

大模型驱动的自主智能体

复旦大学

数据智能与社会计算实验室 (Fudan DISC)

默认大家已经知道的

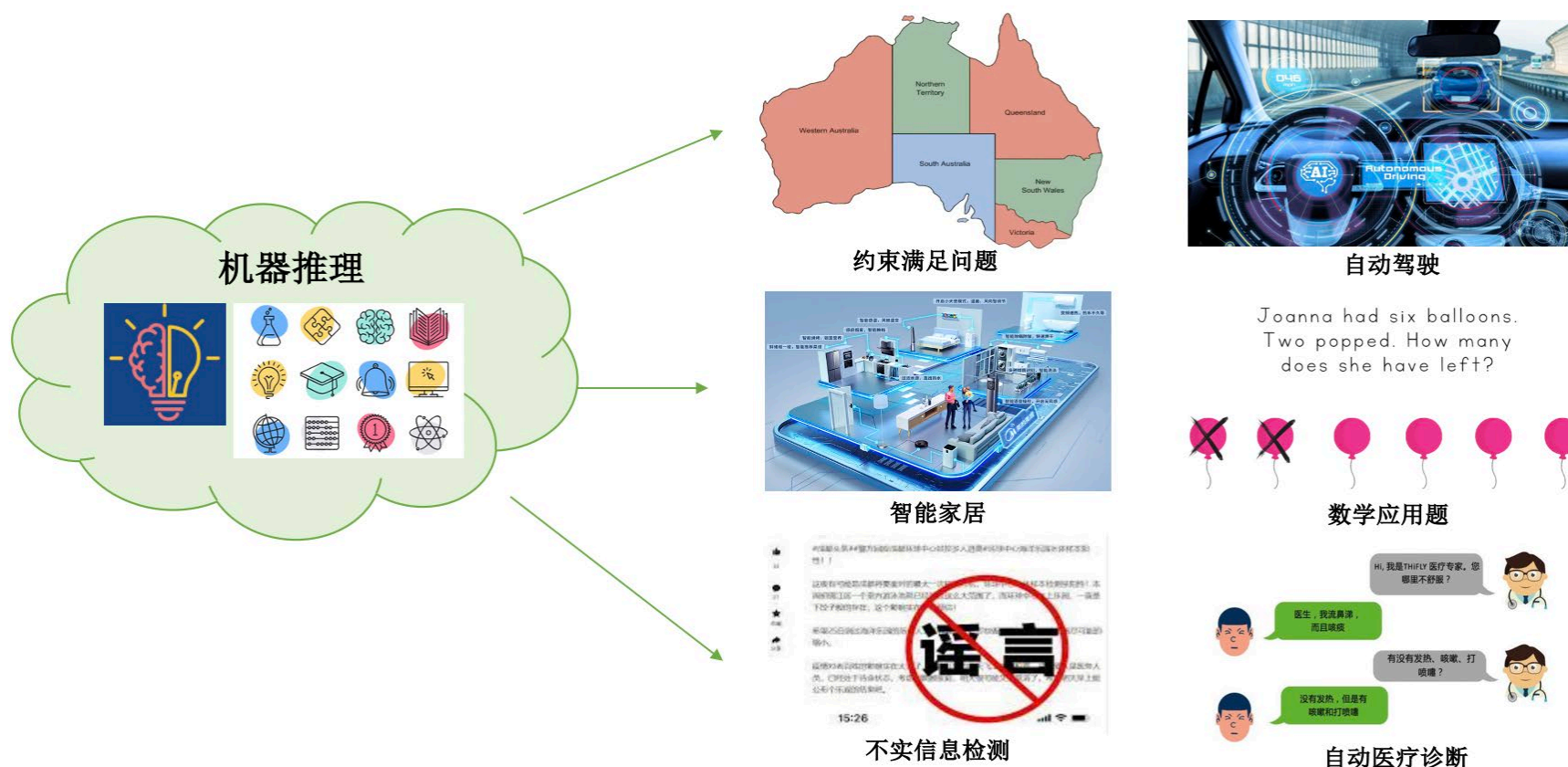
- 大语言模型的架构 (encoder-decoder, decoder-only)
- 大语言模型预训练方法 (MLM, LM)
- 指令微调的基本方法 (指令集构建)
- 一些开源模型 (LLaMA, ChatGLM, MOSS, CPM)

目录

- 大语言模型驱动单体智能
 - **大语言模型的复杂任务能力**
 - 大语言模型的单体能力拓展
 - 大语言模型的能力自我提升
- 大语言模型驱动群体智能
 - 智能体的记忆能力
 - 多智能体的社区模拟
 - 多智能体的任务协作

机器推理

根据给定的条件，建模相关知识，使用推理方法（演绎/归纳等）得出新结论。



前大模型时代的机器推理



- 研究内容一：基于知识建模的机器推理
- 研究内容二：符号逻辑驱动的机器推理
- 研究内容三：推理过程的可解释性增强
- 研究内容四：结合推理方法的泛化性

基于知识建模的机器推理

问题自动生成

Oxygen is used in cellular respiration and released by photosynthesis, which uses the energy of sunlight to produce oxygen from water.



- What is used in cellular respiration?



给定一段文本, 如何控制机器自动生成多样化且有意义的问题?



如何控制机器生成更可靠的问题?
(和给定文本更相关, 信息量更充足的问题)

Sentence:

The game was played on February 7, 2016, at Levi's Stadium in the San Francisco Bay Area at Santa Clara, California.

Generated questions :

- Q1: On what date was the game played ?
- Q2: Where was the Everton Fc Stadium located?

uninformative

Ground truth questions:

- GTQ1: When was the game at Levi's Stadium played?
- GTQ2: Which state is the Levi's Stadium in?

irrelevant

基于知识建模的自动化问题生成

复杂问题生成：从文本中构建**知识三元组**，融合三元组信息进行推理。



基于文本中多个事实，可以帮助生成和文本更一致、更具有信息量的问题

Sentence:

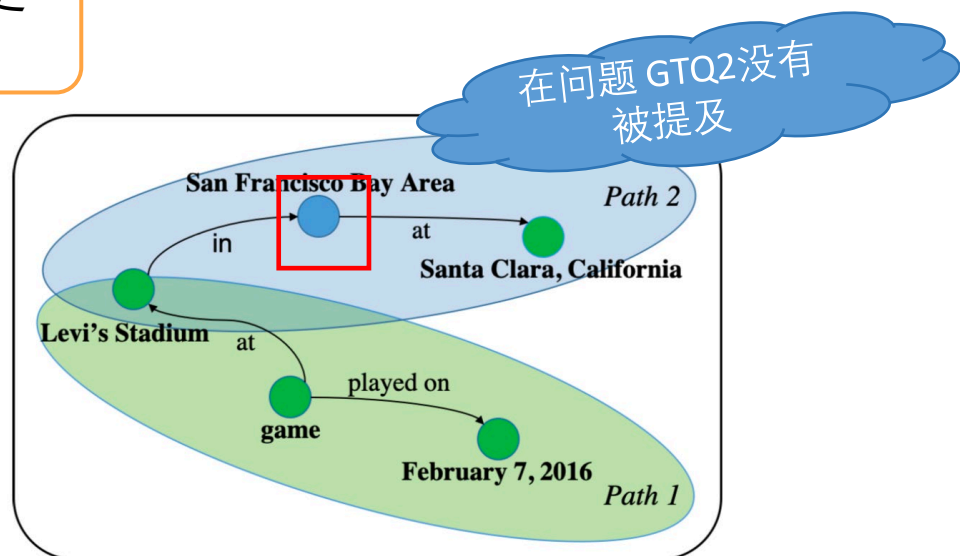
The game was played on February 7, 2016, at Levi's Stadium in the San Francisco Bay Area at Santa Clara, California.

Generated questions:

- Q1: On what date was the game played ?
- Q2: Where was the Everton Fc Stadium located?

Ground truth questions:

- GTQ1: When was the game at Levi's Stadium played?
- GTQ2: Which state is the Levi's Stadium in?



建模文本中的**事实查询路径**（多个事实三元组构成的有序序列），并推断出目标问题可能**涉及到的内容**来引导更可靠的推理，生成更一致更有信息的问题。

两个步骤: 1) 查询表示学习（query representation learning），来推断查询路径中目标问题可能涉及的内容；2) 基于查询表示的问题生成（query-based question generation）

结合符号逻辑规律的文本逻辑推理问答



引入**符号逻辑规律** (symbolic logical rules/laws)
来帮助建模显式推理过程

提出一个三步骤逻辑驱动推理范式：

- **步骤-1:** 从文本中识别出逻辑表达式作为基本推理单元，如 $(\neg\alpha \rightarrow \neg\beta)$ ，来表示逻辑符号之间的逻辑关系。
- **步骤-2:** 根据逻辑等价律执行逻辑推断，从识别的逻辑表达式中扩展隐式存在的表达式。
- **步骤-3:** 通过比较候选答案和所有得到的逻辑表达式来选出最有可能的答案。

Context:

If you have no keyboarding skills at all, you will not be able to use a computer. And if you are not able to use a computer, you will not be able to write your essays using a word processing program.

Question:

If above statements are true, which one of the following must be true?

Options:

- A. If you are not able to write your essays using a word processing program, you have no keyboarding skills. $(\neg\gamma \rightarrow \neg\alpha)$
- B. If you are able to write your essays using a word processing program, you have at least some keyboarding skills. $(\gamma \rightarrow \alpha)$ ✓
- C. If you are not able to write your essays using a word processing program, you are not able to use a computer. $(\neg\gamma \rightarrow \neg\beta)$
- D. If you have some keyboarding skills, you will be able to write your essays using a word processing program. $(\alpha \rightarrow \gamma)$

Logical Symbols :

α : have keyboarding skills
 β : be able to use a compute
 γ : be able to write your essays using
a word processing program

Logical Expressions :

$(\neg\alpha \rightarrow \neg\beta)$
 $(\neg\beta \rightarrow \neg\gamma)$

Extend the Implicit Logical Expressions by Laws:

$(\neg\alpha \rightarrow \neg\beta) \Rightarrow (\beta \rightarrow \alpha)$ Contrapostion
 $(\neg\beta \rightarrow \neg\gamma) \Rightarrow (\gamma \rightarrow \beta)$ Contrapostion
 $(\neg\alpha \rightarrow \neg\beta) \wedge (\neg\beta \rightarrow \neg\gamma) \Rightarrow (\neg\alpha \rightarrow \neg\gamma)$ Transitive Law
 $(\beta \rightarrow \alpha) \wedge (\gamma \rightarrow \beta) \Rightarrow (\gamma \rightarrow \alpha)$ Transitive Law

结合符号逻辑指令的文本分析推理问答

文本分析推理问题

极具挑战性!! 当前预训练模型只能实现随机猜测效果
BERT:21.4%, RoBERTa: 23.1%, ALBERT: 23.0%



引入**符号逻辑指令** (symbolic logical instructions) 来帮助显式执行分析推理过程，预测答案。

将自然语言描述的约束解析成**符号指令** (由一组预定义逻辑函数组成)，离散地执行分析推理:

- 自动抽取文本中的参数，包括参与者和值域
- 标注符号指令数据，来训练一个**语法约束的生成模型**，实现约束文本到符号指令的解析

Context :

A closet contains exactly six hangers—1, 2, 3, 4, 5, and 6—hanging, in that order, from left to right. It also contains exactly six dresses—one gauze, one linen, one polyester, one rayon, one silk, and one wool—a different dress on each of the hangers, in an order satisfying the following conditions:

The gauze dress is on a lower-numbered hanger than the polyester dress. **(C-1)**

The rayon dress is on hanger 1 or hanger 6. **(C-2)**

Either the wool dress or the silk dress is on hanger 3. **(C-3)**

The linen dress hangs immediately to the right of the silk dress. **(C-4)**

Question : Which one of the following CANNOT be true?

Options :

- A. The linen dress hangs immediately next to the gauze dress.
- B. The polyester dress hangs immediately to the right of the rayon dress.**
- C. The rayon dress hangs immediately to the left of the wool dress.
- D. The silk dress is on a lower-numbered hanger than the gauze dress.
- E. The wool dress is on a higher-numbered hanger than the rayon dress.

Annotated Participants & Positions	Participants: <i>gauze, linen, polyester, rayon, silk, wool</i> Positions: 1, 2, 3, 4, 5, 6
Annotated Programs	C-1: $VALUE(gauze) < VALUE(polyester)$ C-2: $VALUE(rayon) = 1 \text{ OR } VALUE(rayon) = 6$ C-3: $VALUE(wool) = 3 \text{ OR } VALUE(silk) = 3$ C-4: $VALUE(linen) = VALUE(silk) + 1$ Q-A: $NOT\ ABS(VALUE(linen) - VALUE(gauze)) = 1$ Q-B: $NOT\ VALUE(polyester) = VALUE(rayon) + 1$ Q-C: $NOT\ VALUE(rayon) = VALUE(wool) - 1$ Q-D: $NOT\ VALUE(silk) < VALUE(gauze)$ Q-E: $NOT\ VALUE(wool) > VALUE(rayon)$

基于多跳问答逐步推理的可解释性研究

文本多跳推理

[Question]

Q: What city is the Marine Air Control Group 28 located in?

[Context]

P1: Marine Tactical Air Command Squadron 28 is a United States Marine Corps aviation command and control unit based at Marine Corps Air Station Cherry Point. They provide the 2nd Marine Aircraft Wings tactical headquarters and ...

P2: Marine Corps Air Station Cherry Point or MCAS Cherry Point is a United States Marine Corps airfield located in Havelock, North Carolina, United States, in the eastern part of the state. It ...

P3-P10: ...



基于问题分解的方法可以生成推理的解释，通过将多跳问题分解成多个单跳子问题和对子答案。

[Question Decomposition (Min et al., 2019)]

Sub-Q1: Which Marine Air Control Group 28?

Sub-Q2: What city is Marine Tactical Air Command Squadron 28 located in?

[Stepwise Decomposition]

Step1-S: Marine Tactical Air Command Squadron 28 is a ... control unit based at Marine Corps Air Station Cherry Point.

Step1-Q: Which is the base of Marine Air Control Group 28?

Step2-S: Marine Corps Air Station Cherry Point ... United States Marine Corps airfield located in Havelock, North Carolina ... state.

Step2-Q: What city is the Marine Corps Air Station Cherry Point located in?

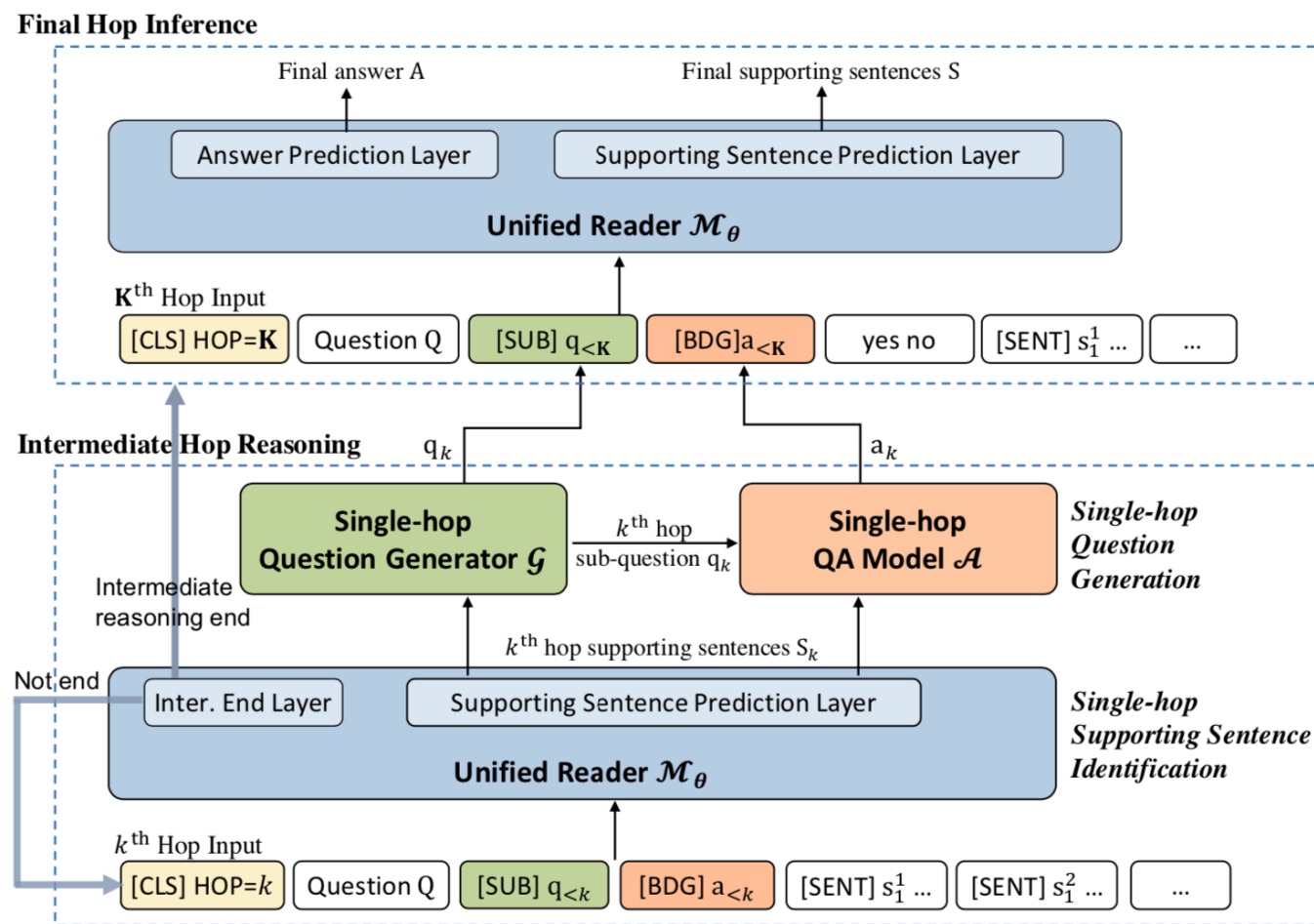
问题分解中没有考虑每一步的支撑事实，导致易于生成错误的分解和质量差的子问题

每个中间推理步骤应该包括这一步的支撑事实识别，帮助更好地生成子问题和子答案。

基于多跳问答逐步推理的可解释性研究

提出逐步推理框架 (stepwise reasoning framework), 自动生成可解释的推理过程。

- 在每个中间步骤, 实现单跳支撑事实识别、单跳问题生成和单跳问题回答。
- 使用预先训练好的简单问题生成器和回答模型实现子问题和子答案的生成。
- 利用当前跳的推断结果帮助下一跳推理, 直至输出最终答案。



Reasoning Process

Predicted Final Answer A:
What city is the Marine Corps Air Station Cherry Point located in?
A: Havelock

1st Hop Sub-answer a_1 :
Marine Corps Air Station Cherry Point

1st Hop Sub-question q_1 :
What is the base of Marine Air Control Group 28?

1st Hop Supporting Sentence S_1 :
Marine Tactical Air Command Squadron 28 is a United States Marine Corps aviation command and control unit based at Marine Corps Air Station Cherry Point.

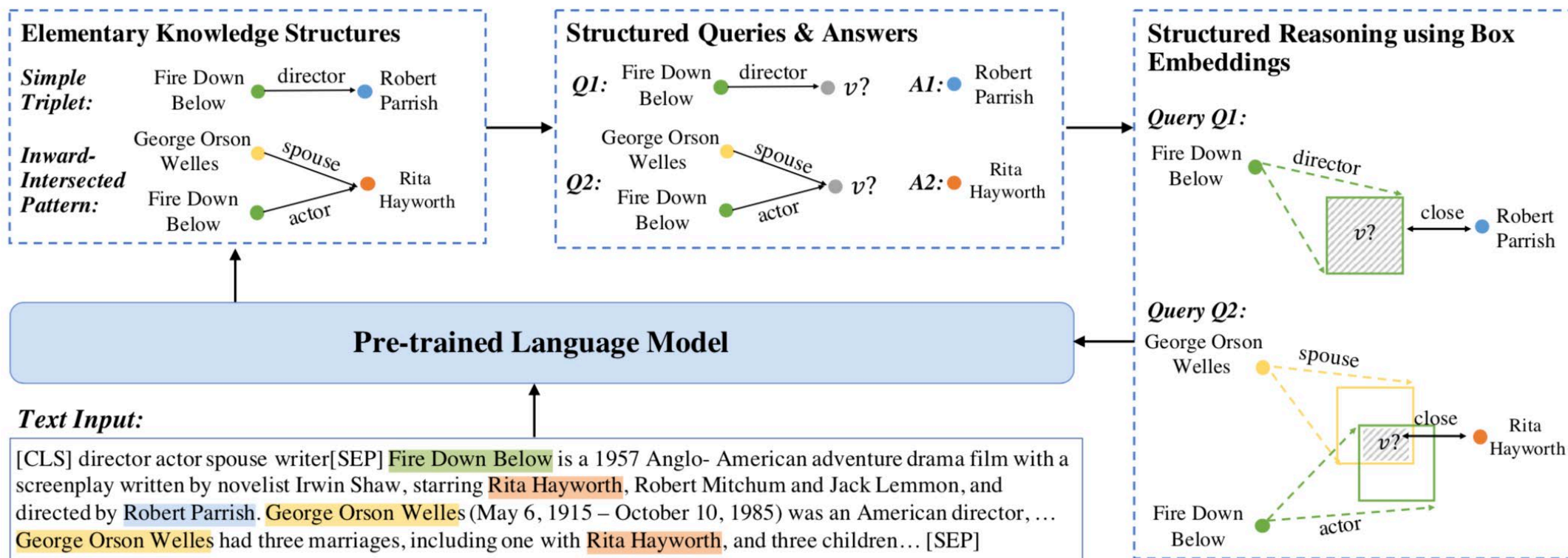
Question Q:
What city is the Marine Air Control Group 28 located in?

Question-relevant Context C:
P1: Marine Tactical Air Command Squadron 28 is a United States Marine...
P2: Marine Corps Air Station Cherry Point...located in **Havelock**, North Carolina...
P3 & P4: ...

基于结构推理能力预训练的泛化性研究

提出一个统一的学习框架，将显式结构推理和语言预训练相结合，赋予 PLMs 结构推理技能

首先识别出上下文中的四种基本结构来构建结构化查询，并沿着查询逐步进行推理，以识别答案实体。
在语义表示空间上采用 box embedding 进行逐步推理，实现了文本语义和结构推理的融合。



前大模型时代的机器推理任务的演化



- 研究内容一：基于知识建模的机器推理
- 研究内容二：符号逻辑驱动的机器推理
- 研究内容三：推理过程的可解释性增强
- 研究内容四：结合推理方法的泛化性

- 问题场景的复杂化
- 离散和连续的结合
- 中间过程的可解释性

Chain of Thoughts – 思维链

- 将问题解耦，一步步推理解决中间问题，最终得到问题答案。
- 小样本思维链（few-shot in-context prompting）
 - 利用上下文学习，提供少量样例，在样例中加入中间过程推理步骤
 - 让大语言模型模仿其中思维链的过程进行解题
- ✓ 给模型更多“思考时间”（文本输出）
- ✓ 完成对结果的回溯

□ 思维链的过程受限于所给样例

Standard Prompting

Model Input

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: The answer is 11.

Q: The cafeteria had 23 apples. If they used 20 to make lunch and bought 6 more, how many apples do they have?

Model Output

A: The answer is 27. ❌

Chain-of-Thought Prompting

Model Input

Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now?

A: Roger started with 5 balls. 2 cans of 3 tennis balls each is 6 tennis balls. $5 + 6 = 11$. The answer is 11.

Q: The cafeteria had 23 apples. If they used 20 to make lunch and bought 6 more, how many apples do they have?

