevin checked <Emerald Tide Oil Paint>, and found the result: It's an oil painting of a green tide.
7, Currently, you see from a note on the ground that says: "Focus on blue it hides the truth."

Provided Interactive actions:

1th action: Input code to the Code Secured Door and try opening it
2th action: Try opening the Code Secured Door with brute force
3th action: Pass through the Watercolour Gallery Entrance and reaches the Watercolour Gallery.
4th action: Pass through the Acrylic Painting Gallery Entrance and reaches the Acrylic Painting Gallery.

Figure 26: Escape room puzzle No. 13

Reactor Puzzle No. 8 (Reverse concatenation rule, Target material: “CADEA”)

Goal: You are Kevin. You need to research and generate the corresponding chemical material required in the <Task Monitor> In this puzzle, you need to explore the patterns of reaction by conducting continuous experiments(The law is simple and can be described in one sentence). Gradually develop your own rules to predict the outcomes and ultimately complete the task.

Initial Memories:

1, Kevin checked <Reactor>, and found the result: It's a very advanced material reactor that can put up to two of any material into it to create a reaction. The reactor provided unlimited one-letter raw material and other materials you've already synthesized.

2, Kevin checked <Task Monitor>, and found the result: It's a quest display that shows you the current quest you need to complete, which currently reads, "Please synthesise a unit of <CADEA> materials and put it into the slot under the monitor."

3, You currently have the following items in your storage: <A>, <B>, <C>, <D>, <E>.

4, You know from an incomplete list of reaction equations that: XY+Z = ZXY.

Provided Interactive actions:

1th action: Choose material you want to synthesize, and turn on the Reactor.

Reactor Puzzle No. 14 (Middle insertion rule, Target material: “ABCDEF”)

Goal: You are Kevin. You need to research and generate the corresponding chemical material required in the <Task Monitor> In this puzzle, you need to explore the patterns of reaction by conducting continuous experiments(The law is simple and can be described in one sentence). Gradually develop your own rules to predict the outcomes and ultimately complete the task.

Initial Memories:

1, Kevin checked <Reactor>, and found the result: It's a very advanced material reactor that can put up to two of any material into it to create a reaction. The reactor provided unlimited one-letter raw material and other materials you've already synthesized.

2, Kevin checked <Task Monitor>, and found the result: It's a quest display that shows you the current quest you need to complete, which currently reads, "Please synthesise a unit of <ABCDEF> materials and put it into the slot under the monitor."

3, You currently have the following items in your storage: <A>, <B>, <C>, <D>, <E>, <F>

4, You know from an incomplete list of reaction equations that: XY+Z = XZY.

Provided Interactive actions:

1th action: Choose material you want to synthesize, and turn on the Reactor.

Figure 27: Reactor puzzle No. 8 and No. 14