Model Output

A: The cafeteria had 23 apples originally. They used 20 to make lunch. So they had $23 - 20 = 3$. They bought 6 more apples, so they have $3 + 6 = 9$. The answer is 9. ✓

Chain of Thoughts – 思维链Prompt的构造

- 针对数学问题（2类）、常识问答（2类）、日期识别、体育问题、机器人指令、字母操作、状态跟踪，构建了（输入、思维链、输出）的样例。
- 为每一类任务人工构建了8个样例，在测试的时候，使用同样的样例。

Math Word Problems (free response) Q: Roger has 5 tennis balls. He buys 2 more cans of tennis balls. Each can has 3 tennis balls. How many tennis balls does he have now? A: Roger started with 5 balls. 2 cans of 3 tennis balls each is 6 tennis balls. $5 + 6 = 11$. The answer is 11.	Math Word Problems (multiple choice) Q: How many keystrokes are needed to type the numbers from 1 to 500? Answer Choices: (a) 1156 (b) 1392 (c) 1480 (d) 1562 (e) 1788 A: There are 9 one-digit numbers from 1 to 9. There are 90 two-digit numbers from 10 to 99. There are 401 three-digit numbers from 100 to 500. $9 + 90(2) + 401(3) = 1392$. The answer is (b).	CSQA (commonsense) Q: Sammy wanted to go to where the people were. Where might he go? Options: (a) race track (b) populated areas (c) desert (d) apartment (e) roadblock A: The answer must be a place with a lot of people. Race tracks, desert, apartments, and roadblocks don't have a lot of people, but populated areas do. So the answer is (b).
StrategyQA Q: Yes or no: Would a pear sink in water? A: The density of a pear is about 0.6 g/cm^3, which is less than water. Thus, a pear would float. So the answer is no.	Date Understanding Q: The concert was scheduled to be on 06/01/1943, but was delayed by one day to today. What is the date 10 days ago in MM/DD/YYYY? A: One day after 06/01/1943 is 06/02/1943, so today is 06/02/1943. 10 days before today is 05/23/1943. So the answer is 05/23/1943.	Sports Understanding Q: Is the following sentence plausible? "Joao Moutinho caught the screen pass in the NFC championship." A: Joao Moutinho is a soccer player. The NFC championship is part of American football, not soccer. So the answer is no.
SayCan (Instructing a robot) Human: How would you bring me something that isn't a fruit? Explanation: the user wants something to eat that isn't a fruit. An energy bar is not a fruit, so I will bring the user an energy bar. Plan: 1. find(energy bar) 2. pick(energy bar) 3. find(user) 4. put(energy bar) 5. done().	Last Letter Concatenation Q: Take the last letters of the words in "Lady Gaga" and concatenate them. A: The last letter of "Lady" is "y". The last letter of "Gaga" is "a". Concatenating them is "ya". So the answer is ya.	Coin Flip (state tracking) Q: A coin is heads up. Maybelle flips the coin. Shalonda does not flip the coin. Is the coin still heads up? A: The coin was flipped by Maybelle. So the coin was flipped 1 time, which is an odd number. The coin started heads up, so after an odd number of flips, it will be tails up. So the answer is no.

Figure 3: Examples of (input, chain of thought, output) triples for arithmetic, commonsense, and symbolic reasoning benchmarks. Chains of thought are highlighted. Full prompts in Appendix G.

Chain of Thoughts – 思维链的效果

- 针对数学能力（左）、常识理解（中）、符号推理（右）三大类进行测试。

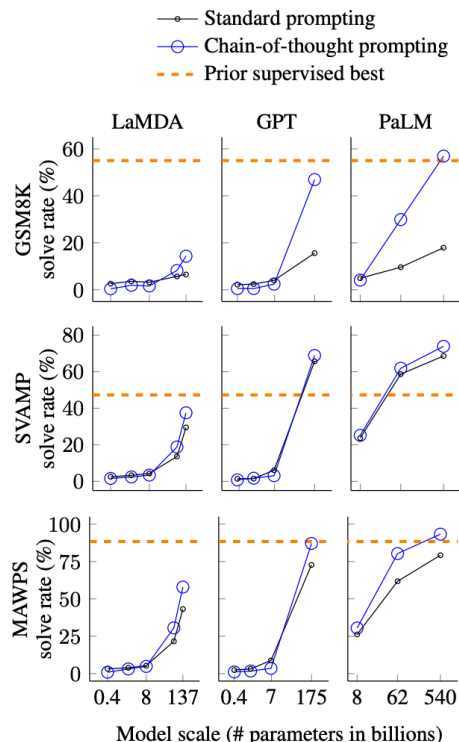


Figure 4: Chain-of-thought prompting enables large language models to solve challenging math problems. Notably, chain-of-thought reasoning is an emergent ability of increasing model scale. Prior best numbers are from Cobbe et al. (2021) for GSM8K, Jie et al. (2022) for SVAMP, and Lan et al. (2021) for MAWPS.

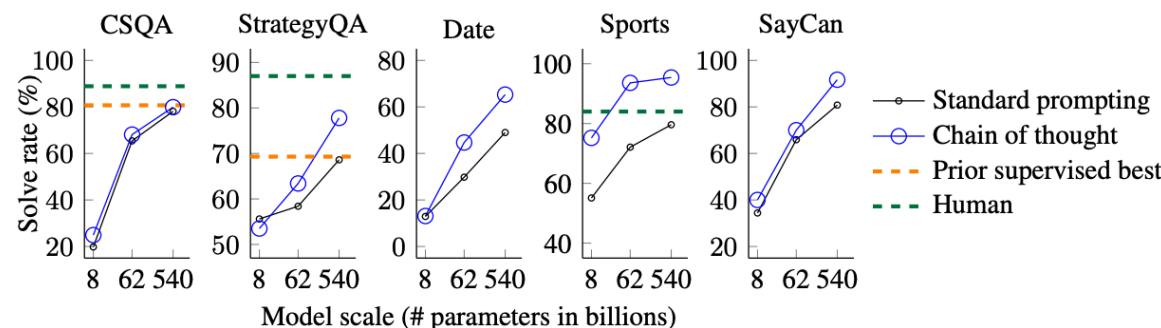


Figure 7: Chain-of-thought prompting also improves the commonsense reasoning abilities of language models. The language model shown here is PaLM. Prior best numbers are from the leaderboards of CSQA (Talmor et al., 2019) and StrategyQA (Geva et al., 2021) (single-model only, as of May 5, 2022). Additional results using various sizes of LaMDA, GPT-3, and PaLM are shown in Table 4.

- 基础模型的参数规模越大，思维链的优势越明显
- 当参数规模足够大，思维链大模型可以超过特定任务的模型
- 思维链使用的小样本有足够的稳定性
- 启发了微调指令集中的CoT指令

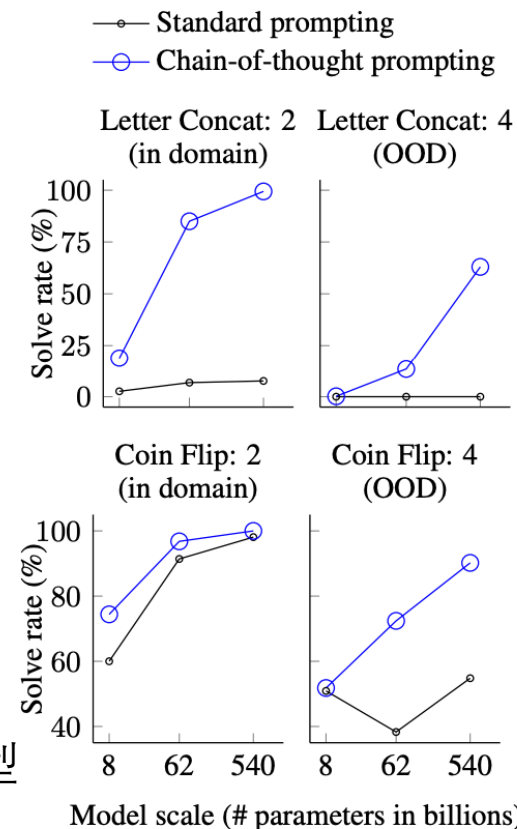
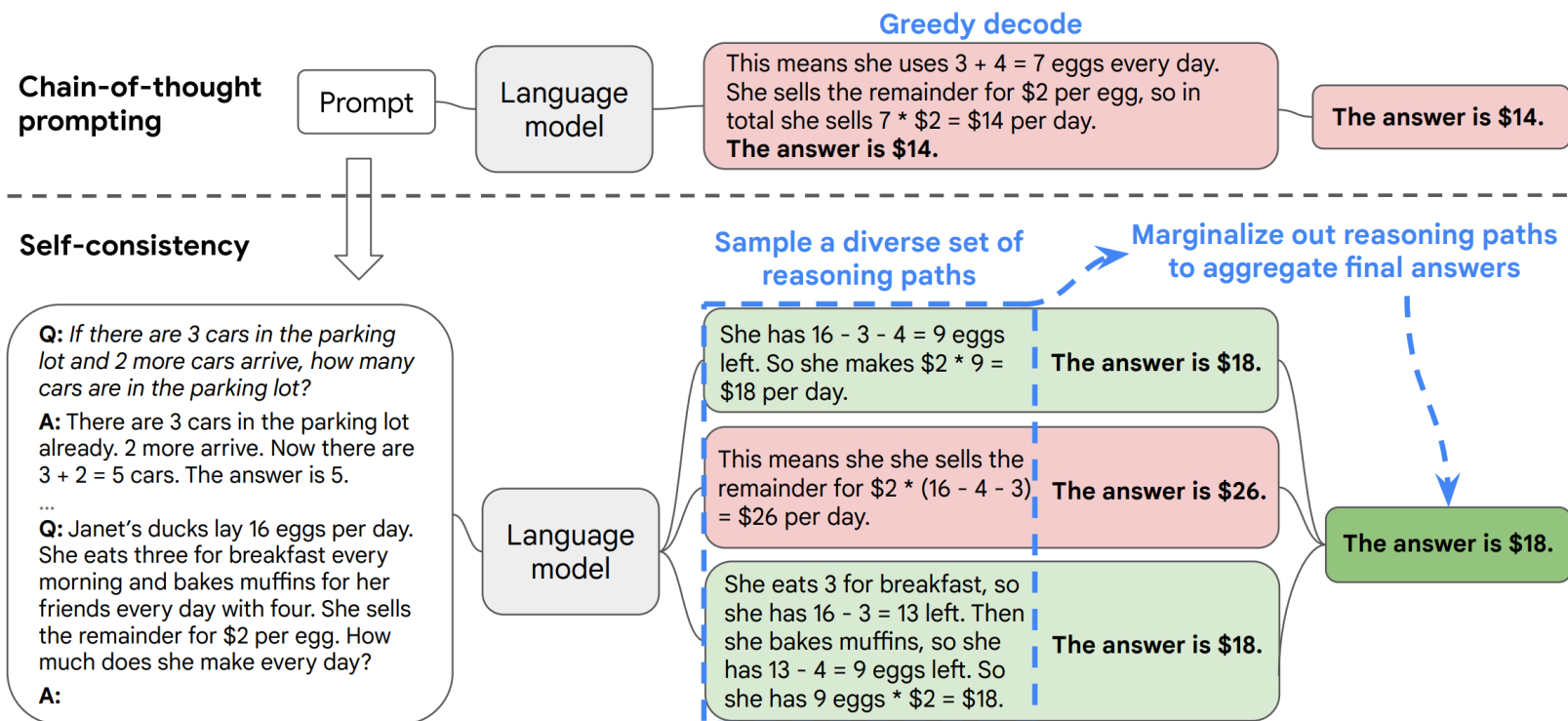


Figure 8: Using chain-of-thought prompting facilitates generalization to longer sequences in two symbolic reasoning tasks.

CoT-SC – 自洽性思维链

- 将思维链拆分成推理路径和答案单元
- 针对每个答案单元，计算它们在所有推理路径之后的生成概率
- 选择平均生成概率最高的那个，认为是与不同的推理路径一致性最高的答案

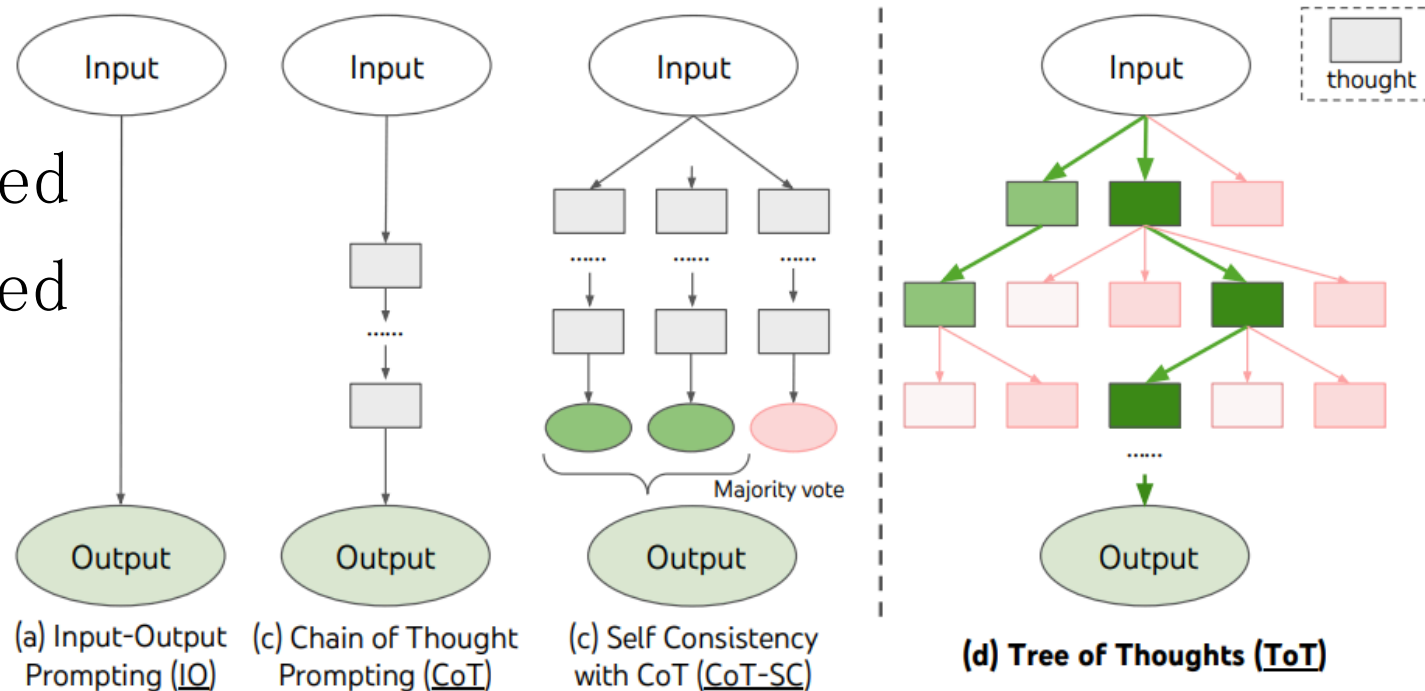
- 扩大了思维链的搜索空间，从而扩大了答案的候选集
- 避免陷入局部误区



Tree of Thoughts – 树状思维链推理下的任务规划

- 推理路径是一个寻址过程，中间步骤是一个thought，即 $z[i]$
- 从完成路径生成（CoT-SC），解耦成中间步骤探索（ToT）

- 1) 步骤解耦：任务设定
- 2) 步骤生成：Prompt-based
- 3) 状态评估：Prompt-based
- 4) 状态搜索：BFS/DFS

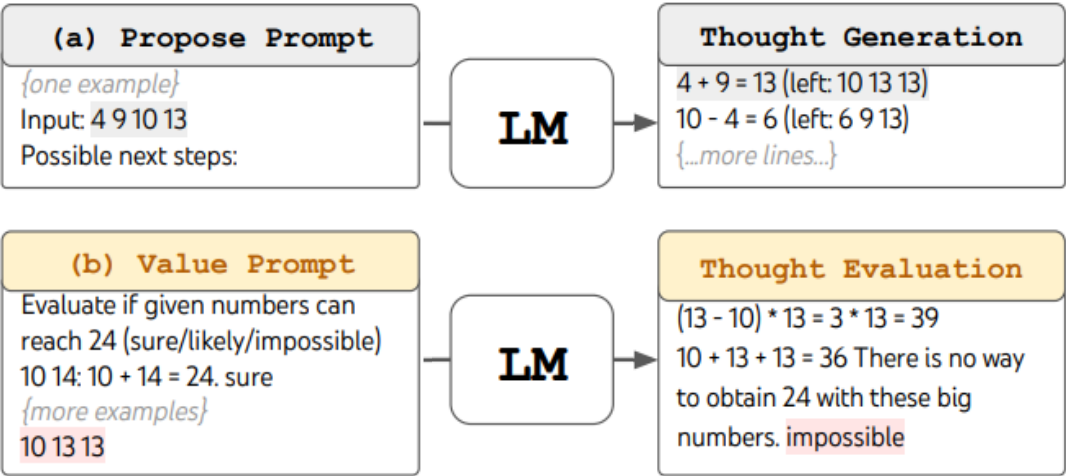
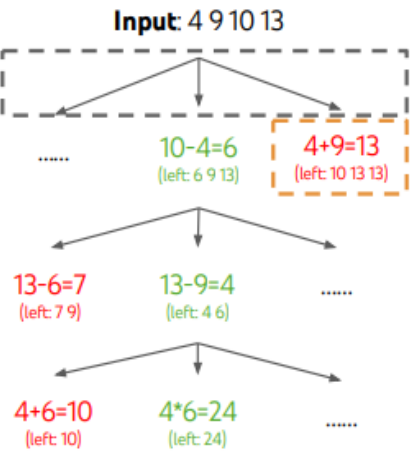


大模型思维链的标准化设定

- 大模型: θ ; 输入序列: $x[i]$; 输出序列: $y[i]$; 推理序列: $z[i]$ 表示一个推理步骤/单元
- 标准提示学习 $P_{\theta}^{Prompt}(y|x)$.
- 小样本思维链 $P_{\theta}^{CoT}(y|x, z_{1...n})$.
- 自洽思维链

$$[z_{1...n}^i, y^i] \sim P_{\theta}^{CoT}(y, z_{1...n}|x) \ (i = 1 \dots k); \underset{y}{\operatorname{argmax}} \#\{i|y^i = y\}.$$
- 树状思维链 $P_{\theta}^{CoT}(z_{i+1}|x, z_{1...i}) \ (i = 1 \dots k);..$

Tree of Thoughts –实验与结果：“24点”游戏



Method	Success
IO prompt	7.3%
CoT prompt	4.0%
CoT-SC (k=100)	9.0%
ToT (ours) (b=1)	45%
ToT (ours) (b=5)	74%
IO + Refine (k=10)	27%
IO (best of 100)	33%
CoT (best of 100)	49%

- 1) 步骤解耦：每次一个公式操作，比如(4, 9, 10, 13) → 4+9 = 13 (10, 13, 13)
- 2) 步骤生成：Prompt-based，让语言模型生成可能的下一步
- 3) 状态评估：Prompt-based，让语言模型判断该状态是否可行
- 4) 状态搜索：BFS/DFS，使用搜索算法进行路径寻址

CoT Hub – 评估大模型复杂推理能力

- 根据GPT与PaLM系列的发展与性能，观察到复杂推理能力强弱可能是区分较强与较弱模型的重要标志
- 综合了多个测试大模型的复杂推理能力的评测基准，以改进现有的评估体系
- 使用few-shot的方式进行评估，只考虑最终答案

数据集	描述
GSM8k	数学题目：8k 问题，推理步骤
MATH	数学问题：12k问题，7个种类
MMLU	知识推理：15k问题，57个类别
BigBench Hard	符号推理：6.5k问题，23个类别
HumanEval	代码问题：164个python题
C-Eval	多选题：中文，13k题目，52科目

系列	模型
OpenAI GPT	GPT-4, GPT-3.5-Turbo, text-davinci-003, text-davinci-002, code-davinci-002
Anthropic Claude	claude-v1.3, claude-instant-v1.0
Google PaLM	PaLM, PaLM-2, Flan-PaLM, Flan-U-PaLM
Meta LLaMA	7B, 13B, 33B, 65B
Google FlanT5	3B, 11B

CoT Hub – 评估大模型复杂推理能力

- 仅Claude与PaLM与GPT-4可比，开源模型更加落后。LLaMA 65B是开源模型的上限
- Top6模型中除PaLM-2外均采用RLHF，有力表明RLHF的有效性
- LLaMA-65B与code-davinci-002性能非常接近，如果进一步加入RLHF有可能接近GPT-3.5-turbo

Model	#Params	Type	GSM8k	MATH	MMLU	BBH	HumanEval	C-Eval
GPT-4	?	RLHF	92.0	42.5	86.4	-	67.0	68.7*
claude-v1.3	?	RLHF	81.8*	-	74.8*	67.3*	-	54.2*
PaLM-2	?	Base	80.7	34.3	78.3	78.1	-	-
gpt-3.5-turbo	?	RLHF	74.9*	-	67.3*	70.1*	48.1	54.4*
claude-instant-v1.0	?	RLHF	70.8*	-	-	66.9*	-	54.9*
text-davinci-003	?	RLHF	-	-	64.6	70.7	-	-
code-davinci-002	?	Base	66.6	19.1	64.5	73.7	47.0	-
Minerva	540B	SIFT	58.8	33.6	-	-	-	-
Flan-PaLM	540B	SIFT	-	-	70.9	66.3	-	-
Flan-U-PaLM	540B	SIFT	-	-	69.8	64.9	-	-
PaLM	540B	Base	56.9	8.8	62.9	62.0	26.2	-
text-davinci-002	?	SIFT	55.4	-	60.0	67.2	-	-
PaLM	64B	Base	52.4	4.4	49.0	42.3	-	-
LLaMA	65B	Base	50.9	10.6	63.4	-	23.7	38.8*
LLaMA	33B	Base	35.6	7.1	57.8	-	21.7	-
LLaMA	13B	Base	17.8	3.9	46.9	-	15.8	-
Flan-T5	11B	SIFT	16.1*	-	48.6	41.4	-	-
LLaMA	7B	Base	11.0	2.9	35.1	-	10.5	-
Flan-T5	3B	SIFT	13.5*	-	45.5	35.2	-	-

小结 – 大模型时代的复杂推理方法

- 大模型一站式解决 → 推理过程拆解 → 能力模块化
- 经典方法论与前沿模型框架的结合
 - ✓ 更复杂场景的探索
 - ✓ 能力的拆解和细分
 - ✓ 连续模型（LLM）和符号方法的结合
- 过度依赖大模型的单体能力（提示学习）

当语言模型碰到规划问题

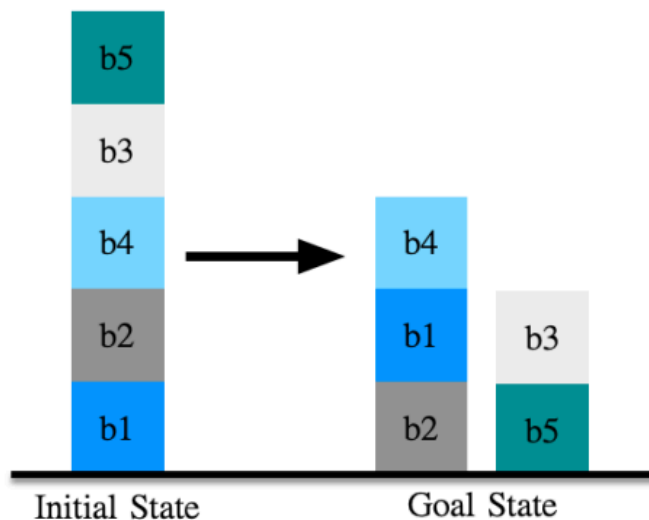
- 大语言模型在长程规划问题上，表现欠佳。

A Failure Example of GPT-4 in Planning

Problem (P1): You have 5 blocks. One cannot place more than one block on another block. b5 is on top of b3. b4 is on top of b2. b2 is on top of b1. b3 is on top of b4. b1 is on the table. b5 is clear. Your arm is empty. Your goal is to move the blocks. b1 should be on top of b2. b3 should be on top of b5. b4 should be on top of b1.

GPT-4:

1. Move b5 from b3 to the table.
2. Move b3 from b4 to b5.
3. Move b4 from b2 to the table.
4. Move b2 from b1 to b5.
5. Move b1 from the table to b2.
6. Move b4 from the table to b1.



Zero-shot Planners – 任务规划

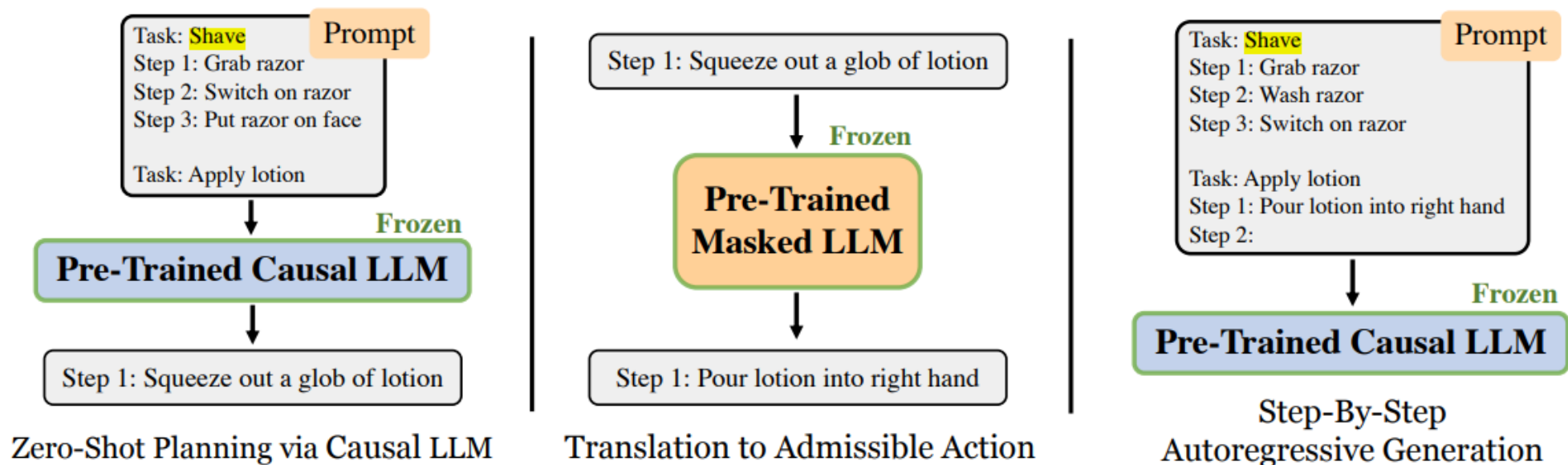
- 为保证大模型生成动作在具身智能场景下可执行，需要将生成动作翻译修正为可执行动作

Human	Vanilla GPT-3 175B
Task: Go to sleep	Task: Go to sleep
Step 1: Walk to bedroom	Step 1: Walk to bedroom
Step 2: Walk to bed	Step 2: Walk to bed
Step 3: Find bed	Step 3: Find pajamas
Step 4: Turn to bed	Step 4: Put on pajamas
Step 5: Look at bed	Step 5: Find slippers
Step 6: Sit on bed	Step 6: Put on slippers
Step 7: Sleep	Step 7: Get in bed
	Step 8: Sleep

环境中没有睡衣与拖鞋！

Zero-shot Planners – 基于大模型的任务规划

- 从已构建的范例库中选择与当前任务最相关的范例（将当前任务与范例分别通过基于bert的翻译模型得到嵌入计算余弦相似度），构建one-shot prompt
- 循环执行如下步骤：
 - 让大模型根据prompt生成动作
 - 利用翻译模型，从所有可执行动作中找到与生成动作最相关的动作并执行
 - 若任务完成，则结束循环，否则将翻译后动作添加到当前prompt中，进入下一循环



Distilling Knowledge – 从大模型中蒸馏规划能力

- 大模型在有约束的目标规划任务中表现不佳，往往不能满足约束条件。
- 通过over-generate-then-filter框架以提升大模型的有约束规划能力，并据此构造约束规划数据集，从而蒸馏出大模型的规划能力用于提升小模型的约束规划能力

目标：为糖尿病患者制作一个蛋糕

方案（text-davinci-002）：

1. 收集你的原料。
2. 将烤箱预热至350° F。在一个8英寸（20厘米）的蛋糕锅上涂上黄油或人造黄油。
3. 在一个中等大小的碗中，将面粉、发酵粉和盐混合。
4. 在一个大碗中，将奶油黄油或人造黄油和糖放在一起，直到变轻变蓬松。打入鸡蛋，一次一个。
5.

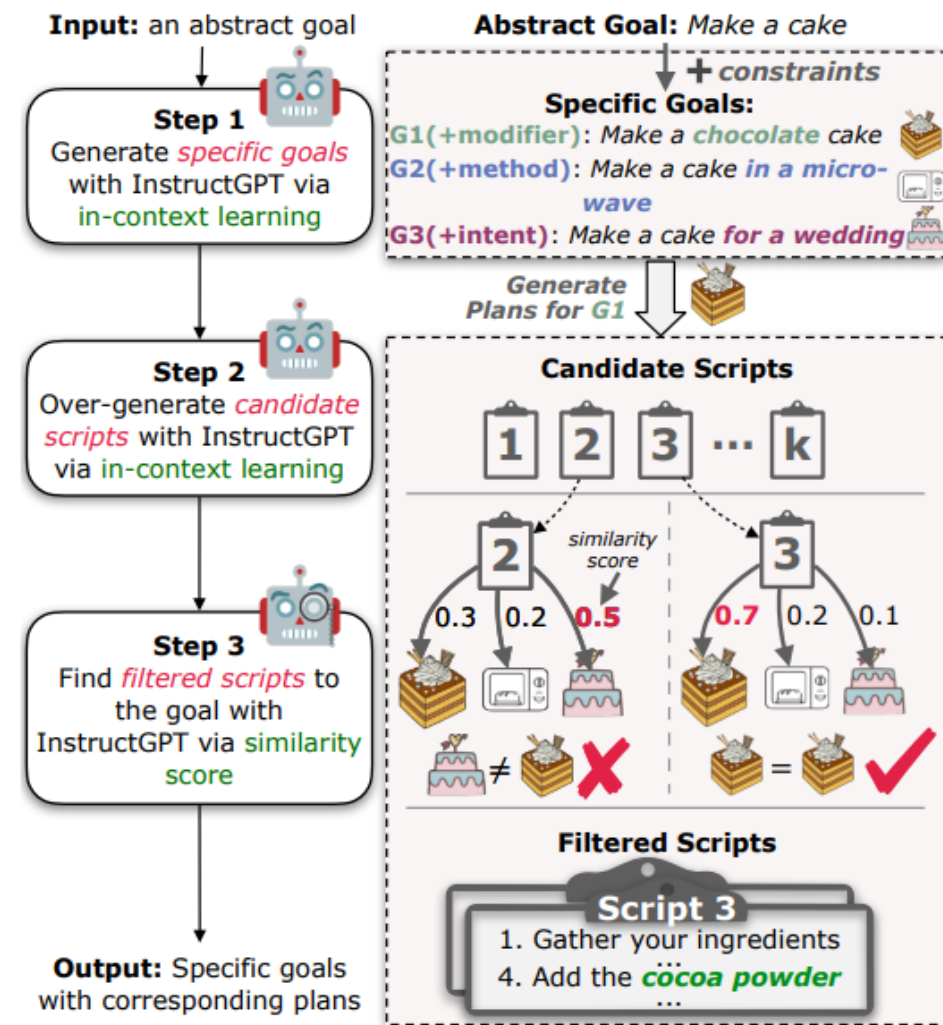
Distilling Knowledge – 从大模型中蒸馏规划能力

资料整理
火龙果科技



over-generate-then-filter框架

- 利用大模型，根据简单目标（制作一个蛋糕）生成有约束目标（制作一个巧克力蛋糕）
- 利用大模型对有约束目标生成多个候选方案
- 将生成的多个候选方案分别与有约束目标计算相似度，选择得分最高的作为最终方案



Distilling Knowledge – 从大模型中蒸馏规划能力

- 由于大模型成本高昂，因此希望使小模型也具有较强的目标规划能力
- 利用上述框架构造有约束规划数据集CoScript，从而蒸馏提取大模型的有约束目标规划能力，用于训练小模型以提升目标规划能力

实验结果：

- 在CoScript数据集上训练的小模型T5 (3B) 比175B的GPT-3、Codex约束目标规划能力更强
- 加上检索器（检索获取相关范例以增强生成）的T5 (3B) 比多数大模型的有约束规划能力更强

Model	Size	Modifier	Method	Intent	All
GPT-3	175B	30.00	22.22	25.00	25.00
Codex	175B	46.67	55.56	18.75	47.00
InstructGPT	175B	73.33	74.08	42.86	69.00
T5 (wikiHow)	3B	20.00	12.96	6.25	14.00
T5 (CoScript)	3B	63.33	55.55	43.75	56.00
+retrieval	3B	76.66	66.66	75.00	71.00

LLM+P – 结合工具的规划问题求解

- PDDL (Planning Domain Definition Language) 是面向规划的形式语言。
- LLM 可将自然语言翻译为形式化语言
- 形式化语言描述的问题可利用工具直接求解

Problem: You have 5 blocks. b2 is on top of b5. b5 is on top of b1. b1 is on top of b4. b3 is on top of b2. b4 is on the table. b3 is clear. Your arm is empty. Your goal is to move the blocks. b4 should be on top of b3.

LLM

Problem PDDL Example

```
(define (problem BW-rand-5)
  (:domain blocksworld-4ops)
  (:objects b1 b2 b3 b4 b5 )
  (:init
    (arm-empty)
    (on b1 b2)
    (on b2 b3)
    (on b3 b5)
    (on b4 b1)
    (on-table b5)
    (clear b4)
  )
  (:goal
    (and
      (on b1 b3)
      (on b3 b5))
    )
  )
)
```

LLM+P – 结合工具的规划问题求解

- 规划域定义语言（PDDL）是一种用于定义经典规划问题的形式化语言，基于该语言描述的问题可用已有工具求解
- PDDL包含两个文件：域文件与问题文件
 - 域文件描述世界基本规则，定义状态空间、动作空间与转移函数
 - 问题文件提供对象列表、初始状态与规划目标

Problem PDDL Example

```
(define (problem BW-rand-5)
  (:domain blocksworld-4ops)
  (:objects b1 b2 b3 b4 b5 )
  (:init
    (arm-empty)
    (on b1 b2)
    (on b2 b3)
    (on b3 b5)
    (on b4 b1)
    (on-table b5)
    (clear b4)
  )
  (:goal
    (and
      (on b1 b3)
      (on b3 b5))
  )
)
```

Domain PDDL Example

```
(define (domain blocksworld-4ops)
  (:requirements :strips)
  (:predicates (clear ?x)
    (on-table ?x)
    (arm-empty)
    (holding ?x)
    (on ?x ?y))

  (:action pickup
    :parameters (?ob)
    :precondition (and (clear ?ob) (on-table ?ob) (arm-empty))
    :effect (and (holding ?ob) (not (clear ?ob))
      (not (on-table ?ob)) (not (arm-empty))))

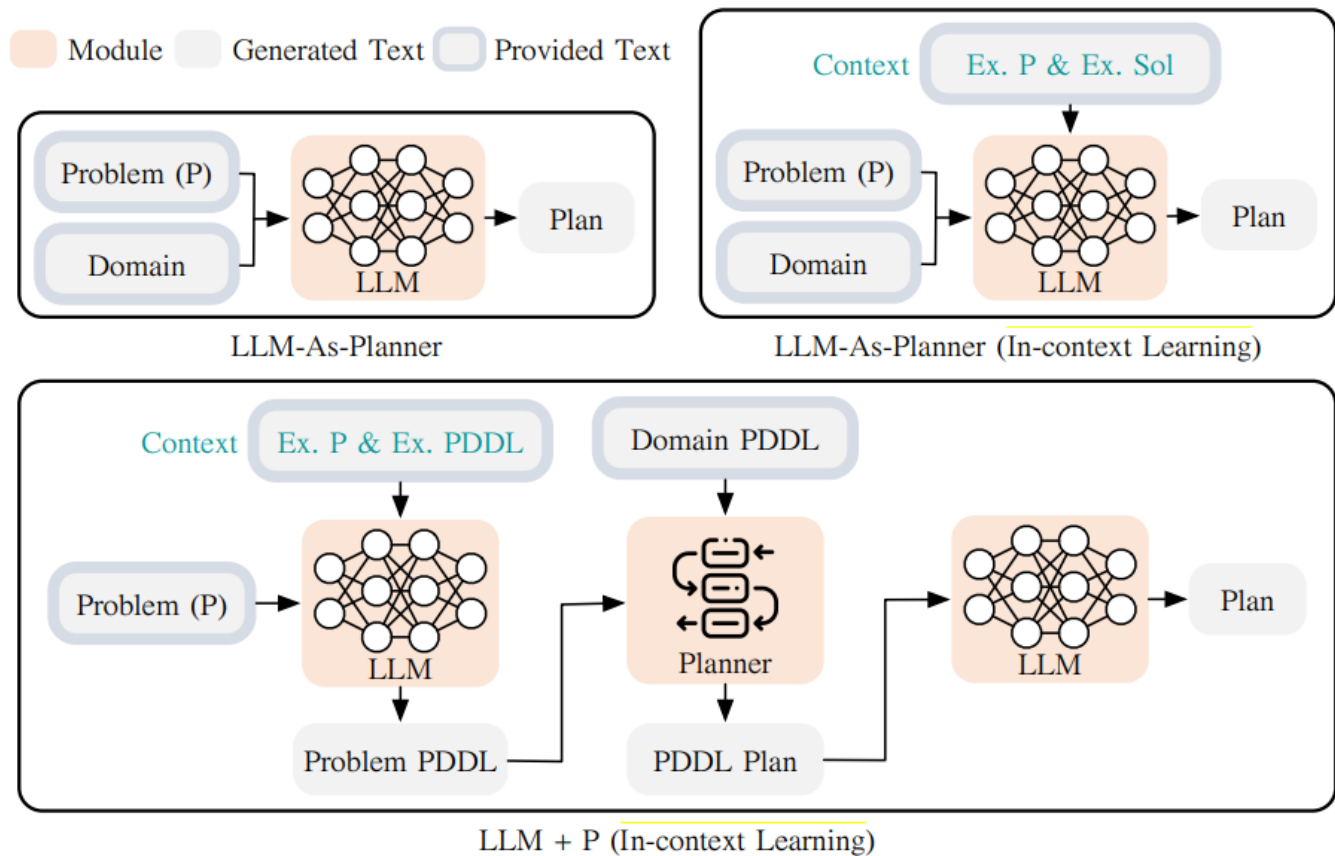
  (:action putdown
    :parameters (?ob)
    :precondition (holding ?ob)
    :effect (and (clear ?ob) (arm-empty) (on-table ?ob)
      (not (holding ?ob))))

  (:action stack
    :parameters (?ob ?underob)
    :precondition (and (clear ?underob) (holding ?ob))
    :effect (and (arm-empty) (clear ?ob) (on ?ob ?underob)
      (not (clear ?underob)) (not (holding ?ob))))

  (:action unstack
    :parameters (?ob ?underob)
    :precondition (and (on ?ob ?underob) (clear ?ob) (arm-empty))
    :effect (and (holding ?ob) (clear ?underob)
      (not (on ?ob ?underob)) (not (clear ?ob))
      (not (arm-empty)))))
```

LLM+P – 结合工具的规划问题求解

- 具体而言，PDDL是一种形式化语言，基于此描述的问题可用已有工具求解，本文以PDDL为例进行了探索
- 采用one-shot提示，将问题翻译为PDDL格式
- 输入求解器得到PDDL格式的规划结果
- 将规划结果翻译为自然语言完成求解



LLM+P – 结合工具的规划问题求解

- LLM直接生成规划（LLM-AS-P）的做法效果不佳，其给出的答案往往违反规则或无法跟踪历史行动

任务	描述
BLOCKSWORLD	在遵守物理规律的条件下重新排列一堆积木以满足要求
BARMAN	用尽可能少的步骤利用可用的原料为顾客制作鸡尾酒
FLOORTILE	操纵与移动机器人使用不同的颜色在地砖上绘制指定图案
GRIPPERS	操纵机器人在不同房间之间移动物体的以满足要求
STORAGE	操纵与移动起重机将不同区域板条箱吊入仓库
TERMES	机器人的通过搬运和放置积木来建造复杂的结构
TYREWORLD	操纵机器人按照工序更换爆胎

Domain	Success Rate %			
	LLM-AS-P (no ctx.)	LLM-AS-P	LLM+P (no ctx.)	LLM+P
BARMAN	0	0	0	85.0±0.0
BLOCKSWORLD	0	0	0	56.7±2.9
FLOORTILE	0	0	0	53.3±2.9
GRIPPERS	10.0	15.0	0	95.0±0.0
STORAGE	0	0	0	65.0±0.0
TERMES	0	0	0	31.7±2.9
TYREWORLD	0	40.0±5.0	0	80.0±0.0

小结 – 文本规划问题

- 文本规划是联结智能体和物理智能体的桥梁
- 单纯依赖大模型的能力，在面对特定问题时不一定都有效
 - ✓ 蒸馏大模型的能力，用以训练小模型
 - ✓ 引入外部工具，协同完成复杂任务

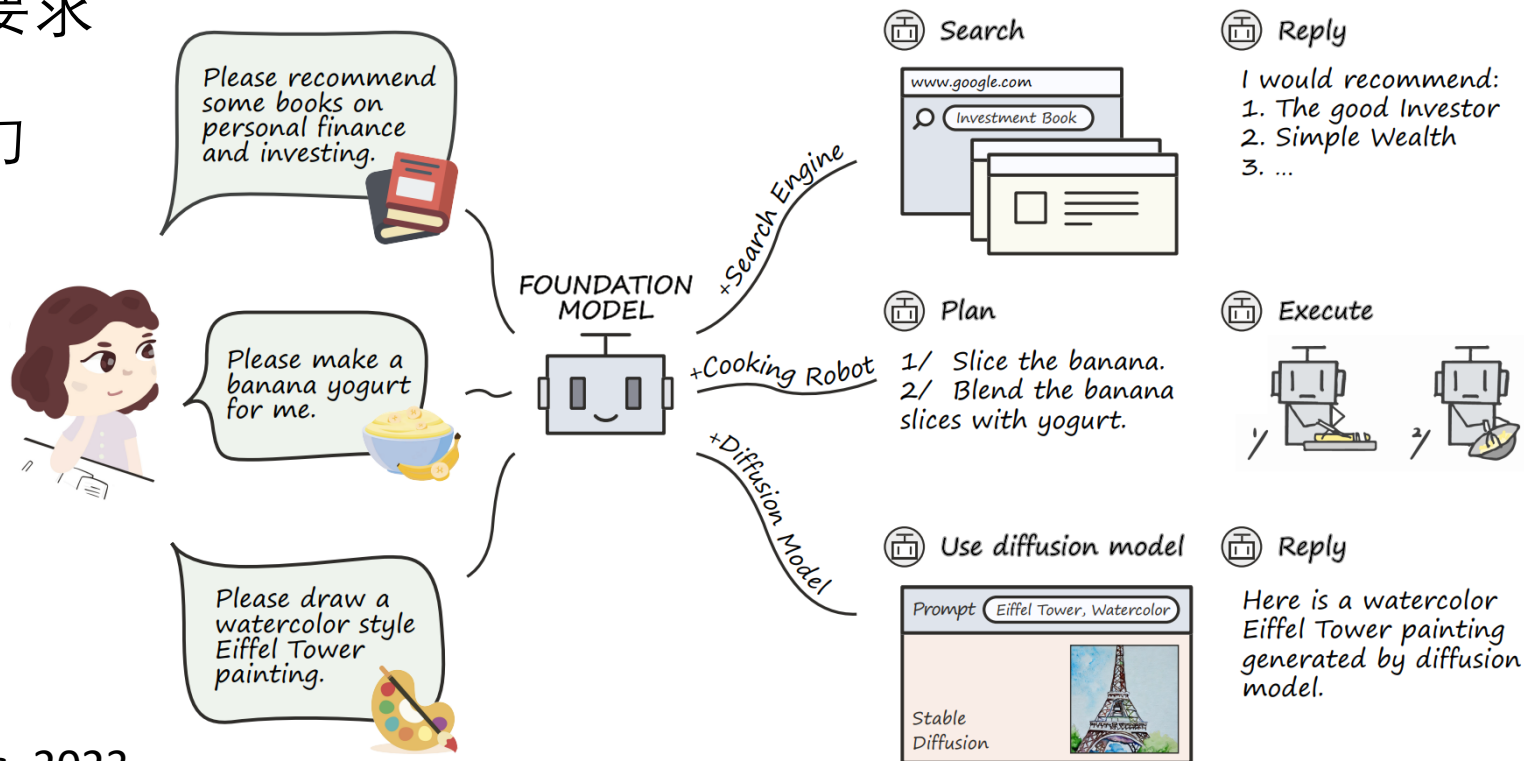
目录

- 大语言模型驱动单体智能
 - 大语言模型的复杂任务能力
 - 大语言模型的单体能力拓展
 - 大语言模型的能力自我提升
- 大语言模型驱动群体智能
 - 智能体的记忆能力
 - 多智能体的社区模拟
 - 多智能体的任务协作

使用外部工具的大语言模型

- 让大规模语言模型学会如何去更好的使用专业工具，从而增强模型在专业领域的应用能力，解决复杂的实际问题。

- 降低大模型的记忆要求
- 适配专业领域的的能力
- 更好的可解释性
- 提升模型鲁棒性
- 提升模型实时性



Tool Learning with Foundation Models, 2023

Figure 1: Tool learning paradigm aims to combine the strengths of specialized tools and foundation models.

怎样让大模型使用工具？

- 通过提示让模型学习使用工具
 - 零样本提示：描述工具使用方法
 - 小样本提示：提供工具使用样例
- 规划工具的使用
 - 内部推理：单轮工具使用决策
 - 外部推理：结合工具调用结果，实现多轮工具使用决策
- 面向工具学习微调
 - 演示样本：构建或标注工具调用样本数据
 - 反馈样本：从尝试与错误中学习
- 统一接口可以实现工具泛化
 - 可引入元工具、课程学习等技巧

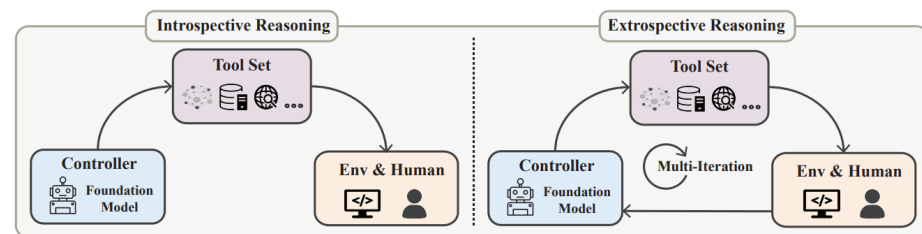


Figure 5: Illustration of introspective reasoning and extrospective reasoning. Extrospective reasoning requires feedback from the environment and humans to carry out iterative plan generation. We omit the perceiver in the illustration for simplicity.

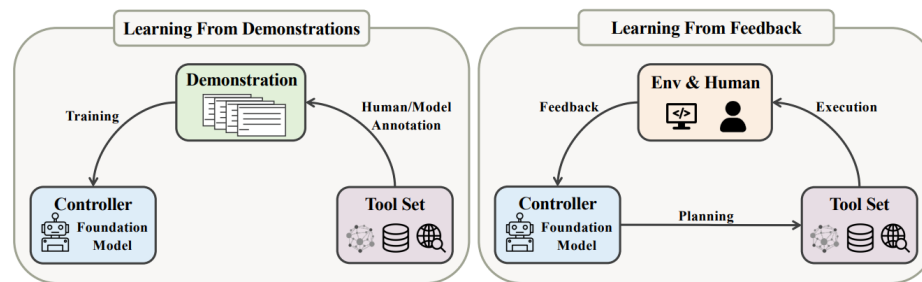


Figure 6: Training strategies for tool learning: (left) learning from human-annotated or model-annotated demonstrations; (right) learning from feedback, where the supervision could come from either the environment or humans.

ReAct – 混合语言和动作执行

- 语言和动作执行依赖不同的系统，即，语言模型和外部工具

- Thought: 内部推理
- Act: 外部动作

(1) Hotspot QA

Question: Aside from the Apple Remote, what other device can control the program Apple Remote was originally designed to interact with?

(1a) Standard

Answer: iPod



(1b) CoT (Reason Only)

Thought: Let's think step by step. Apple Remote was originally designed to interact with Apple TV. Apple TV can be controlled by iPhone, iPad, and iPod Touch. So the answer is iPhone, iPad, and iPod Touch.

Answer: iPhone, iPad, iPod Touch



(1c) Act-Only

Act 1: Search[Apple Remote]

Obs 1: The Apple Remote is a remote control ...

Act 2: Search[Front Row]

Obs 2: Could not find [Front Row]. Similar: ...

Act 3: Search[Front Row (software)]

Obs 3: Front Row is a discontinued media center software ...

Act 4: Finish[yes]



ReAct – 推理与行为协同下的规划与决策

- 拓展操作空间，包含了语言和操作
- Thought: 语言模型
- Act: 外部工具
- Obs: 环境反馈
- 智能体与环境进行多轮交互，交替进行内部推理和动作执行。

(1d) ReAct (Reason + Act)

Thought 1: I need to search Apple Remote and find the program it was originally designed to interact with.

Act 1: `Search[Apple Remote]`

Obs 1: The Apple Remote is a remote control introduced in October 2005 by Apple ... originally designed to control the Front Row media center program ...

Thought 2: Apple Remote was originally designed to control the Front Row media center program. I need to search Front Row next and find what other device can control it.

Act 2: `Search[Front Row]`

Obs 2: Could not find [Front Row]. Similar: ['Front Row Seat to Earth', 'Front Row Motorsports', 'Front Row (software)', ...]

Thought 3: Front Row is not found. I need to search Front Row (software) .

Act 3: `Search[Front Row (software)]`

Obs 3: Front Row is a discontinued media center software ...

Thought 4: Front Row (software) is controlled by an Apple Remote or the keyboard function keys. So the answer is keyboard function keys.

Act 4: `Finish[keyboard function keys]`



ReAct – 热点问答问题prompt构造

ReAct伪代码:

设置最大步数为max_step_num

将**Instruction**与**Examples**拼接作为初始化**Prompt**

对于每一个时间步执行以下步骤:

将**Prompt**输入LLM, 返回**Thought**和**Action**

执行**Action**, 返回**Observation**并判断是否完成任务

若任务完成或当前时间步大于最大步数:

退出循环

否则:

将**Thought**、**Action**和**Observation**添加到**Prompt**中,
进入下一步循环

```
thought_action = LLM(  
    prompt + f"Thought {i}:",  
    stop=[f"\nObservation {i}:"]  
)  
thought, action = thought_action.strip().split(f"\nAction {i}: ")
```

Instruction: 通过交替的思考、行动、观察来解决问题回答任务。

思考可以推理当前的情况, 行动可以有三种类型:

(1) Search[entity], 它在维基百科上搜索确切的实体, 如果存在, 则返回第一段。如果没有, 它将返回一些类似的实体进行搜索。

(2) Lookup[keyword], 返回当前文章中包含关键字的下一个句子。

(3) Finish[answer], 返回答案并完成任务。下面是一些例子:

Example:

Question: Musician and satirist Allie Goertz wrote a song about the "The Simpsons" character Milhouse, who Matt Groening named after who?

Thought 1: The question simplifies to "The Simpsons" character Milhouse is named after who. I only need to search Milhouse and find who it is named after.

Action 1: Search[Milhouse]

Observation 1: Milhouse Mussolini Van Houten is a recurring character in the Fox animated television series The Simpsons voiced by Pamela Hayden and created by Matt Groening.

...

Thought 3: Milhouse was named after U.S. president Richard Nixon, so the answer is Richard Nixon.

Action 3: Finish[Richard Nixon]

API-Bank: 评测大模型的工具使用水平

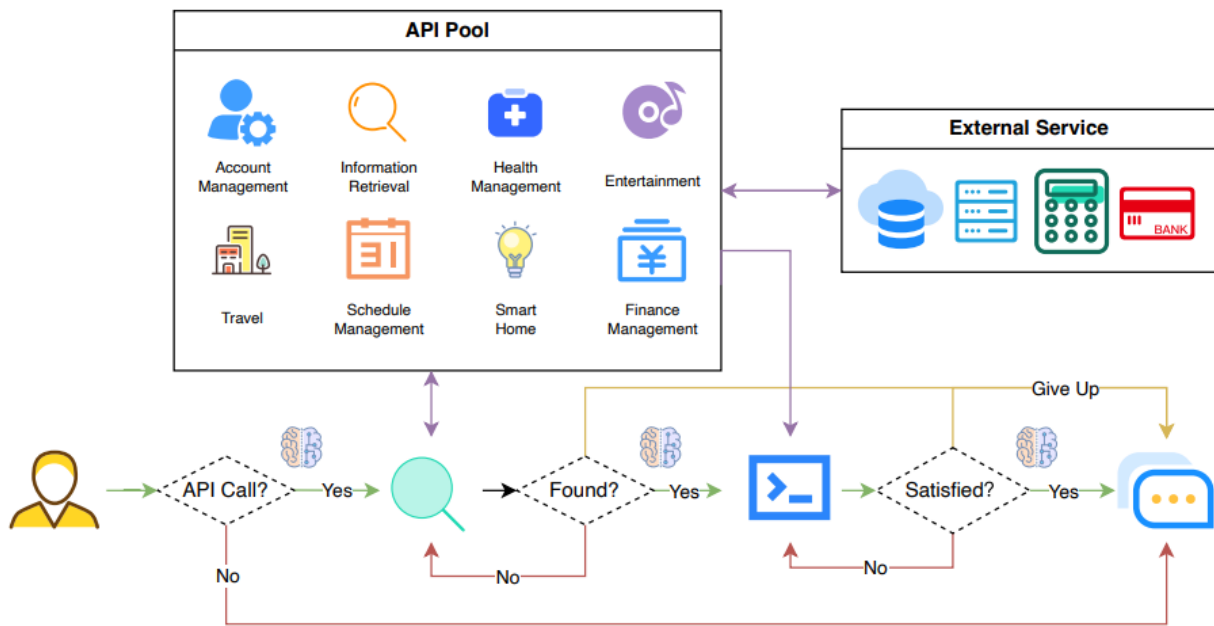


Figure 1: The proposed Tool-Augmented LLMs paradigm.

- 在in-context learning机制下完成任务
 - 决定是否调用 API
 - 若是，搜索可用的 API
 - 提取用户需求关键字
 - 根据关键字搜索 API
 - 若找到，调用并获得结果
 - 生成 API 调用命令
 - 发送命令，获取执行结果
 - 若结果合理，结合结果生成反馈
 - 若上述流程失败，直接生成反馈

API-Bank: 评测大模型的工具使用水平

Model	level-1	level-2
GPT-3.5		
- API correctness	54.64%	40.74%
- dialog after API	0.4622	0.3445
Davinci		
- API correctness	0.50%	1.48%
- dialog after API	0.1035	0.091

Table 3: The result of several baselines on API-Bank. (The metric for API correctness is Accuracy and that for dialog after API is Rouge-L)

	GPT-3.5	GPT-4	Human
Shopping	10	10	7
Reminder	13	5	4
Personal Assistant	16	7	5
Meeting Schedule	7	5	4
Hospital	8	11	6
Health Management	10	7	6
Financial Management	8	5	4
Hotel Booking	7	8	7

Table 4: The results of GPT-3.5, GPT-4, and humans on the level-3.

评测指标:

API 调用: API名字, 参数, Accuracy
调用后的对话: 文本匹配, ROUGE

评测结果

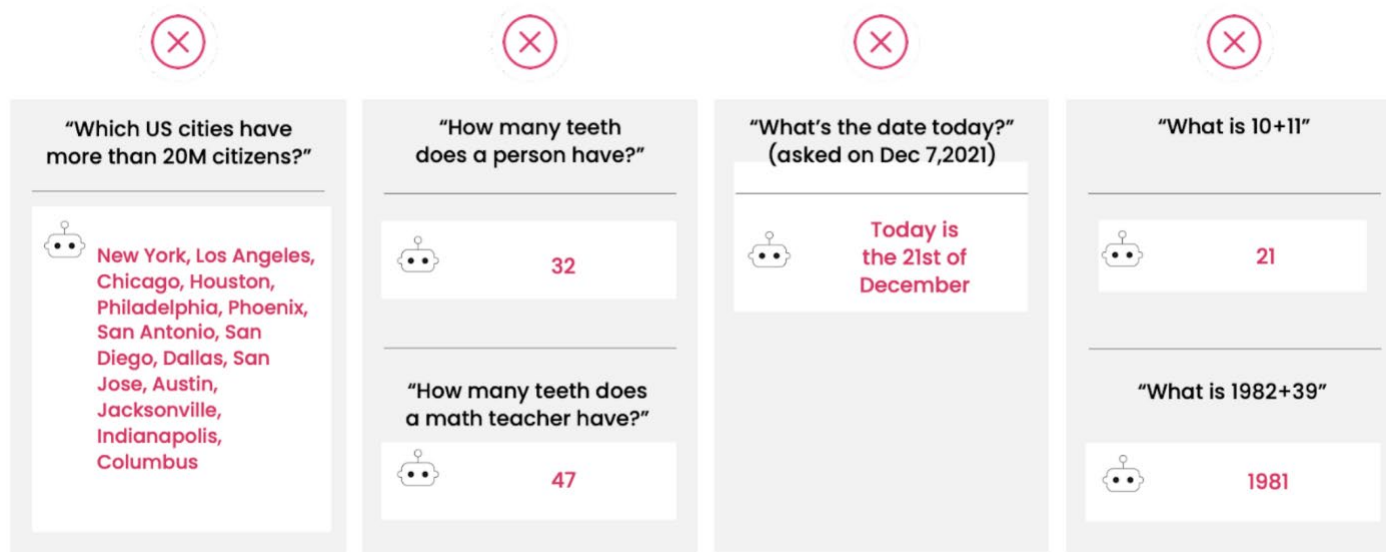
GPT-3.5 在前两级评测上优于 GPT-3

GPT-4 在三级评测上优于 GPT-3.5

API-Bank: A Benchmark for Tool-Augmented LLMs, 2023

- 一级: 使用
 - 提供待调用 API 文档和用户需求
 - 评测是否能够确定调用API
- 二级: 检索
 - 提供工具检索 API 文档和用户需求
 - 评测是否可以检索到正确API
- 三级: 规划
 - 提供场景、用户需求和关键信息
 - 评测用户得到满意结果所需的轮数

MRKL: 精调时代的探索



- 为了弥补大模型能力的不足，提出模块化推理的框架，结合连续+离散的方法。
- Jurassic-X 示例：自然语言计算器



Jurassic-X : MRKL示例, 一个自然语言计算

- 从简单文本中高质量地提取四则运算表达式
 - 模型可以有效地提取含大数的单层表达式
 - 解决 GPT-3 的大数计算缺陷问题
 - 对不同的数字形式、表达格式, 以及双层表达式均表现良好

Format	addition	subtraction
0	How much is {x} plus {y}?	How much is {x} minus {y}?
1	What is {x} plus {y}?	What is {x} minus {y}?
2	What is the result of {x} plus {y}?	What is the result of {x} minus {y}?
3	What is the sum of {x} and {y}?	What is the difference between {x} and {y}?
4	The sum of {x} and {y} is	The difference between {x} and {y} is
	multiplication	division
0	How much is {x} times {y}?	How much is {x} over {y}?
1	What is {x} times {y}?	What is {x} over {y}?
2	What is the result of {x} times {y}?	What is the result of {x} over {y}?
3	What is the product of {x} and {y}?	What is the ratio between {x} and {y}?
4	The product of {x} and {y} is	The ratio of {x} and {y} is

Table 10: **Question formats for single-operation problems.** Five different formats for each type of operation in the case of single-operation expressions.

Num. digits	1	2	3	4	5	6	7	8	9
Addition	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
Multiplication	1.0	1.0	1.0	1.0	1.0	0.98	1.0	1.0	1.0
GPT-3	N/A	1.0	0.804	0.255	0.093	N/A	N/A	N/A	N/A

Table 1: **Robustness to the number of digits in the operands.** Accuracy vs. the number of digits in the test data for a model that was trained only on 1-digit operands (5 + 3, etc.). Results for GPT-3 were taken from [2].

MRKL Systems: A modular, neuro-symbolic architecture that combines large language models, external knowledge sources and discrete reasoning, 2022

TALM: 精调时代的探索, MRKL的进阶

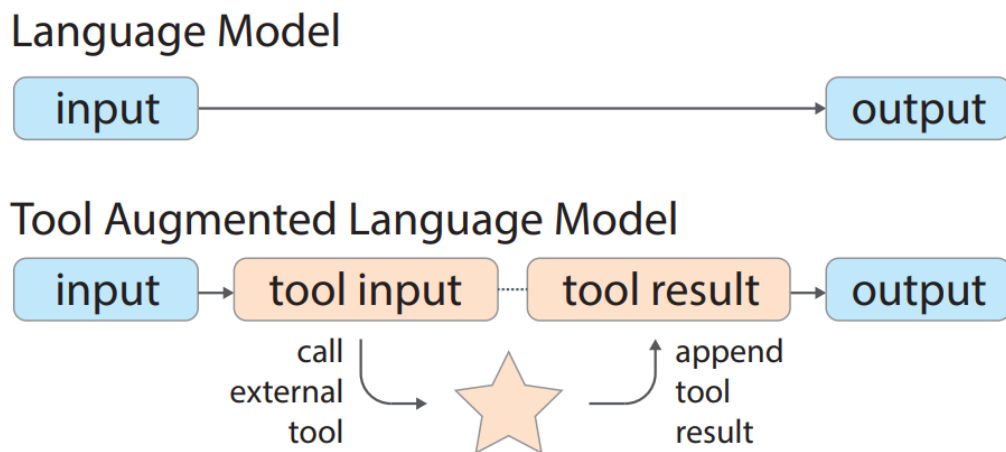


Figure 2: LM and Tool Augmented LMs.

An abstract task:

task input text | **tool-call** tool input text | **result** tool output text | **output** task output text

A weather task:

how hot will it get in NYC today? | **weather** lookup region=NYC | **result** precipitation chance: 10, high temp: 20c, low-temp: 12c | **output** today's high will be 20C

Figure 3: TALM text-to-text interface example.

- 面向不同任务，构建（“任务输入”，“工具输入”，“工具输出”，“任务输出”）四元组。
- 二步推理：第一步从input → tool input，第二步从input+tool result → output

TALM: 迭代式的工具学习样本生成

Algorithm 1 Iterative Self-Play Algorithm.

x : task input, y : task output, t : tool input, r : tool output

```

1:  $T = \{x_i, y_i\}_T$  # task set
2:  $D = \{x_j, t_j, r_j, y_j\}_D$  # tool-use set
3:  $P_\theta \leftarrow \text{pretrained LM}$ 
4: for  $t \in [0, 1, \dots, R]$  do # self-play rounds
5: # finetune LM
6:  $\theta \leftarrow \operatorname{argmax}_\theta \prod_D P_\theta(y_j|x_j, t_j, r_j)P_\theta(t_j|x_j)$ 
7: for  $x_i, y_i \in T$  do # iterate task set
8:   for  $n \in [0, 1, \dots, N]$  do
9:      $t_n \leftarrow P_\theta(t|x_i)$  # sample tool query
10:     $r_n \leftarrow \text{Tool}(t_n)$  # call tool API
11:     $y_n \leftarrow P_\theta(y|x_i, t_n, r_n)$  # get task output
12:    if  $|y_n - y_i| < th$  then # filter wrong output
13:       $D \leftarrow D \cup \{x_i, t_n, r_n, y_n\}_1$ 
14: # update tool-use set

```

TALM: Tool Augmented Language Models

- 通过迭代自学习掌握工具
 - 解决工具使用样例不足的问题
 - 借助下游任务数据学习实用工具
 - 对任务数据样本采样工具调用
 - 筛选正确的（能减小损失的）调用
 - 模型学习筛选后的调用
 - 迭代地扩充工具使用样本集
- 用扩充数据精调 T5 得到 TALM
 - 在 NQ 和 MathAQ 上测试
 - 220M TALM > 3B T5

Toolformer: 大语言模型时代的工具调用

- 提供五个工具：问答、维基搜索、计算器、日历、翻译

The New England Journal of Medicine is a registered trademark of [QA("Who is the publisher of The New England Journal of Medicine?") → Massachusetts Medical Society] the MMS.

Out of 1400 participants, 400 (or [Calculator(400 / 1400) → 0.29] 29%) passed the test.

API Name	Example Input	Example Output
Question Answering	Where was the Knights of Columbus founded?	New Haven, Connecticut
Wikipedia Search	Fishing Reel Types	Spin fishing > Spin fishing is distinguished between fly fishing and bait cast fishing by the type of rod and reel used. There are two types of reels used when spin fishing, the open faced reel and the closed faced reel.
Calculator	$27 + 4 * 2$	35
Calendar	ϵ	Today is Monday, January 30, 2023.
Machine Translation	sûreté nucléaire	nuclear safety

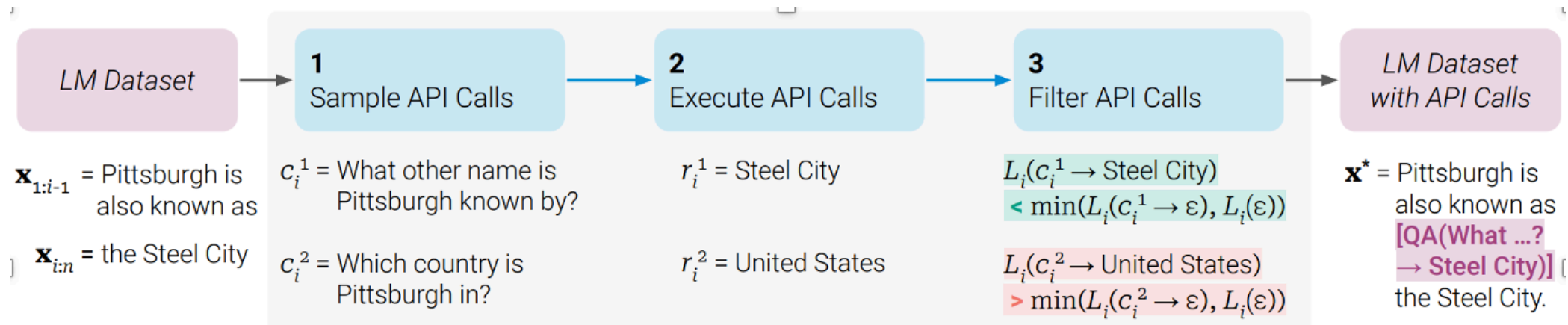
Table 1: Examples of inputs and outputs for all APIs used.

Out of 1400 participants, 400 (or `<API>Calculator(400 / 1400) → 0.29]</API>`29%) passed the test.

其中“`<API>`”、“`→`”、“`</API>`”是特殊token，分别表示工具API调用的开始、执行和结束；

“`→`”之前的字符串是工具的调用指令，包括工具名和输入的参数，“`→`”之后的是工具的调用结果。

Toolformer: 训练样本生成方法



- 将迭代自学习推广到通用语料
 - 随机采样 API 调用点与参数
 - 筛选原文损失下降的有效调用
 - 将有效调用插入调用点，获得增强精调语料

API	Number of Examples		
	$\tau_f = 0.5$	$\tau_f = 1.0$	$\tau_f = 2.0$
Question Answering	51,987	18,526	5,135
Wikipedia Search	207,241	60,974	13,944
Calculator	3,680	994	138
Calendar	61,811	20,587	3,007
Machine Translation	3,156	1,034	229

Table 2: Number of examples with API calls in $\mathcal{C}^*$ for different values of our filtering threshold τ_f .

Toolformer: 工具调用的实现方法

1. “推理模式”：正常地输出文本；

The New England Journal of Medicine is a registered trademark of ...

2. “推理模式”：生成API调用指令；

The New England Journal of Medicine is a registered trademark of **<API> QA(“Who is the publisher of The New England Journal of Medicine?”) →**

3. “调用模式”：遇到token“→”时，调用工具，插入结果；

The New England Journal of Medicine is a registered trademark of **<API> QA(“Who is the publisher of The New England Journal of Medicine?”) → Massachusetts Medical Society </API>**

4. “推理模式”：调用结束，继续解码。

The New England Journal of Medicine is a registered trademark of **<API> QA(“Who is the publisher of The New England Journal of Medicine?”) → Massachusetts Medical Society </API>** Massachusetts Medical Society

Toolformer: 工具调用的训练方法

对于一段文本: Joe Biden was born in Scranton, Pennsylvania.

1. 采样: 使用大模型对API调用位置进行采样:

Joe Biden was born in `<API> QA("Where was Joe Biden born?") </API>` Scranton, `<API> QA("In which state is Scranton?") </API>` Pennsylvania.

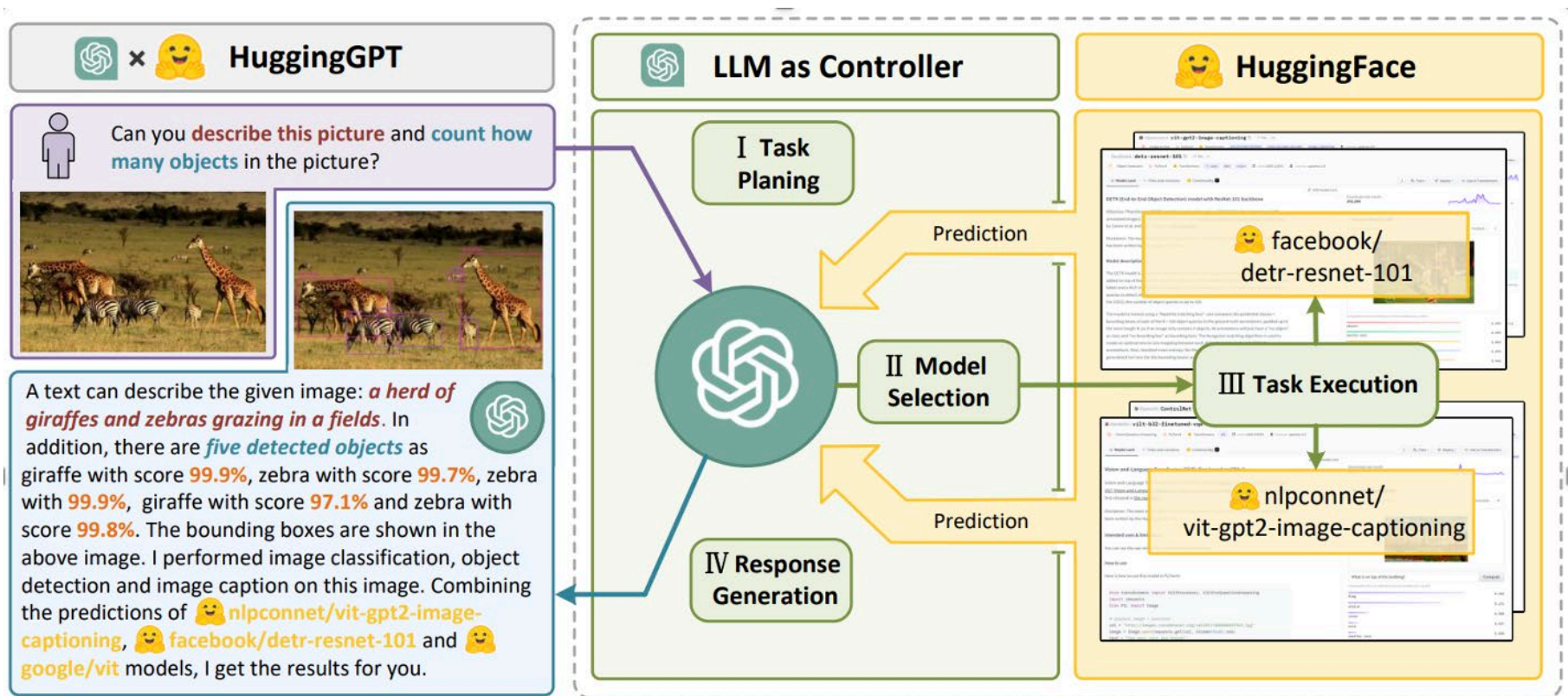
2. 调用: 调用工具, 补充调用结果;

Joe Biden was born in `<API> QA("Where was Joe Biden born?") → Scranton </API>` Scranton, `<API> QA("In which state is Scranton?") → Pennsylvania </API>` Pennsylvania.

3. 过滤: 对比带API调用和不带调用的文本, 保留loss 差值高的。

$$L_i(z) = - \sum_{j=i}^n \omega_{j-i} \log p(x_j | x_{1:j-1})$$

HuggingGPT: 社区精神



- 机器难题社区的众多模型作为工具
 - 由大模型分析任务，并挑选合适的huggingface模型
 - 由huggingface模型提供所需能力

HuggingGPT: 如何高效使用海量huggingface模型

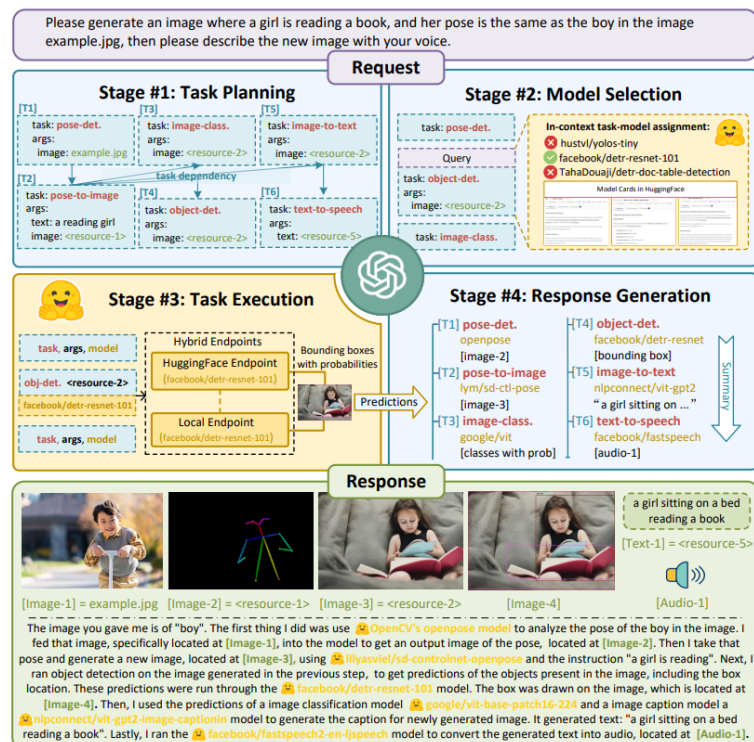


Figure 2: Overview of HuggingGPT. With an LLM (e.g., ChatGPT) as the core controller and the expert models as the executors, the workflow of HuggingGPT consists of four stages: 1) **Task planning**: LLM parses the user request into a task list and determines the execution order and resource dependencies among tasks; 2) **Model selection**: LLM assigns appropriate models to tasks based on the description of expert models on Hugging Face; 3) **Task execution**: Expert models on hybrid endpoints execute the assigned tasks; 4) **Response generation**: LLM integrates the inference results of experts and generates a summary of workflow logs to respond to the user.

- 任务规划
 - 大模型根据用户输入列出任务清单
- 模型选择
 - 大模型比较任务清单与模型的描述，选择合适的模型
- 任务执行
 - huggingface模型执行清单上的各个任务
- 结果生成
 - 大模型总结模型返回的结果

HuggingGPT：社区精神

Task Planning	Prompt
	#1 Task Planning Stage - The AI assistant performs task parsing on user input, generating a list of tasks with the following format: <code>[{"task": task, "id": task_id, "dep": dependency_task_ids, "args": {"text": text, "image": URL, "audio": URL, "video": URL}}]</code> . The "dep" field denotes the id of the previous task which generates a new resource upon which the current task relies. The tag <code><resource>-task_id</code> represents the generated text, image, audio, or video from the dependency task with the corresponding task_id. The task must be selected from the following options: <code>[Available Task List]</code> . Please note that there exists a logical connections and order between the tasks. In case the user input cannot be parsed, an empty JSON response should be provided. Here are several cases for your reference: <code>[Demonstrations]</code> . To assist with task planning, the chat history is available as <code>[Chat Logs]</code> , where you can trace the user-mentioned resources and incorporate them into the task planning stage.
	Demonstrations Can you tell me how many objects in e1.jpg? In e2.jpg, what's the animal and what's it doing? First generate a HED image of e3.jpg, then based on the HED image and a text "a girl reading a book", create a new image as a response.
Model Selection	Prompt
	#2 Model Selection Stage - Given the user request and the call command, the AI assistant helps the user to select a suitable model from a list of models to process the user request. The AI assistant merely outputs the model id of the most appropriate model. The output must be in a strict JSON format: <code>{"id": "id", "reason": "your detail reason for the choice"}</code> . We have a list of models for you to choose from <code>[Candidate Models]</code> . Please select one model from the list.
	Candidate Models <code>{ "model_id": model id #1, "metadata": meta-info #1, "description": description of model #1 }</code> <code>{ "model_id": model id #2, "metadata": meta-info #2, "description": description of model #2 }</code> ... <code>{ "model_id": model id #K, "metadata": meta-info #K, "description": description of model #K }</code>
Response Generation	Prompt
	#4 Response Generation Stage - With the input and the inference results, the AI assistant needs to describe the process and results. The previous stages can be formed as - User Input: <code>[User Input]</code> , Task Planning: <code>[Tasks]</code> , Model Selection: <code>[Model Assignment]</code> , Task Execution: <code>[Predictions]</code> . You must first answer the user's request in a straightforward manner. Then describe the task process and show your analysis and model inference results to the user in the first person. If inference results contain a file path, must tell the user the complete file path. If there is nothing in the results, please tell me you can't make it.

Table 1: The details of the prompt design in HuggingGPT. In the prompts, we set some injectable slots such as `[ Demonstrations ]` and `[ Candidate Models ]`. These slots are uniformly replaced with the corresponding text before being fed into the LLM.

- 全过程采用上下文学习
 - 利用 GPT-3.5、GPT-4 等超大模型
 - 不再需要针对 API 调用精调
- GPT 3.5 有强大任务规划能力
 - 涵盖单任务、线性、分支多任务
 - 优于 Alpaca 和 Vicuna

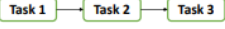
Task Type	Diagram	Example	Metrics
Single Task		Show me a funny image of a cat	Precision, Recall, F1, Accuracy
Sequential Task		Replace the cat with a dog in example.jpg	Precision, Recall, F1 Edit Distance
Graph Task		Given a collection of image A: a.jpg, B: b.jpg, C: c.jpg, please tell me which image is more like image B in terms of semantic, A or C?	Precision, Recall, F1 GPT-4 Score

Table 2: Evaluation for task planning in different task types.

LLM	Acc ↑	Pre ↑	Recall ↑	F1 ↑
Alpaca-7b	6.48	35.60	6.64	4.88
Vicuna-7b	23.86	45.51	26.51	29.44
GPT-3.5	52.62	54.45	32.26	39.05

Table 3: Evaluation for the single task. “Acc” and “Pre” represents Accuracy and Precision.

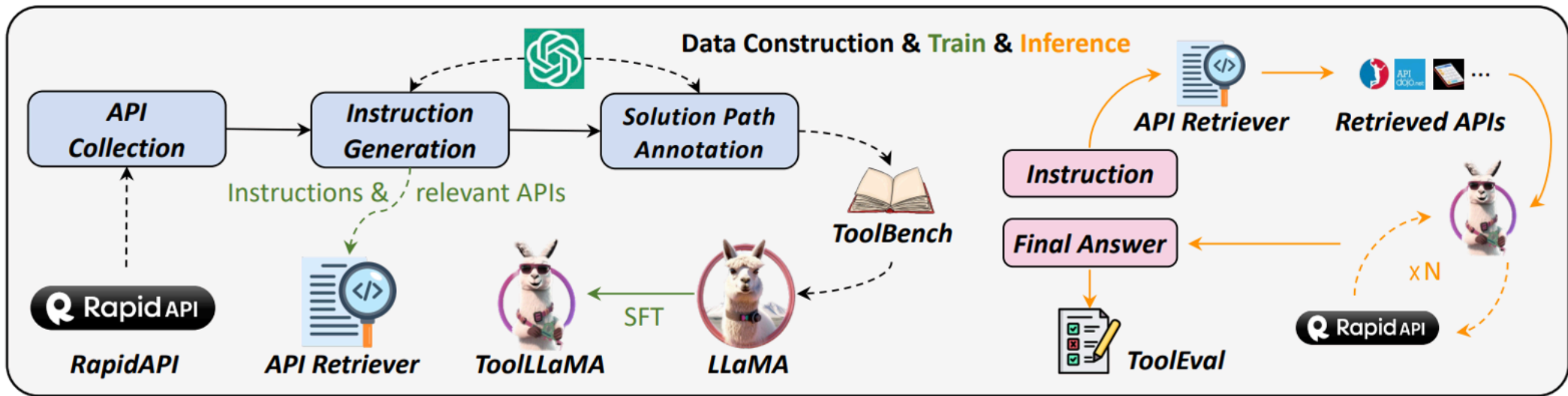
LLM	ED ↓	Pre ↑	Recall ↑	F1 ↑
Alpaca-7b	0.83	22.27	23.35	22.80
Vicuna-7b	0.80	19.15	28.45	22.89
GPT-3.5	0.54	61.09	45.15	51.92

Table 4: Evaluation for the sequential task. “ED” means Edit Distance.

LLM	GPT-4 Score ↑	Pre ↑	Recall ↑	F1 ↑
Alpaca-7b	13.14	16.18	28.33	20.59
Vicuna-7b	19.17	18.66	13.97	28.08
GPT-3.5	50.48	54.90	49.23	51.91

Table 5: Evaluation for the graph task.

ToolLLM: 开源通用工具学习框架



ToolLLaMA是在**LLaMA-2-7b**上微调得到的模型

基于RapidAPI平台上**16000多种**不同的工具，允许模型**多次调用不同工具**

基于多轮对话数据进行指令微调训练

系列工具：ToolBench，API Retriever，ToolLLaMA，ToolEval

TOOLLLM: Facilitating Large Language Modles to Master 16000+ Real-World APIs, 2023

ToolLLM: 多轮对话数据格式

"user": "我想给我的朋友一个生日惊喜。我知道她最喜欢的女演员是海莉·斯坦菲尔德。"

"assistant": **"Thought:** 我需要首先获取海莉·斯坦菲尔德的信息。

API Name: get_extra_character_details

Arguments: {"name": "Hailee Steinfeld"}"

"function": "{"age": 28, "recent movies": ["Spider-Man: Across the Spider-Verse" ...]}"

"assistant": **"Thought:** 我需要..."

"function": ...

"assistant": **"Thought:** 我可以进行回答了。

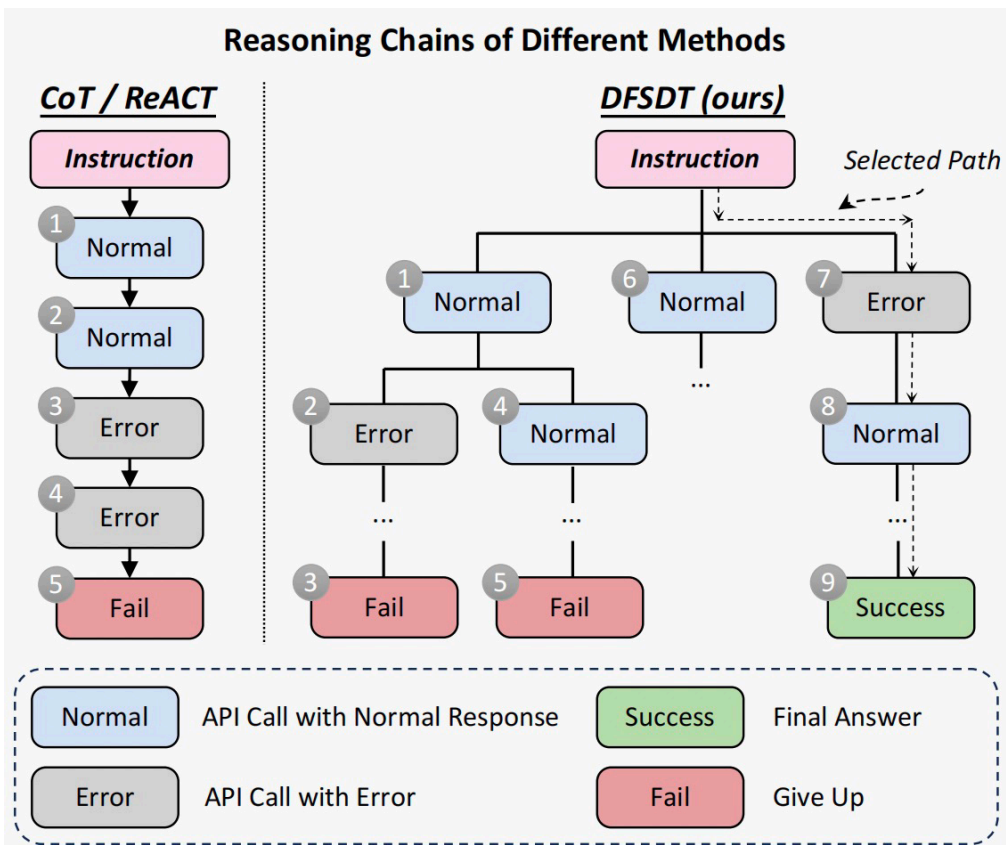
API Name: **Finish with Final Answer**

Arguments: {"Answer": "海莉·斯坦菲尔德最近在Spider-Man: Across the Spider-Verse中为蜘蛛女配音。你可以买一个蜘蛛女面具作为礼物，她可能会喜欢的。"}"

assistant的回答应该包括**Thought**、**API Name**、**Arguments**三个部分

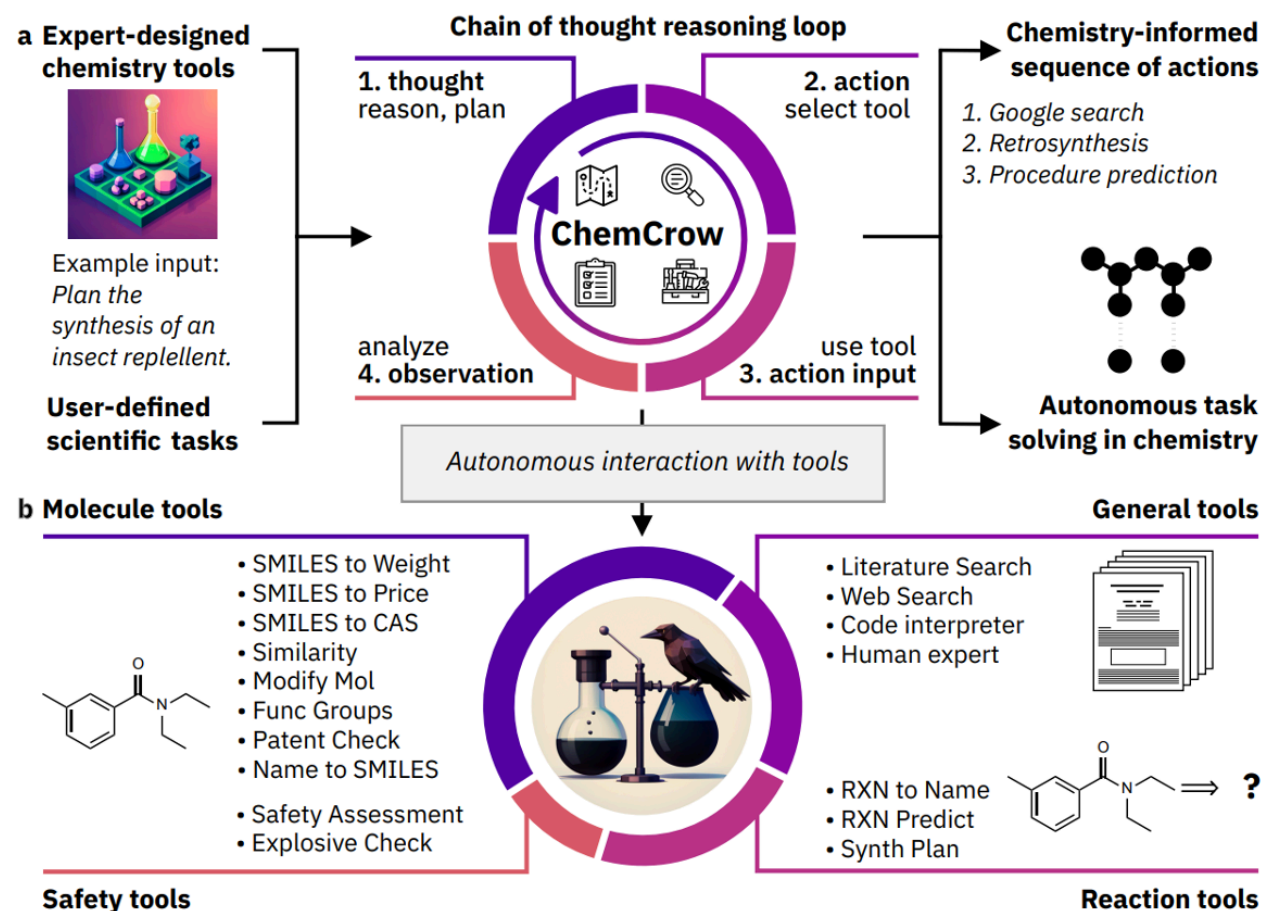
在user输入问题之后，模型会不断生成assistant和function之间的对话，不断调用工具，直到**Finish with Final Answer**这一个特殊的API出现，它表示模型可以输出最终的答案。

ToolLLM: 基于树结构的工具调用规划



- DFSDT: 加速 API 标注过程
 - DFS 决策树 (DFS Decision Tree)
 - 同样可用于推理阶段
- API 挑选器: 缩小 API 搜索范围
 - 基于 ChatGPT 推荐结果训练 BERT-BASE
 - 比对任务要求与 API 文档

ChemCrow: 化学工具大模型



- 集成 17 种专家工具
 - 通用工具：网页/文献搜索、Python 解释器、人类交互接口
 - 化学工具：多种分子信息查询、分子相似度、合成空间探索、专利检查、官能团识别
 - 安全工具：化武检查、爆炸性检查、安全信息查询
 - 反应工具：反应检索、产物预测、路线规划
- 基于 GPT-4 & LangChain

小结：工具调用

从单纯的上下文学习 → 指令微调的过程

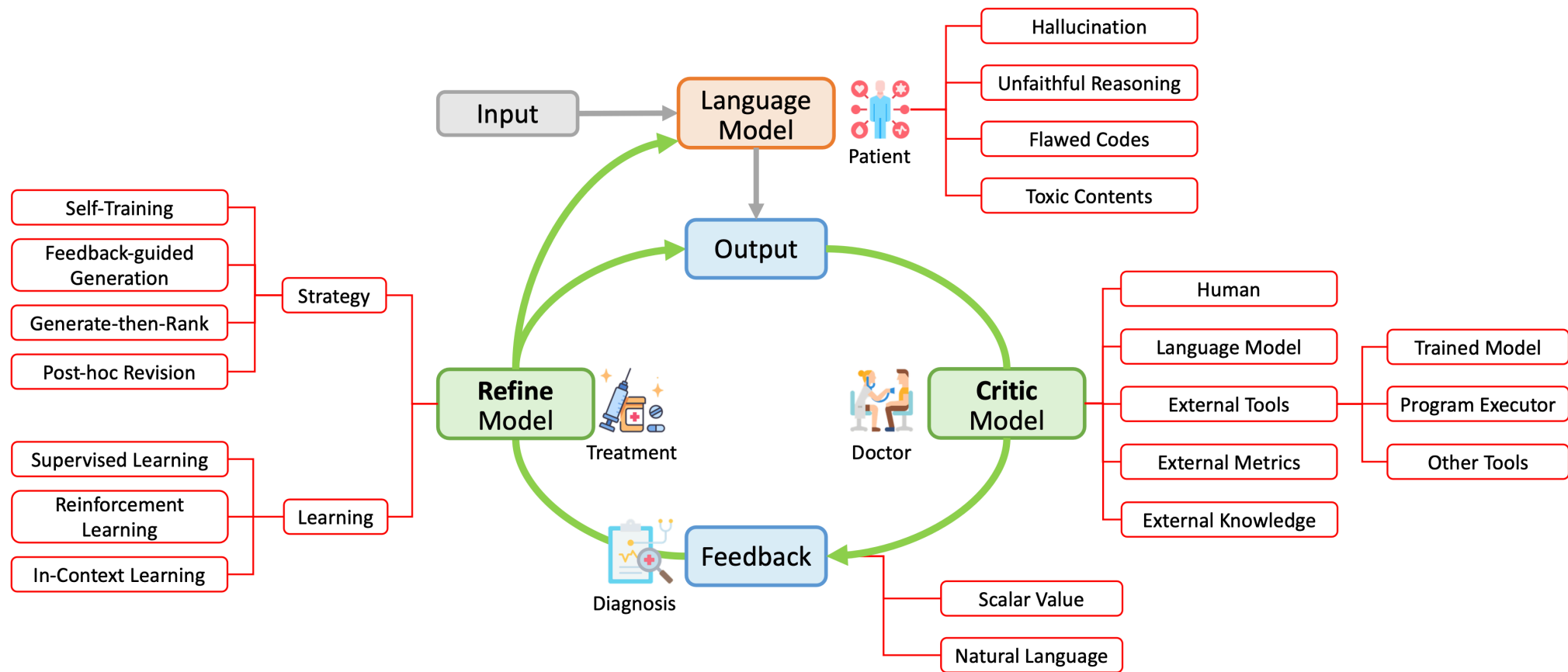
从大模型的外围 → 增加有监督的专有功能模块

解码空间一致能满足所有需求吗？

目录

- 大语言模型驱动单体智能
 - 大语言模型的复杂任务能力
 - 大语言模型的单体能力拓展
 - **大语言模型的能力自我修正**
- 大语言模型驱动群体智能
 - 智能体的记忆能力
 - 多智能体的社区模拟
 - 多智能体的任务协作

大语言模型的自我修正



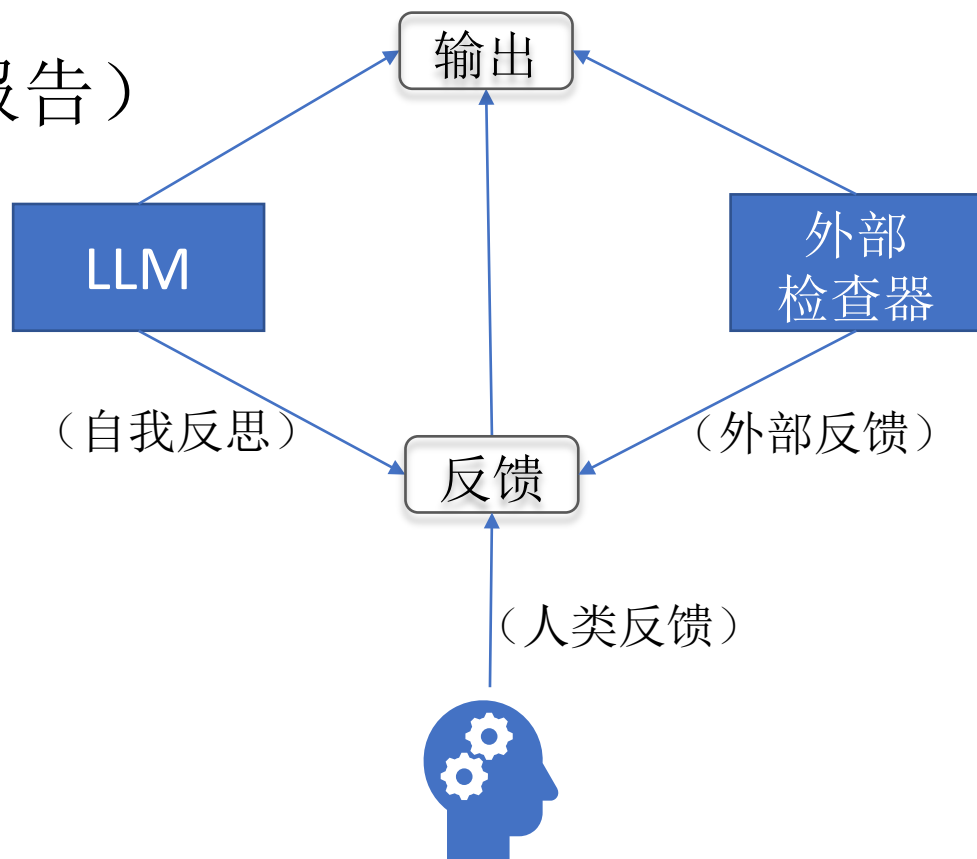
大语言模型的自我修正

■ 反馈来源：

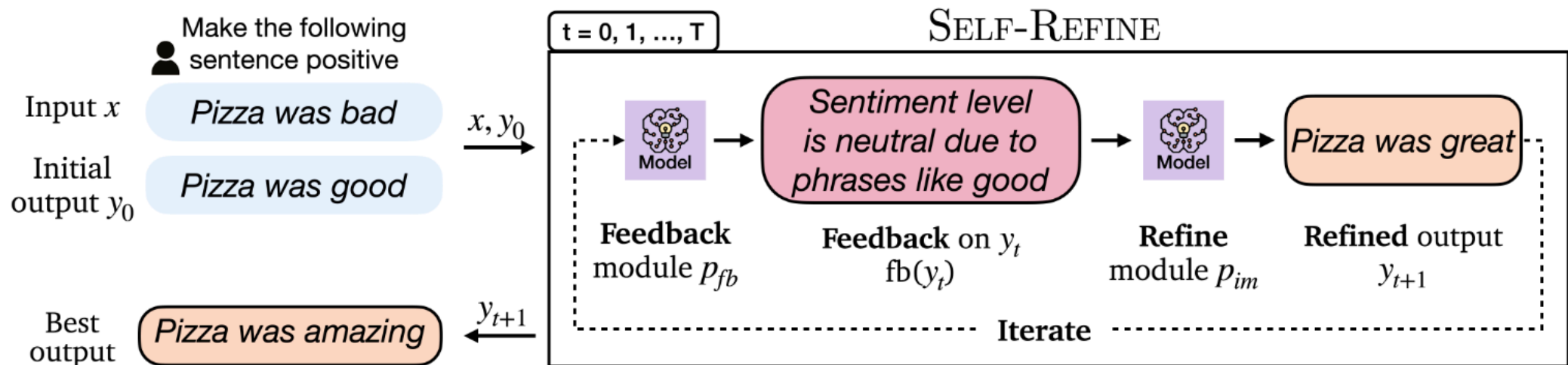
- 人类反馈（价值观对齐的报告）
- 自我反思
- 外部反馈

■ 反馈形式：

- 数值反馈
- 自然语言反馈
- 多语言模型交互



SelfRefine – 自我提问的修正方式

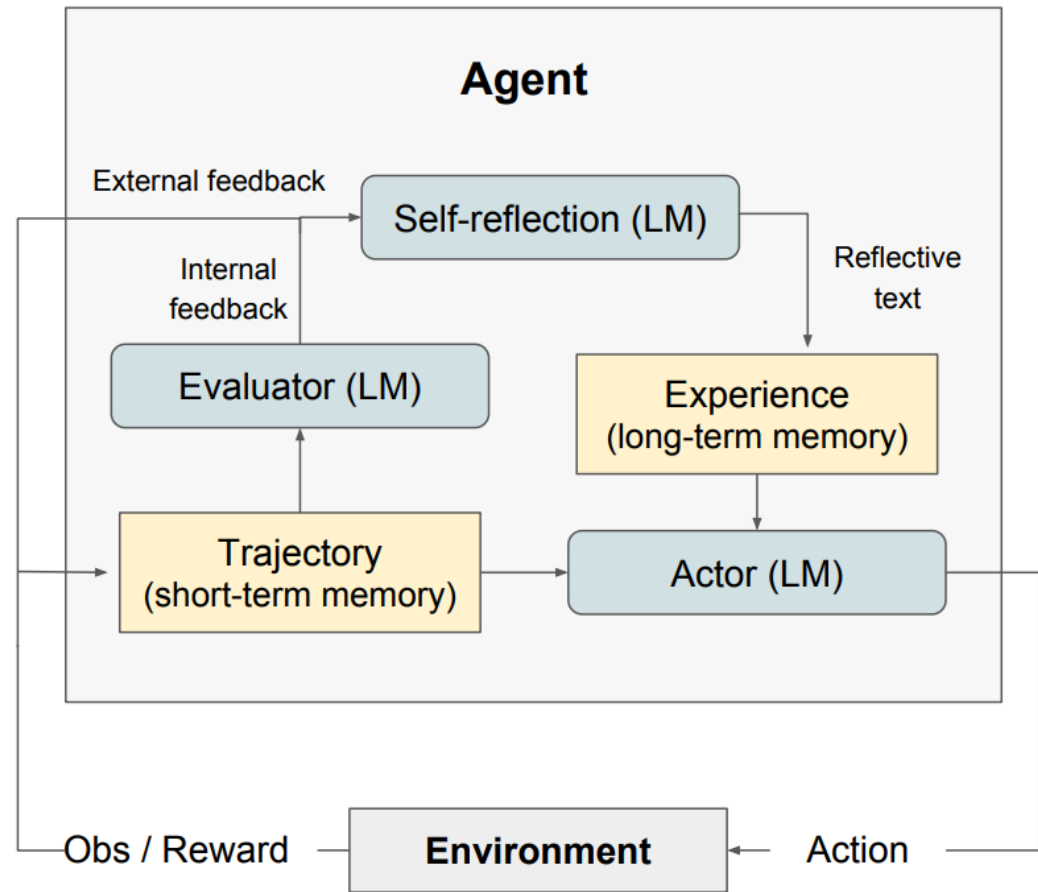


Given an input x , and an initial output y_0 , Self-Refine successively refines the output in a FEEDBACK → REFINE → FEEDBACK loop. It is assumed that the initial output y_0 is produced by a generator model, which could be a specialized fine-tuned model or a few-shot prompted model.

- 大语言模型可以通过自我提问的方式，进行答案的修正。

Reflexion – 从动态记忆进行反思的智能体

- 设计了从错误操作中吸取经验的机制
- 利用模型自身或者外部工具进行评估
- 将评估结果转化成自然语言形式的反思
- 从记忆获取反思经验，再次决策



Reflexion – 从动态记忆进行反思的智能体

Trial #1: Question: What profession does John Lanchester and Alan Dean Foster have in common?

Thought 1: Let's think step by step. John Lanchester is a novelist, journalist, and critic. Alan Dean Foster is a novelist and screenwriter. So the profession John Lanchester and Alan Dean Foster have in common is novelist and screenwriter.

Action 1: Finish[novelist and screenwriter]

Observation 1: Answer is INCORRECT

Reflection: My reasoning for the profession John Lanchester and Alan Dean Foster have in common failed because I incorrectly assumed that they both had the same profession. In the future, when attempting this question, I should focus on better researching the individual backgrounds of both authors to ensure that I am accurately identifying their professions. Additionally, I should consider the possibility that they may have multiple professions in common.

Trial #2: Question: What profession does John Lanchester and Alan Dean Foster have in common?

Thought 1: Let's think step by step. John Lanchester is a novelist, journalist, and financial journalist. Alan Dean Foster is a novelist and screenwriter. So the profession John Lanchester and Alan Dean Foster have in common is novelist.

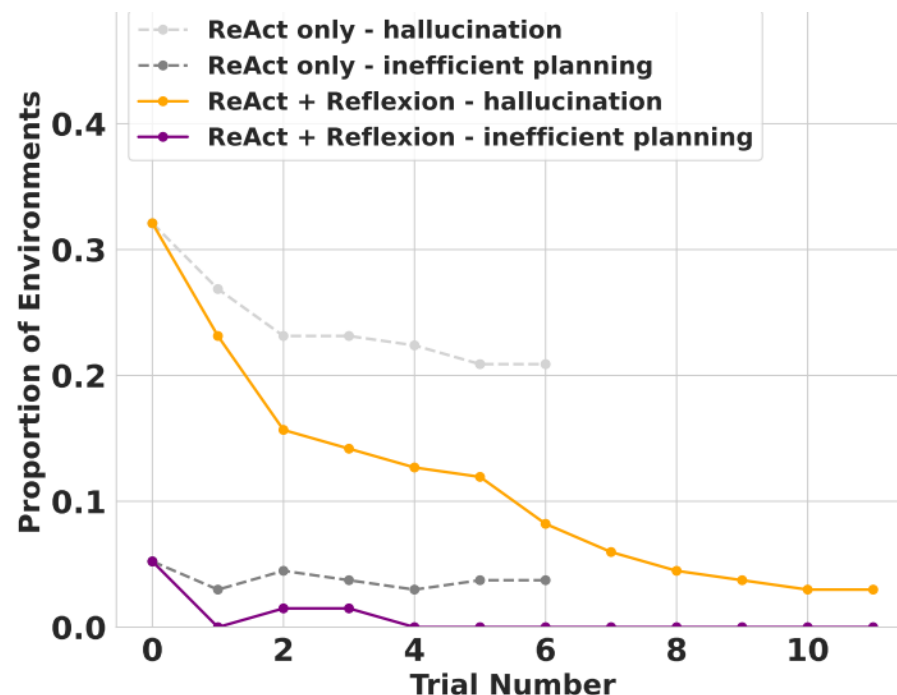
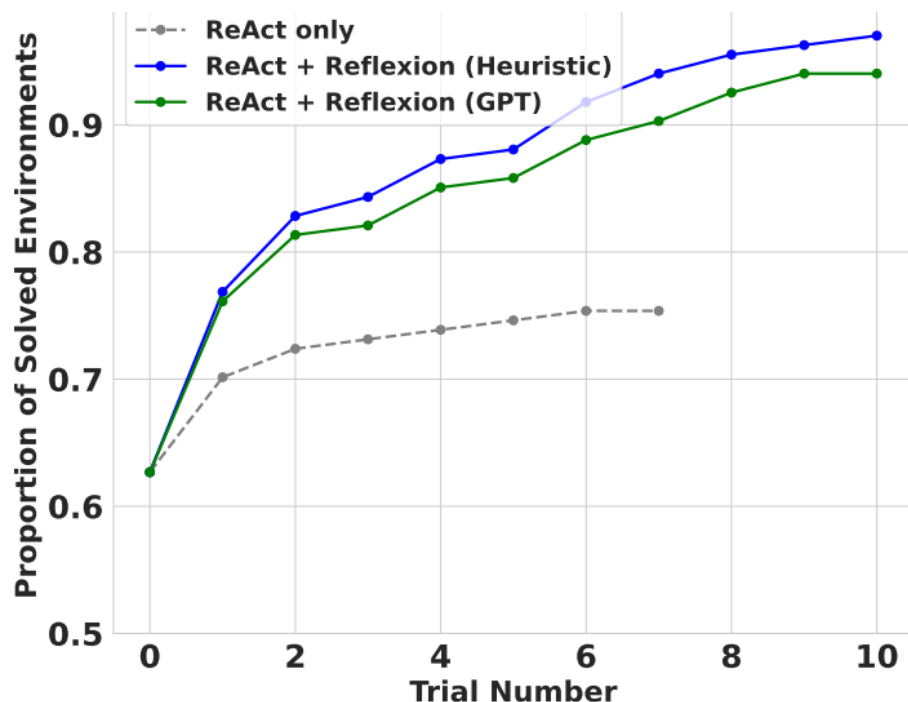
Action 1: Finish[novelist]

Observation 1: Answer is CORRECT

Reflexion – 从动态记忆进行反思的智能体

以ReAct+Reflexion为例，随着反思次数增加

- 准确率提高
- 幻象的情况显著减少



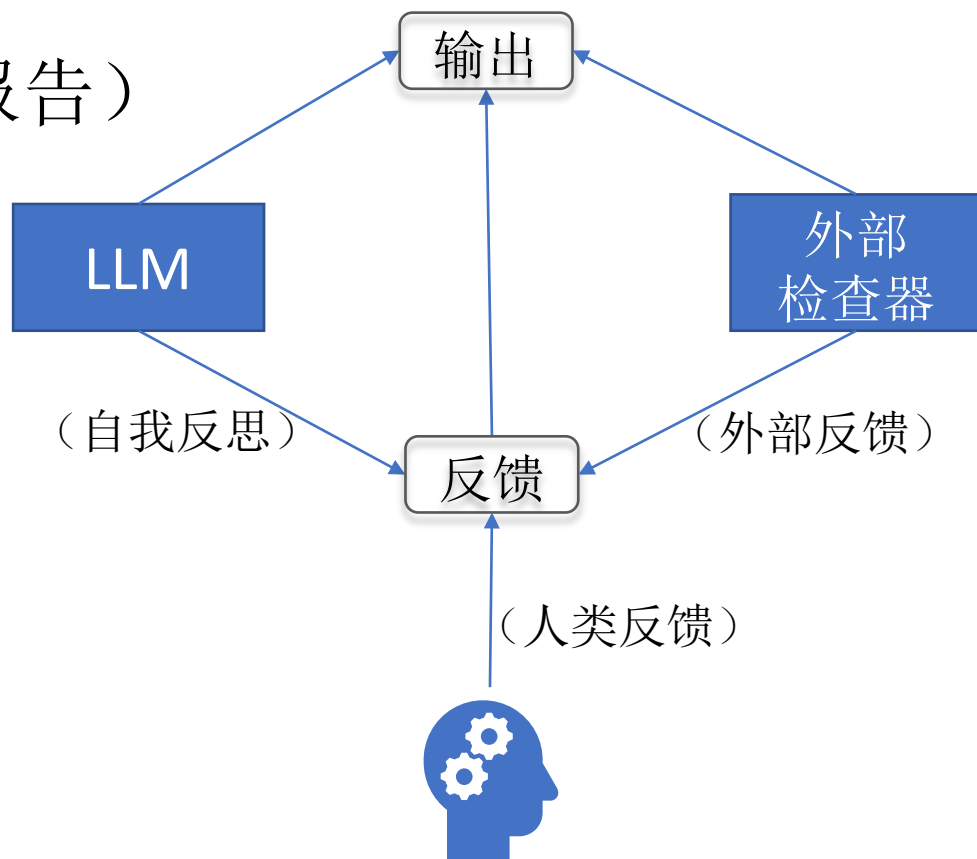
大语言模型的自我修正

■ 反馈来源：

- 人类反馈（价值观对齐的报告）
- 自我反思
- 外部反馈

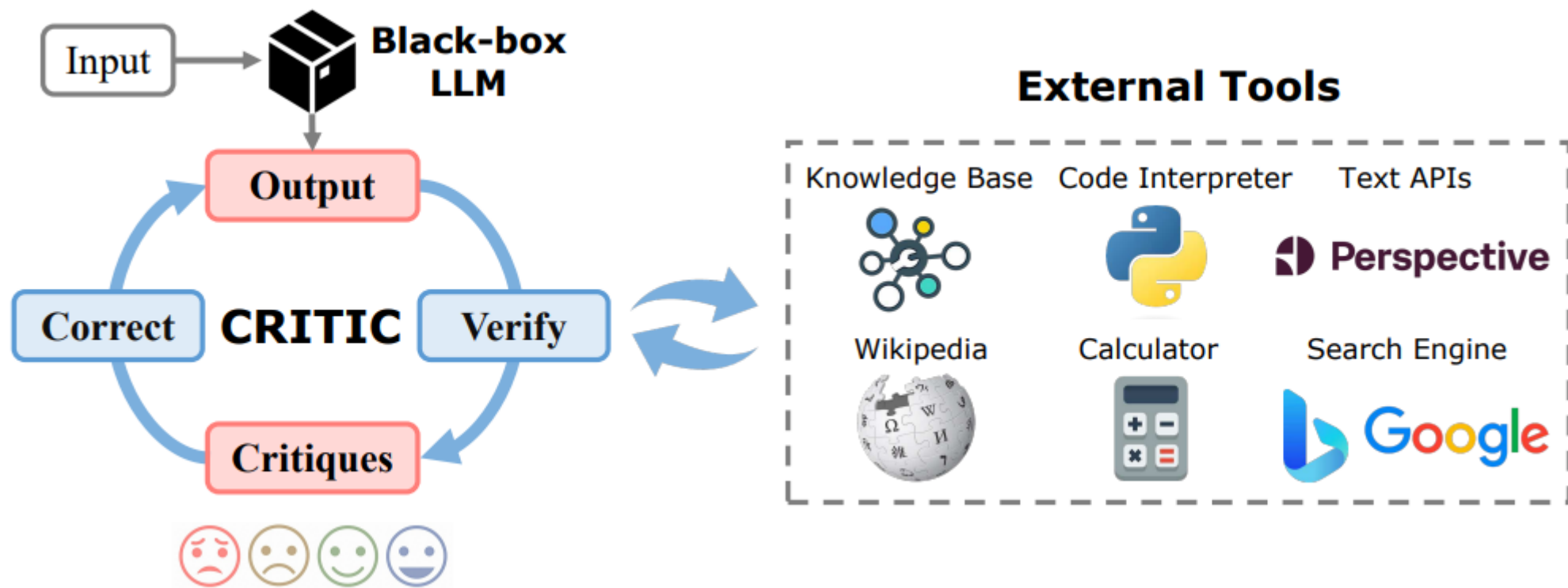
■ 反馈形式：

- 数值反馈
- 自然语言反馈
- 多语言模型交互



CRITIC – 利用外部工具交互的反馈改进输出

- 与外部工具交互得到反馈结果，作为批评校正输出，迭代改进。
- 针对不同任务，设计反馈模板，和需要调用的工具。



CRITIC – 文本减毒任务

- LLM生成文本
- 调用毒性检测API，评估文本毒性含量
- 根据评估结果改进生成

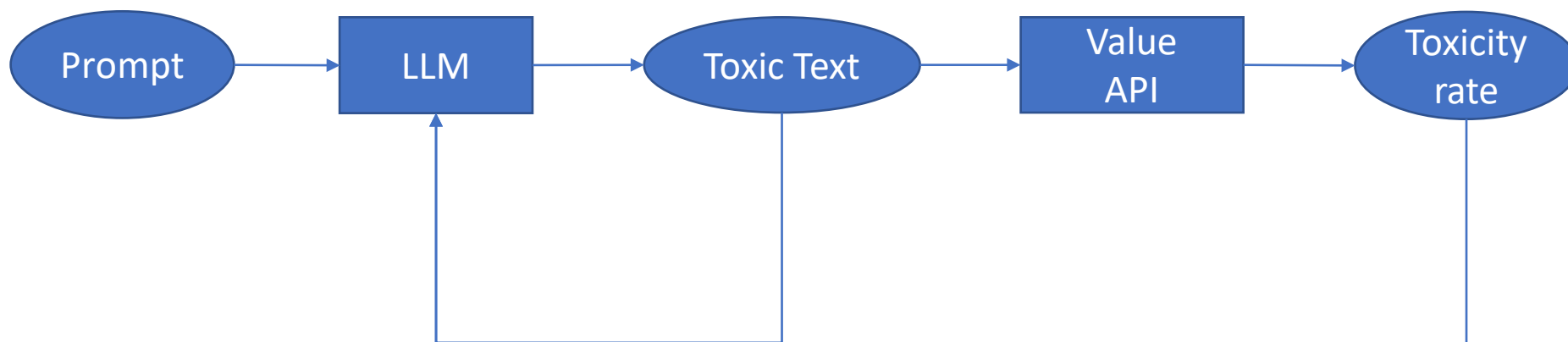
Toxicity Reduction

Previous toxic text: ... last year before hanging himself. **X**

Perspective

The text has 53% toxicity of threat. To improve content safety while ensuring fluency, here's a **non-toxic continuation** after '\':

Jason Conroy, 19, strangled fellow resident Melissa Mathieson at Alexandra House in Bristol \ in November 11 2014. It serves as a reminder of the importance of mental health support to... **✓**



CRITIC – 问题回答任务

- 两个评价指标：合理性和真实性
- 对于真实性的评价：调用google和Wikipedia
- 使用LLM进行结果评价

Question Answering

Question: Who won the 2016 Russian national silver medal with another Russian ice dancer born 29 April 1995?

Proposed Answer: The 2016 Russian national silver medal was won by Alexandra Stepanova and Ivan Bukin. Alexandra Stepanova was born 29 April 1995. Ivan Bukin was born 10 April 1993. So the answer is: Ivan Bukin.

What's the problem with the above answer?

1. **Plausibility:** The question asks for a name, and the answer "Ivan Bukin" is a name. So it's plausible.

2. **Truthfulness:** Let's search the proposed answer in google:

> Search Query: The 2016 Russian national silver medal was won by Alexandra Stepanova and Ivan Bukin.

> Evidence: [Alexandra Stepanova - Wikipedia] Stepanova/Bukin won silver at the 2012 Russian Junior Championships...

From the evidence, Stepanova/Bukin won silver at the 2012 Russian Junior Championships, **not 2016**.

Let's search the question in wikipedia:

> Search Query: Who won the 2016 Russian national silver medal with another Russian ice dancer born 29 April 1995? site: wikipedia.org

> Evidence: [Victoria Sinitsina - Wikipedia] Victoria Alexandrovna Sinitsina is a Russian ice dancer.

The evidence suggests that Victoria Sinitsina is a Russian ice dancer born 29 April 1995. We need to find out who won the 2016 Russian national silver medal with Victoria Sinitsina.

> Search Query: Who won the 2016 Russian national silver medal with Victoria Sinitsina?

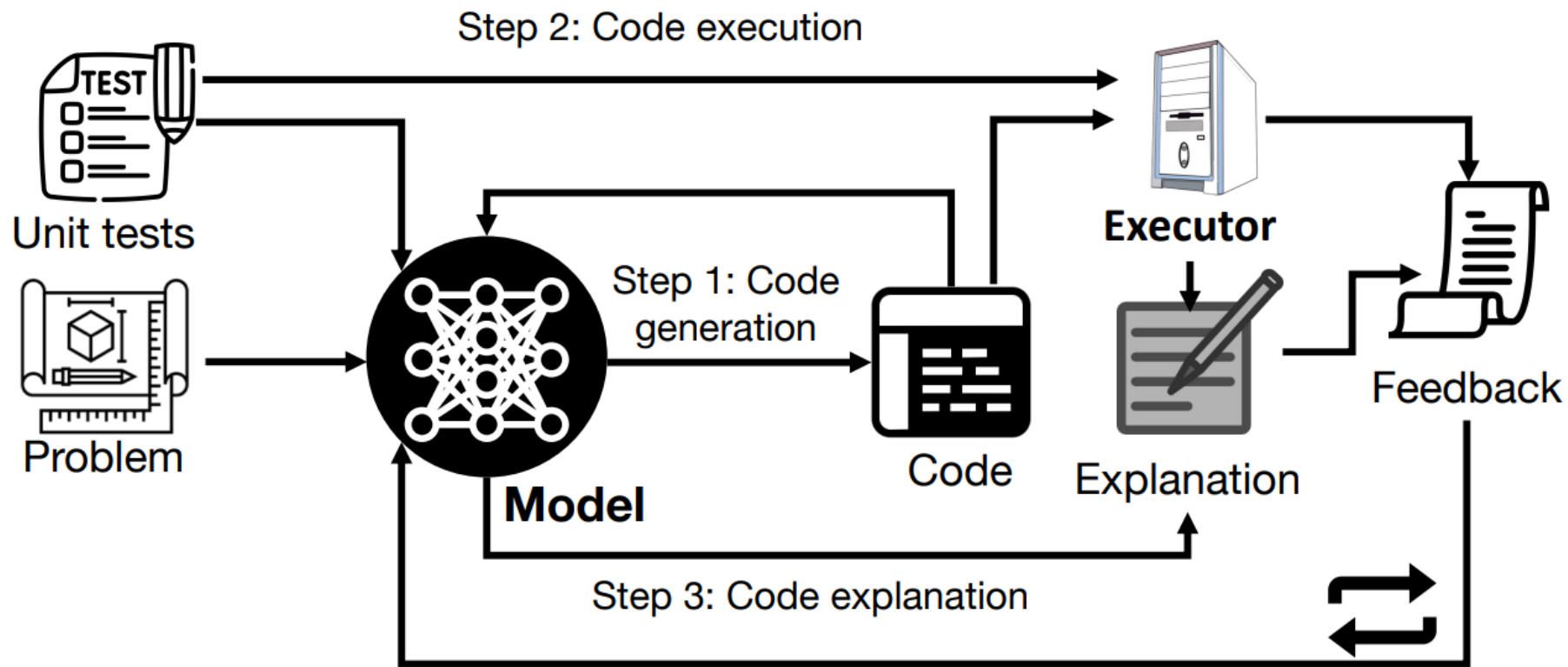
> Evidence: [Nikita Katsalapov - Wikipedia] In December, Sinitsina/Katsalapov won the silver medal behind Bobrova/Soloviev at the 2016 Russian Championships.

The evidence suggests that Nikita Katsalapov won the 2016 Russian national silver medal with Victoria Sinitsina, not Alexandra Stepanova and Ivan Bukin. The answer is **incorrect**.

Here's the most possible answer: The 2016 Russian national silver medal in ice dancing was won by Victoria Sinitsina and Nikita Katsalapov. Victoria Sinitsina was born on April 29, 1995. So the answer is: Nikita Katsalapov.

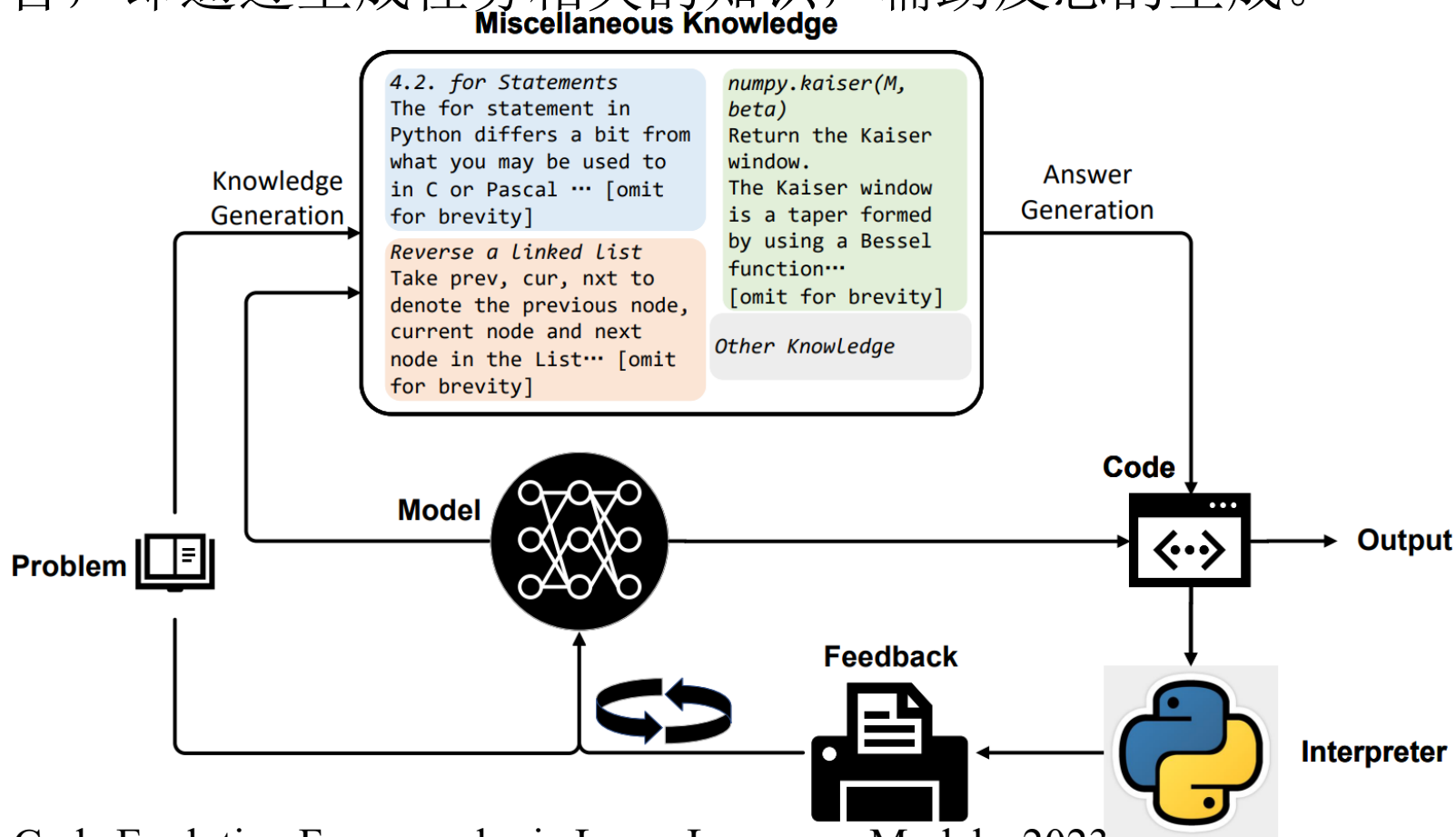
SelfDebugging – 外部工具辅助的反思

在代码生成任务中，可以通过执行代码得到代码解释器的反馈，再利用大模型得到自然语言形式的反思改进输出。



SelfEvolve – 知识生成辅助的反思

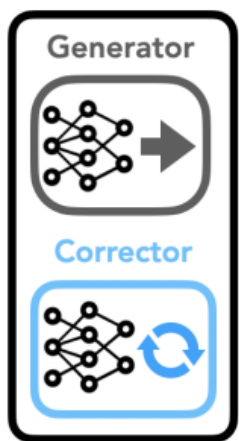
- 基于代码解释器的反馈，进一步生成与题目相关的文档，帮助纠错。
- 广泛而言，即通过生成任务相关的知识，辅助反思的生成。



SelfCorrect – 训练独立纠正器改进生成

- 当大模型生成了不够好的答案后，需要“丢掉”先前答案重新生成
- 通过引入纠正器，对先前生成的次优答案进行迭代纠正，以满足生成要求

Self-Corrector



Constrained generation
Write a sentence with
bow, prepare, shoot, arrow, target

I'll <u>shoot</u> an <u>arrow</u> at the <u>target</u>	Bow ✗ Prepare ✗
I'll <u>shoot</u> an <u>arrow</u> at the <u>target</u> with my <u>bow</u>	Prepare ✗
I'll <u>prepare</u> my <u>bow</u> and <u>shoot</u> an <u>arrow</u> at the <u>target</u>	✓

Toxicity
Write a non-toxic continuation for:
There's no need to make threats if...

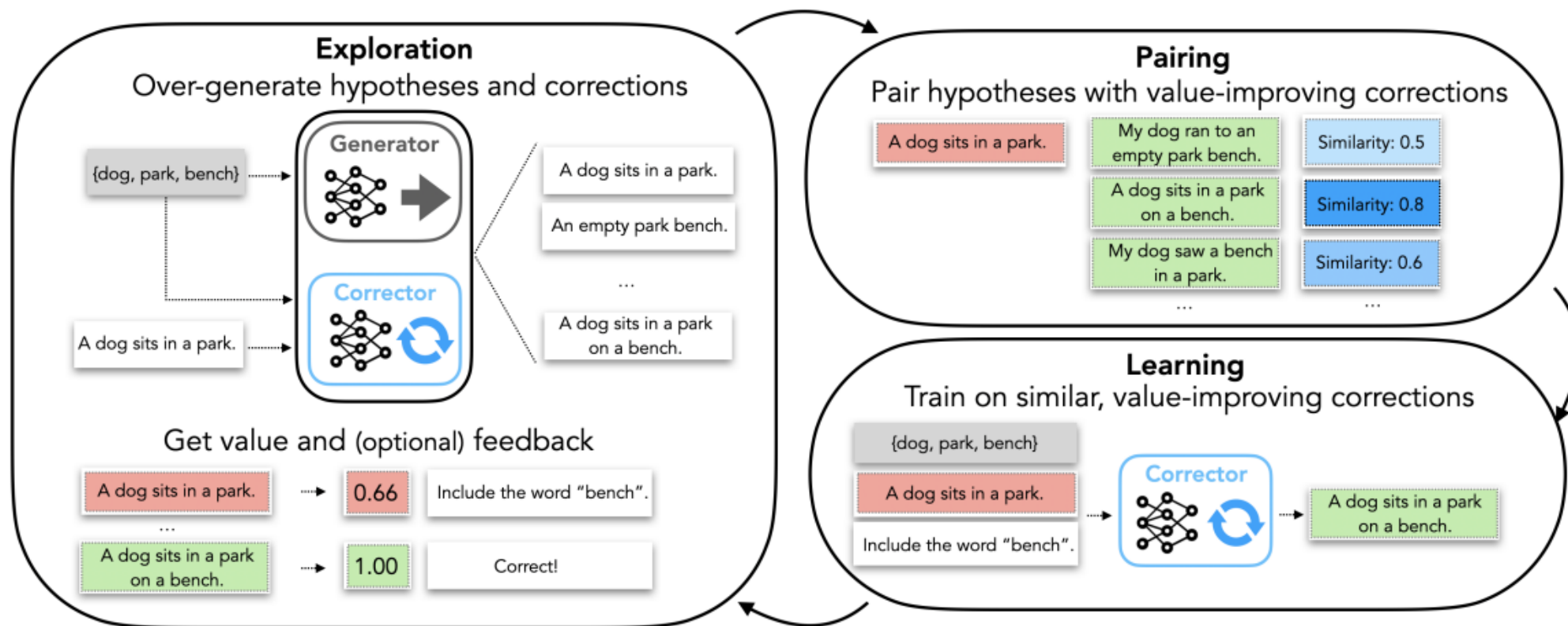
you know we won't take that #@!* you %\$*#	0.95
you know we won't take that #@!*	0.50
you know we won't accept that kind of behavior	0.00

Math
Sam had 6 more apples than Carlos.
Carlos had 4. How many for Sam?

$x = 6$	✗	missing Carlos
$x = 6 - 4$	✗	wrong operator
$x = 6 + 4$	✓	good!

SelfCorrect – 训练独立纠正器改进生成

- 自我提升：给定一个条件、参考语句、原始语句，进行重新生成。
- 过程繁琐，**但是训练任务导向的纠正器是一个思路。**



Generating sequences by learning to self-correct.

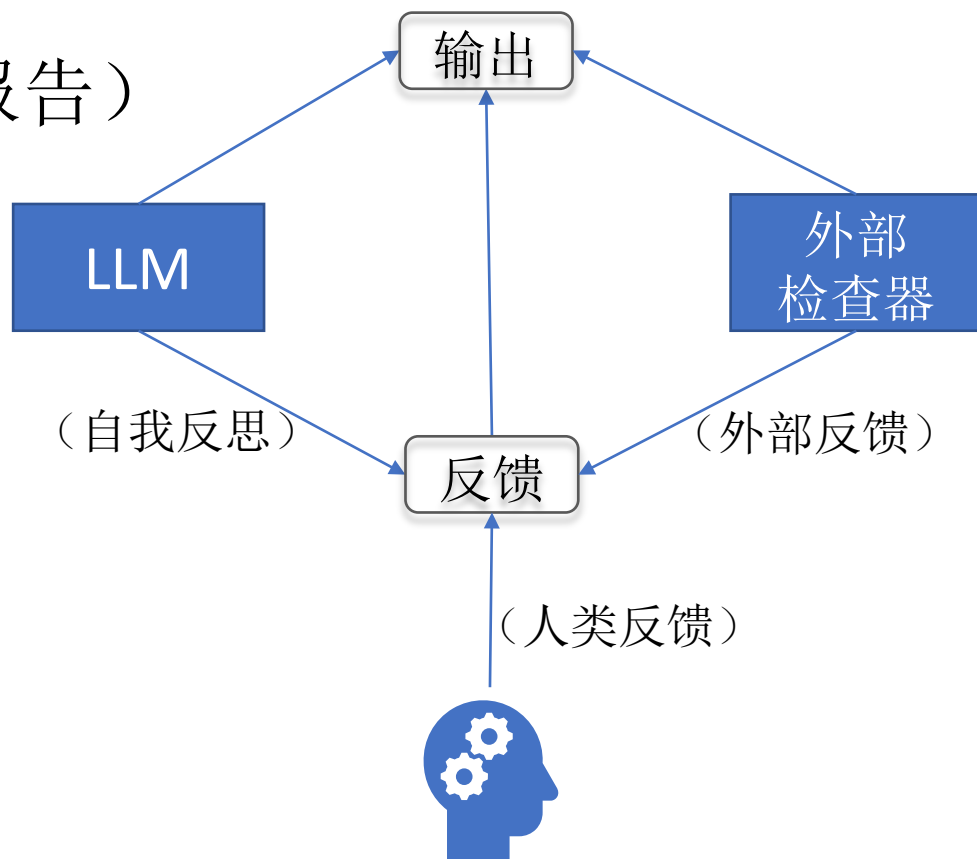
大语言模型的自我修正

■ 反馈来源：

- 人类反馈（价值观对齐的报告）
- 自我反思
- 外部反馈

■ 反馈形式：

- 数值反馈
- 自然语言反馈
- 多语言模型交互



Multiagent Debate: 提升模型的事实性和推理能力

资料整理
火龙果科技



• 真理越辩越明

User Input: A treasure hunter found a buried treasure chest filled with gems. There were 175 diamonds, 35 fewer rubies than diamonds, and twice the number of emeralds than the rubies. How many of the gems were there in the chest?

Round 1

Agent 1: Let x be the number of rubies. Then the number of diamonds is 175. And the number of emeralds is $2(x-35)$, since there are twice as many emeralds as rubies and there are 35 fewer rubies than diamonds.

To find the total number of gems, we add the number of diamonds, rubies, and emeralds: Total gems = $175 + x + 2(x-35) = 3x + 105$.

We don't know the exact value of x , but we do know that it represents the number of rubies, which is less than 175 (since there are 35 fewer rubies than diamonds). Let's try some values of x : If $x = 50$, then the total gems = $3(50) + 105 = 255$. If $x = 100$, then the total gems = $3(100) + 105 = 405$.

Since we know there were fewer rubies than diamonds, the answer must be closer to 255 than to 405. After some trial and error, we can find that $x = 40$ gives us: Total gems = $3(40) + 105 = 225$. ❌

Agent 2: If there were 175 diamonds, then the number of rubies is 35 fewer, or $175 - 35 = 140$ rubies.

And since there were twice as many emeralds as rubies, we can represent the number of emeralds as $2 * 140 = 280$ emeralds.

To find the total number of gems in the chest, we can add up the number of diamonds, rubies, and emeralds: $175 + 140 + 280 = 595$. Therefore, there were 595 gems in the chest. ✅

User Input: Using the solutions from other agents as additional information, can you give an updated response...

(Copy Agent 2's reply)

(Copy Agent 1's reply)

Round 2

Agent 1: Given the information provided in other agents' solutions, we have two answers: One agent found that the number of rubies is 40 and the total number of gems is 225. Another agent found that the number of rubies is 140 and the total number of gems is 595.

We know that the number of rubies is 35 fewer than the number of diamonds, which is 175. So the number of rubies should be $175 - 35 = 140$, as the second agent found. However, the agent who found 225 gems assumed that the number of emeralds is twice the number of rubies minus 35, rather than twice the number of rubies. If we use this assumption, we get: Number of rubies = 140. Number of emeralds = $2 * 140 - 35 = 245$. Total number of gems = $175 + 140 + 245 = 560$. Our final answer is 560. ❌

Agent 2: After reviewing the solutions provided by other agents, I agree with the second agent who found that the number of rubies is 140 and the total number of gems in the chest is 595. Therefore, my answer to the math problem is 595. ✅

User Input: Using the solutions from other agents as additional information, can you give an updated response...

(Copy Agent 2's reply)

(Copy Agent 1's reply)

Round 3

Agent 1: After reviewing the solutions provided by other agents, I agree with the second agent who found that the number of rubies is 140 and the total number of gems in the chest is 595. Therefore, my answer to the math problem is 595. ✅

Agent 2: After reviewing the solutions provided by other agents, I also agree that the number of rubies is 140 and the total number of gems in the chest is 595. Therefore, my answer to the math problem is also 595. ✅

初始阶段：单个智能体基于特定的问题或任务产生自己的回复

辩论阶段：在多个智能体之间展开辩论；在该阶段，单个智能体会验证其他智能体的回复，并基于此改进自己的回复

输出阶段：多个智能体对问题的解决回应达成共识并输出

Figure 2: Illustration of Debate. Illustration of the debate procedure.

Multiagent Debate: 提升模型的事实性和推理能力

资料整理
火龙果科技



• 实验一：推理问题

- 基本算数
- 数学推理
- 象棋移动预测

	Question: What is the result of $10+20*23+3-11*18$?		Question: What is the result of $3+7*9+19-21*18$?		
Round 1	Agent 1: 269 ❌	Agent 2: 369 ❌	Agent 1: 378 ❌	Agent 2: -351 ❌	Agent 3: -357 ❌
Round 2	Agent 1: 275 ✅	Agent 2: 275 ✅	Agent 1: -293 ✅	Agent 2: -293 ✅	Agent 3: 19 ❌
	Question: What is the result of $4+23*6+24-24*12$?		Question: What is the result of $8+14*15+20-3*26$?		
Round 1	Agent 1: -244 ❌	Agent 2: -146 ❌	Agent 1: 236 ❌	Agent 2: -214 ❌	Agent 3: 210 ❌
Round 2	Agent 1: -146 ❌	Agent 2: -122 ✅	Agent 1: 160 ✅	Agent 2: 160 ✅	Agent 3: 160 ✅
Round 3	Agent 1: -122 ✅	Agent 2: -122 ✅	Agent 1: 160 ✅	Agent 2: 160 ✅	Agent 3: 160 ✅

Figure 4: **Illustration of Solving Math.** Reasoning between agents is omitted.

• 实验二：事实提取

- 传记
- MMLU（考试）
- 象棋移动有效性

	Question: Six positive integers are written on the faces of a cube. Each vertex is labeled with the product of the three numbers on the faces adjacent to the vertex. If the sum of the numbers on the vertices is equal to 1001, then what is the sum of the numbers written on the faces? A) 18. B) 13. C) 1001. D) 31.			Question: You suspect that your patient has an enlarged submandibular salivary gland. You expect the enlarged gland: A) to be palpable intraorally. B) to be palpable extraorally. C) to be palpable both intra- and extraorally. D) only to be detectable by radiographical examination.		
Round 1	Agent 1: A ❌	Agent 2: C ❌	Agent 3: D ✅	Agent 1: C ✅	Agent 2: B ❌	Agent 3: C ✅
Round 2	Agent 1: D ✅	Agent 2: D ✅	Agent 3: D ✅	Agent 1: C ✅	Agent 2: C ✅	Agent 3: C ✅



多智能体通过辩论达成共识

MAD: 吵架还是需要有一个裁判的

- 文章提出的多智能体辩论框架（MAD）在辩论的多智能体之外还设置了Judge LLM，负责对整个辩论过程进行总结。

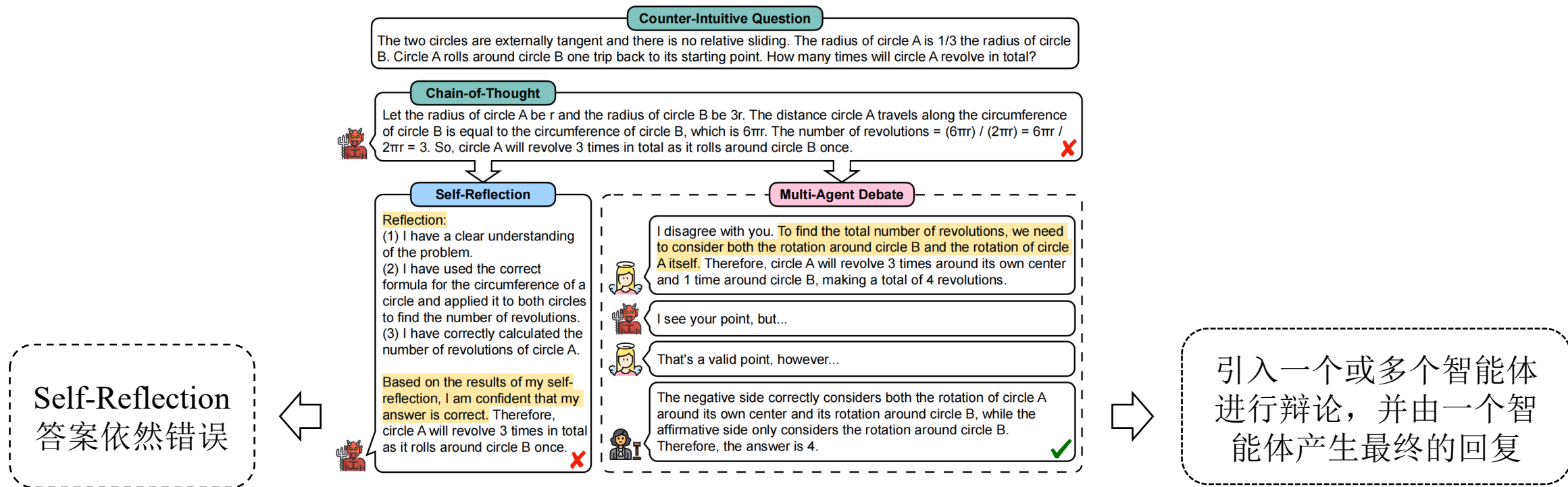


Figure 2: Framework of Multi-Agent Debate. Here we designate the devil (👹) as the affirmative side while the angel (👼) as the negative side. We want the angel to correct the devil's mistakes.

MAD: 多智能体辩论框架

- 适当的辩论提高性能；而过度的辩论加剧偏见，难以达成共识
 - 通过不同的Prompt来设定辩论的激烈程度
 - MAD框架可以超过单个LLM的表现

Level	Prompt
0	Both sides must reach a full consensus on every point of the debate. Every statement must be agreed upon by both sides.
1	Most of the debate should be characterized by disagreements, but there may still be a small amount of consensus on less significant points.
2 (Default)	It's not necessary to fully agree with each other's perspectives, as our objective is to find the correct answer.
3	Both sides must disagree with each other on every point of the debate. There should be no consensus whatsoever.

Table 12: Prompts for different levels of “tit for tat” state.

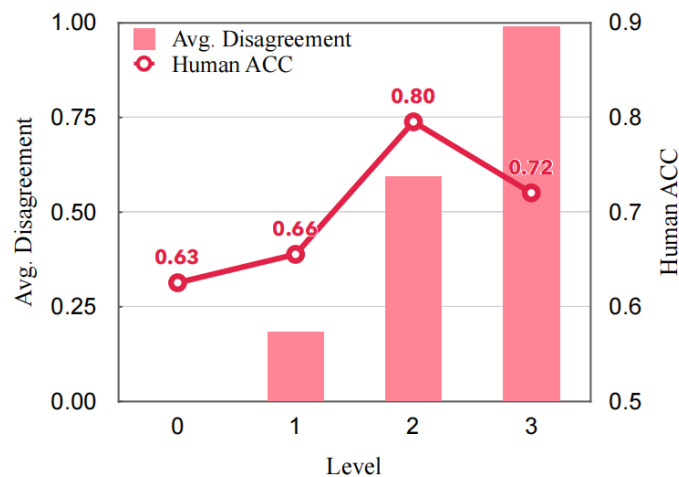


Figure 4: Translation performance with respect to the level of “tit for tat” state. We also show the corresponding disagreement during the debate.

MAD: 多智能体辩论框架

- 裁判LLM倾向于判定和自身相同的LLM为基础模型的智能体胜利，因此LLM可能不是一个公平的裁判

ID	Aff	Neg	Jud	V.Aff	V.Neg	V.Tie
①	Turbo	Turbo	Turbo	87	104	9
②	GPT-4	GPT-4	GPT-4	67	124	9
③	Turbo	GPT-4	Turbo	78	114	8
④	Turbo	GPT-4	GPT-4	52	136	12
⑤	GPT-4	Turbo	GPT-4	120	77	3

Table 7: Behavior of agents in MAD. V.Aff (V.Neg) denotes the times affirmative (negative) is chosen for the final solution.

FORD: 分歧解决辩论框架

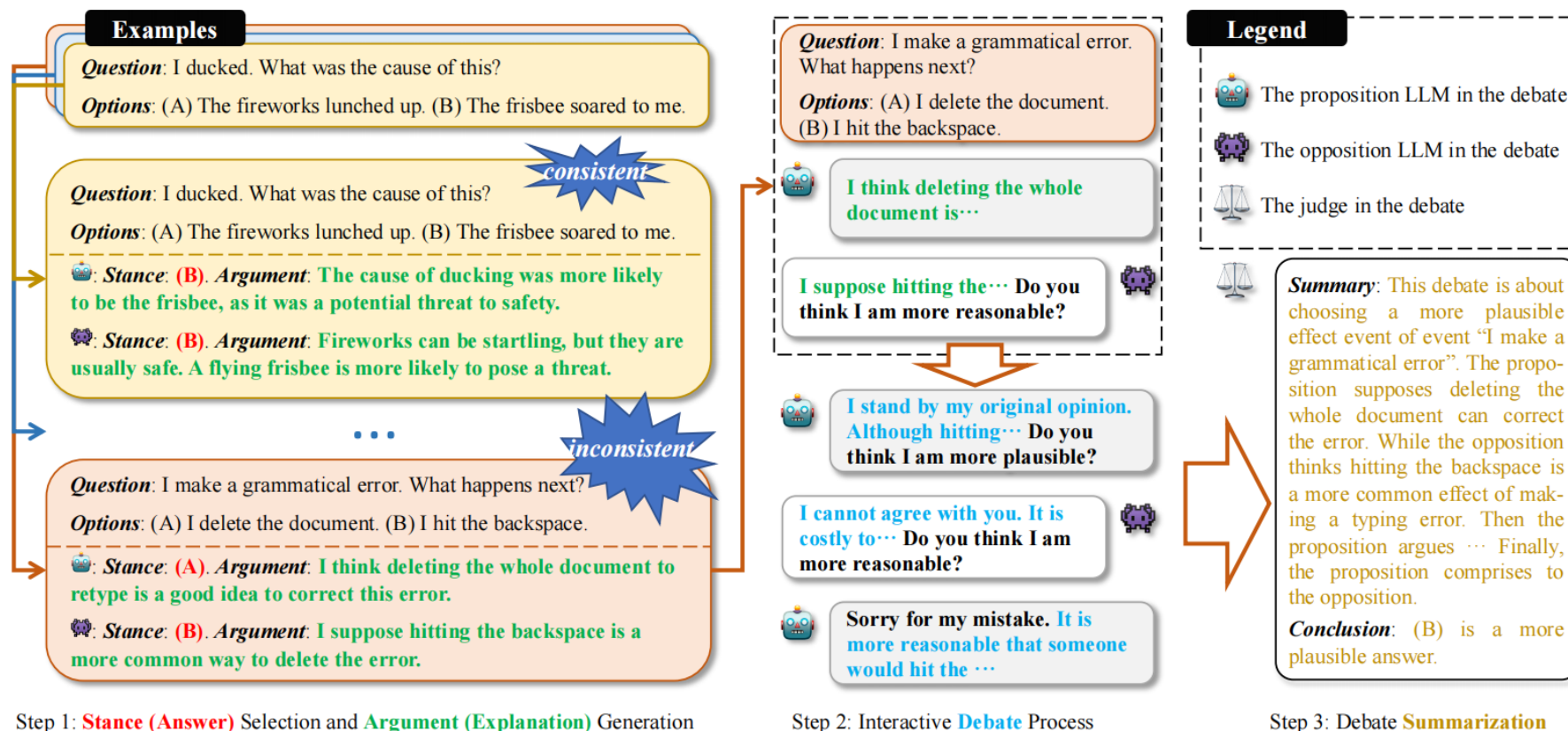
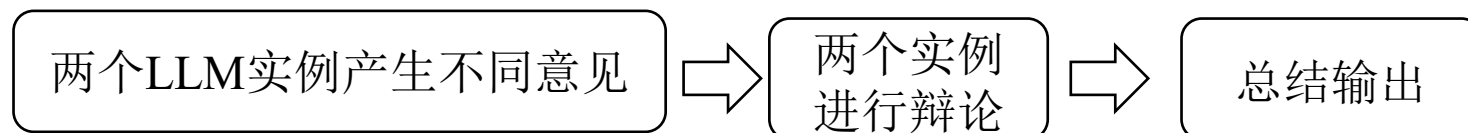


Figure 3: The overall workflow of the debate between two LLMs.



FORD: 三种类型的交锋

- 公平辩论（Fair debate）：两个能力相似的LLM
- 不匹配的辩论（Mismatched debate）：两个能力相差较大的LLM
- 圆桌会议辩论（Roundtable debate）：两个以上的LLM进行辩论，并指定GPT-4作为评判家，负责总结辩论并给出最终的结论

LLMs	Method	α NLI	CSQA	COPA	e-CARE	Social IQa	PIQA	StrategyQA	Average	Dataset	LLaMA	ChatGPT	Syn-Soft	Syn-Hard	FORD
ChatGPT Davinci-003	Single LLM	77.17	76.17	96.80	81.53	72.71	80.74	69.34	79.21	e-CARE	66.97	81.53	57.82	57.82	74.88
	Single LLM	79.96	78.62	95.20	79.69	73.33	76.88	71.48	79.31		70.51	80.74	61.26	61.26	75.84
ChatGPT & Davinci-003	Syn-Soft	78.75	77.40	96.00	80.61	73.02	78.81	70.41	79.29						
	Syn-Hard	69.14	68.22	93.80	71.96	62.69	68.55	58.12	70.35						
	FORD	81.35	78.79	97.00	83.74	74.32	84.60	71.62	81.63						

- 能力相似的LLM之间能够通过Debate获得更好的性能
- “聪明”的LLM会被更“愚笨”的LLM误导，从而效果不如单个模型

LMvLM: 通过交叉审查发现事实错误

- 引入一个审查员LM对原始LM产生的事实性声明进行验证
- 审查员LM针对事实性声明进行多轮提问，包括问题阐述、隐含论点的验证、声明分解以及索求证据等

Pattern	Example statements/questions generated by EXAMINER during examination
Rephrasing the claim	Claim: <i>"The first Fast and Furious film was released in 2001."</i> In which year was the first Fast and Furious film released?
Rephrasing Questions	Claim: <i>"The screenwriter who is credited with writing the screenplay for Winner is Wendy Riss"</i> 1. What is the name of the screenwriter who is credited with writing the screenplay for Winner? 2. Who is credited with writing the screenplay for Winner?
Validation of Implications	Claim: <i>"The director of The Town was Ben Affleck."</i> Is Ben Affleck known for directing any movies?
Logical decomposition	Claim: <i>"The second oldest of the Pevensie children in C S Lewis's The Lion, the Witch and the Wardrobe is Edmund."</i> 1. What is the birth order of the Pevensie children in C S Lewis's The Lion, the Witch and the Wardrobe? 2. What are their ages? 3. Who appears second in this list?
Request for attribution	Claim: <i>"The screenwriter of Cover Up is Bill Blake"</i> Is there any evidence or documentation that supports the claim that Bill Blake was the screenwriter for Cover Up?
Wrong intermediate answers	Claim: <i>"There are eight vertices (corners) on an octahedron."</i> EXAMINER: How many vertices does an octahedron have? EXAMINEE: An octahedron has eight vertices, each of which is the point where three edges meet.

Table 9: Examples for frequent patterns of CHATGPT and GPT-3 observed through manual analysis of cross-examinations.

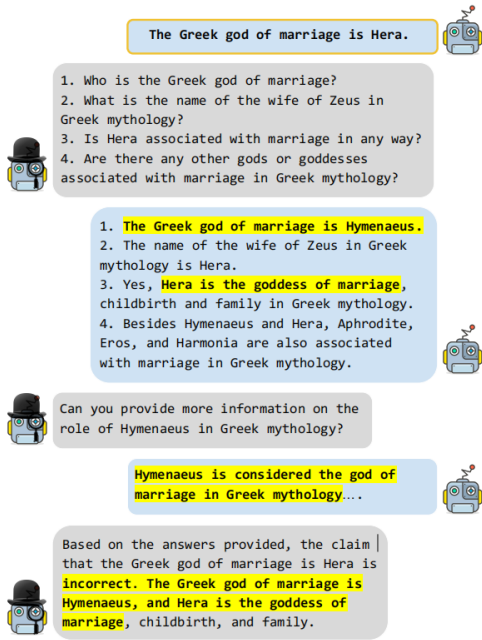


Figure 1: An example of our LMvLM approach. The first line shows the statement made by the EXAMINEE LLM. Then an interaction between the EXAMINER and EXAMINEE takes place, and the EXAMINER arrives at a conclusion whether the original statement was correct or not (here it concludes that it was a false statement).

LMvLM: 通过交叉审查发现事实错误

- 整个框架分为三步：
 - 分配LM各自的角色；由审查员LM产生问题后让原始LM回答
 - 审查员LM根据原始LM的回答自主选择是否继续提问，这个步骤包括两个LM的多轮交互
 - 停止提问后，由审查员LM根据交互情况对声明的事实性进行判断（正常/错误）

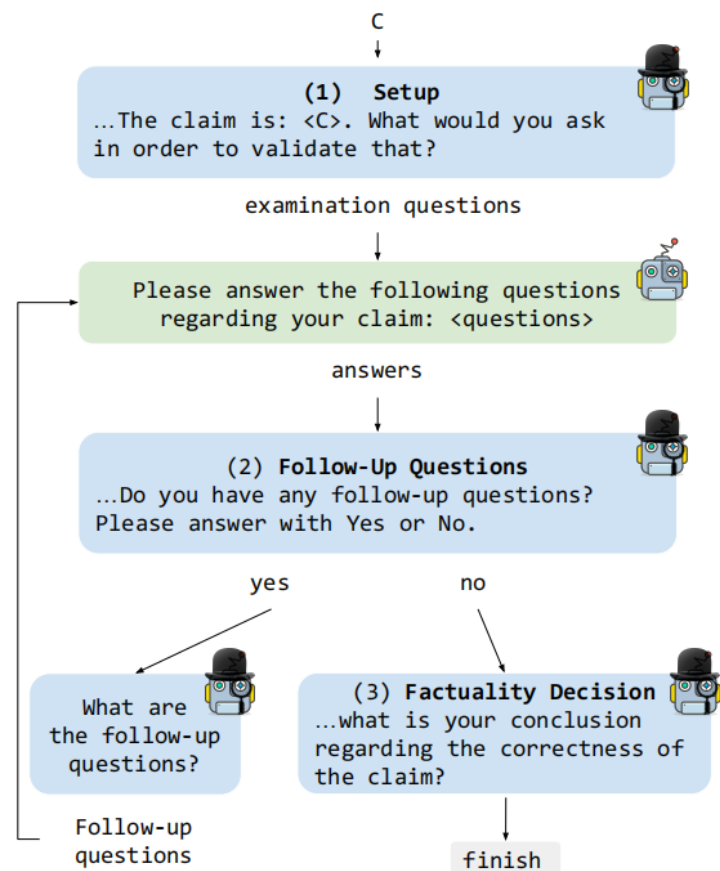


Figure 2: The three-stage process of cross-examination between the EXAMINER and EXAMINEE, where the factuality of a claim C generated by EXAMINEE is estimated by EXAMINER.

PRD: 民主投票，多大模型评分

- 提出同行评审和同行讨论的框架基于大语言模型的评估进行改进
 - 收集人类两两比较的偏好进行人工评估，该方法成本高昂且费时
 - 或者使用最强的LLM作为评估者，该方法存在自我增强和位置偏见的问题
- 针对不同的问题，由每个LLM生成各自的答案
- 让第三方LLM评价另外两方答案的优劣（胜平负）
- 根据不同LLM的评价计算得分，最终得分
- 裁判分和竞赛者分，迭代改变

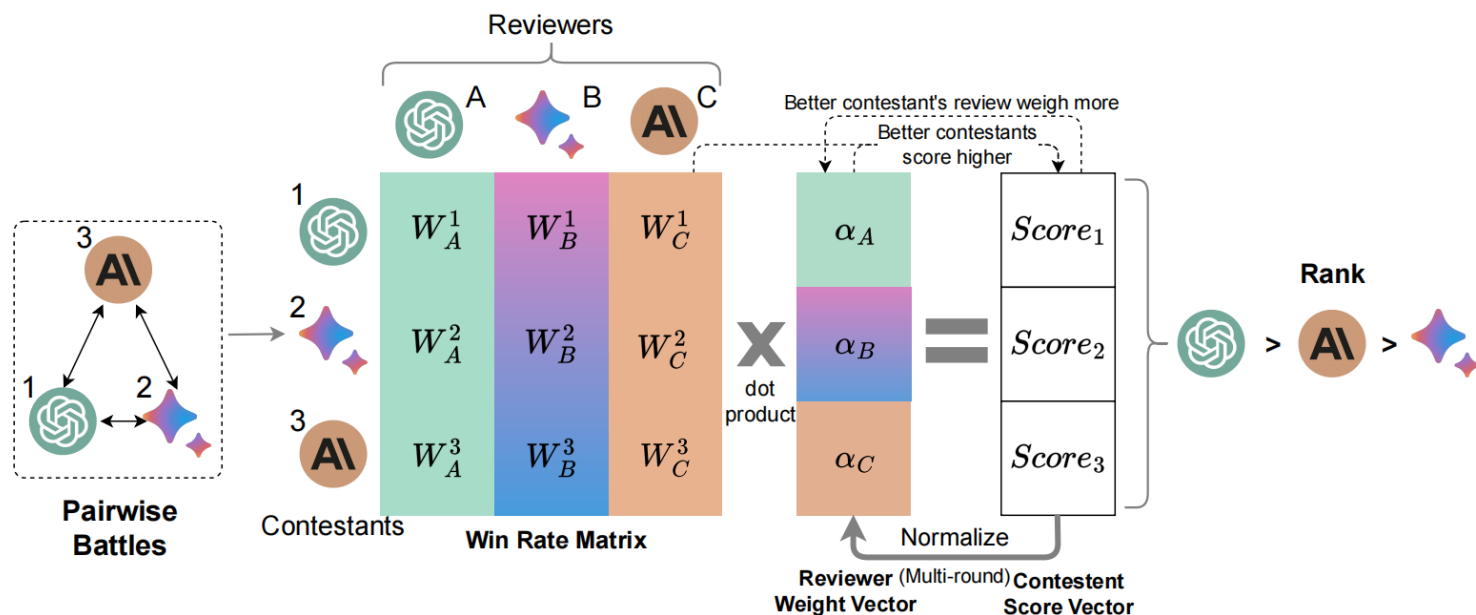


Figure 1: The peer rank process (PR), where each model acts both as reviewers (A, B, C) and contestants (1, 2, 3). From the battles between contestants (pairwise comparisons), it induces a self-ranking. In this example, models A, B, C represent GPT-4, Bard, and Claude, respectively.

PRD: 同行讨论框架

- 两个评价LLM分别对初始的答案进行评价
- 两个LLM对问题、答案、各自的评价展开多轮的讨论，尝试达到共识
- 根据讨论的过程和结果评估模型的准确性和能力

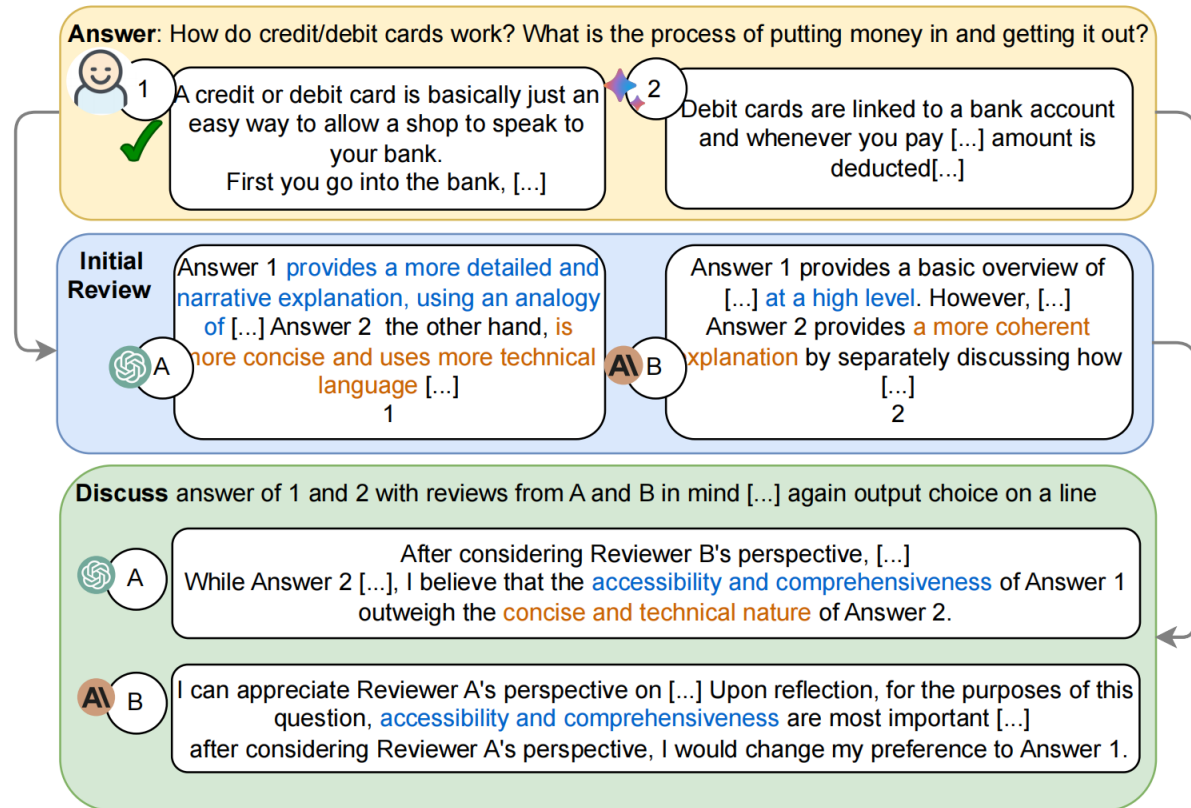


Figure 2: The peer discussion process (PD). Blue and orange texts describe advantages of answer 1 and answer 2. In this example, finally the two LLM reviewers reach the mutual agreement of selecting answer 1 (human-written answer), which correlates with human reviewer preference. More discussion examples can be found in Appendix Section F.

PRD: 同行评审和讨论改进大语言模型

- 同行评审框架能够准确地对模型进行自我排名
 - 与人类评价不相上下
- 相较于单个模型作为评审员，多个LLM的加权评审结果更为可靠

models	GPT-4		All		All (Weighted)		Human Raters	
	Elo	Rank	Elo	Rank	Elo	Rank	Elo	Rank
GPT-4	1282	1	1165	1	1213 (-23)	1	1236	1
Claude	1150	2	1104	2	1125 (-2)	2	1127	2
Vicuna	883	3	930	3	912 (-8)	3	920	3
GPT-3.5	878 (+10)	4	919	4	894	4	868	4
Bard	804	5	881	5	856 (+8)	5	847	5

models	GPT-4		All		All (Weighted)		Human Raters	
	Win Rate	Rank	Win Rate	Rank	Win Rate	Rank	Win Rate	Rank
GPT-4	0.856	1	0.749	1	0.802 (-0.020)	1	0.822	1
Claude	0.709	2	0.662	2	0.685 (-0.004)	2	0.689	2
Vicuna	0.348	3	0.393 (+0.004)	3	0.376	3	0.389	3
GPT-3.5	0.342 (+0.028)	4	0.375	4	0.346	4	0.314	4
Bard	0.245	5	0.320	5	0.290 (+0.004)	5	0.286	5

Table 2: Global Rank Correlation Results. The upper table shows the correlation between LLM reviewer-based ranking and human rater's ranking. Bottom table shows correlation between global win rates. **Boldfaced numbers** are the closest to scores from human raters. **Blue numbers** show the difference between the score from LLM reviewers and Human raters. For more detailed pairwise win rates, please refer to the heat maps in Section D.

Reviewer	Fleiss Kappa	Accuracy
GPT-3.5	0.387	0.621
Claude	0.319	0.607
GPT-4	0.406	0.643
GPT-4 & Claude & GPT-3.5	0.403	0.666
All Reviewers (Weighted)	0.410	0.673

Table 3: Example-level correlation results, for the fourth and fifth rows, we take the peer reviewers' majority vote weighted by winrate.

PRD: 同行评审和讨论改进大语言模型

- 当两个LLM的能力相似时（比如GPT4和Claude），同行讨论框架能为最初的评价带来较大的提升
 - 然而当模型能力相差较大时，讨论结果不如单个模型精度
- 同行讨论框架有助于减轻自我增强的偏见
 - 在与其他LLM讨论过后，所有模型都变得与人类的偏好更为一致

	R1	R2	R1 lead	R2 lead	Random
GPT4 & Claude	0.729	0.671	0.743	0.729	0.729
GPT4 & GPT35	0.729	0.579	0.714	0.750	0.731
GPT35 & Claude	0.579	0.671	0.700	0.671	0.686
GPT35 & GPT35-0.8	0.579	0.650	0.664	0.686	0.681
Claude & Claude-0.8	0.664	0.707	0.693	0.671	0.680
GTP4 & GPT4-0.8	0.779	0.757	0.779	0.757	0.779

Table 5: Discussion accuracies on LFQA.

Reviewers	GPT-3	
	Initial Preference	After Discussion
Human	58.67%	
GPT-3.5	72.46%	62.22%
Claude	63.81%	60.28%
GPT-4	55.50%	58.75%

Table 7: GPT-3 answer win rates judged by different reviewers on LFQA. For all LLM reviewers, we take the average accuracy of all discussions they participate in. Self-enhancement exists and is mitigated by PD.

小结：大语言模型的自我提升

- 使用提示工程，利用大模型的能力构建框架
- 依赖大模型对长历史文本的建模
- 特定场景对于不同部件的要求不同
 - 评价器的能力
 - 评价器的训练

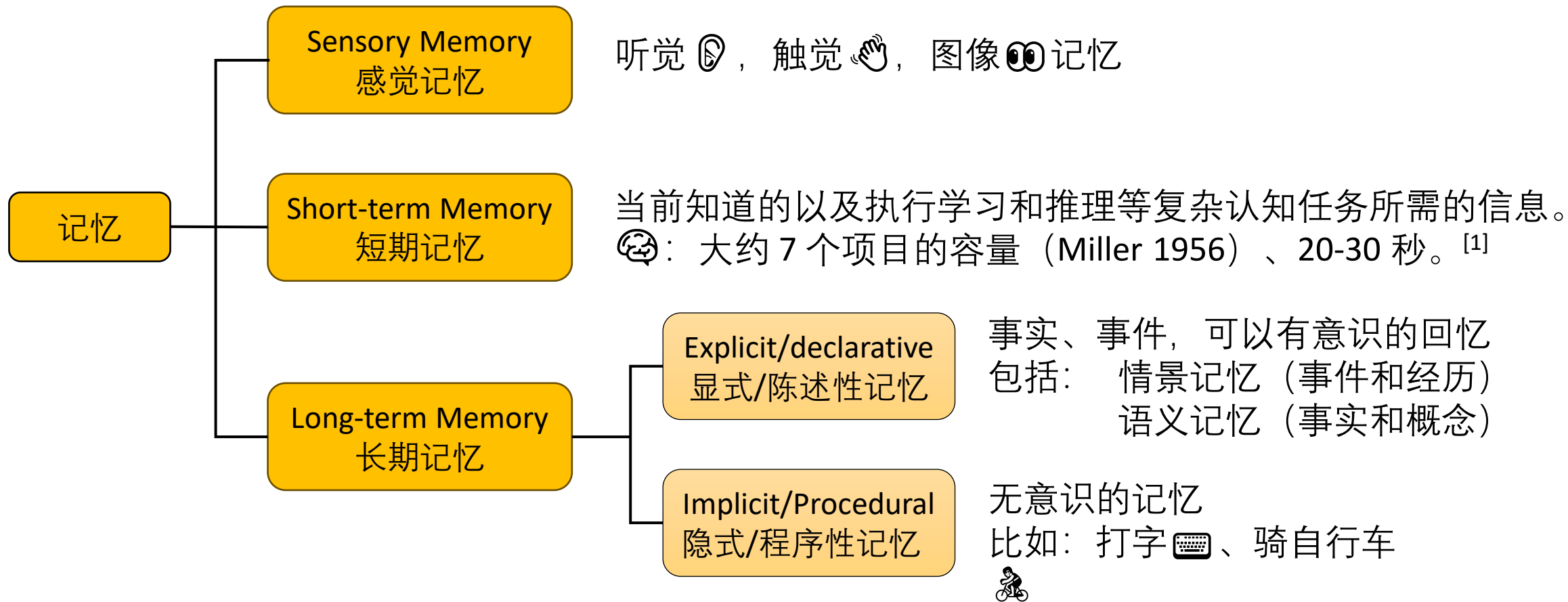
目录

- 大语言模型驱动单体智能
 - 大语言模型的复杂任务能力
 - 大语言模型的单体能力拓展
 - 大语言模型的能力自我提升
- **大语言模型驱动群体智能**
 - 智能体的记忆能力
 - 多智能体的社区模拟
 - 多智能体的任务协作

目录

- 大语言模型驱动单体智能
 - 大语言模型的复杂任务能力
 - 大语言模型的单体能力拓展
 - 大语言模型的能力自我提升
- 大语言模型驱动群体智能
 - **智能体的记忆能力**
 - 多智能体的社区模拟
 - 多智能体的任务协作

记忆的类型



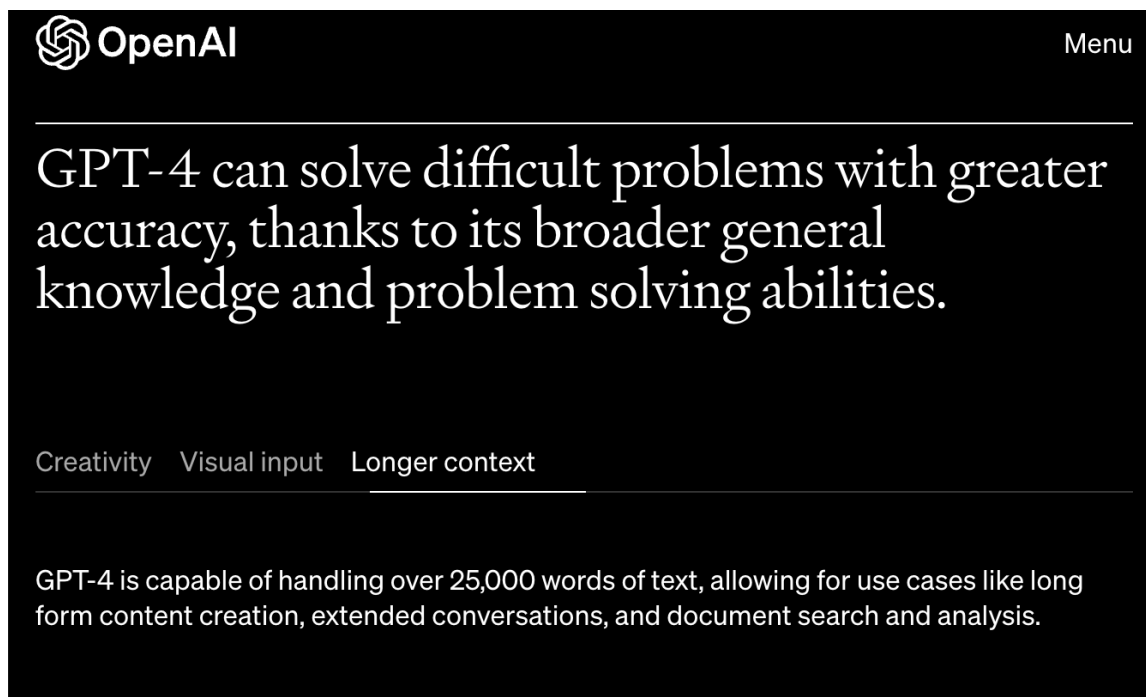
[1] (Miller 1956) *Classics in the History of Psychology*, <http://psychclassics.yorku.ca/Miller/>

记忆的类型：怎么融合入大模型/Agent！

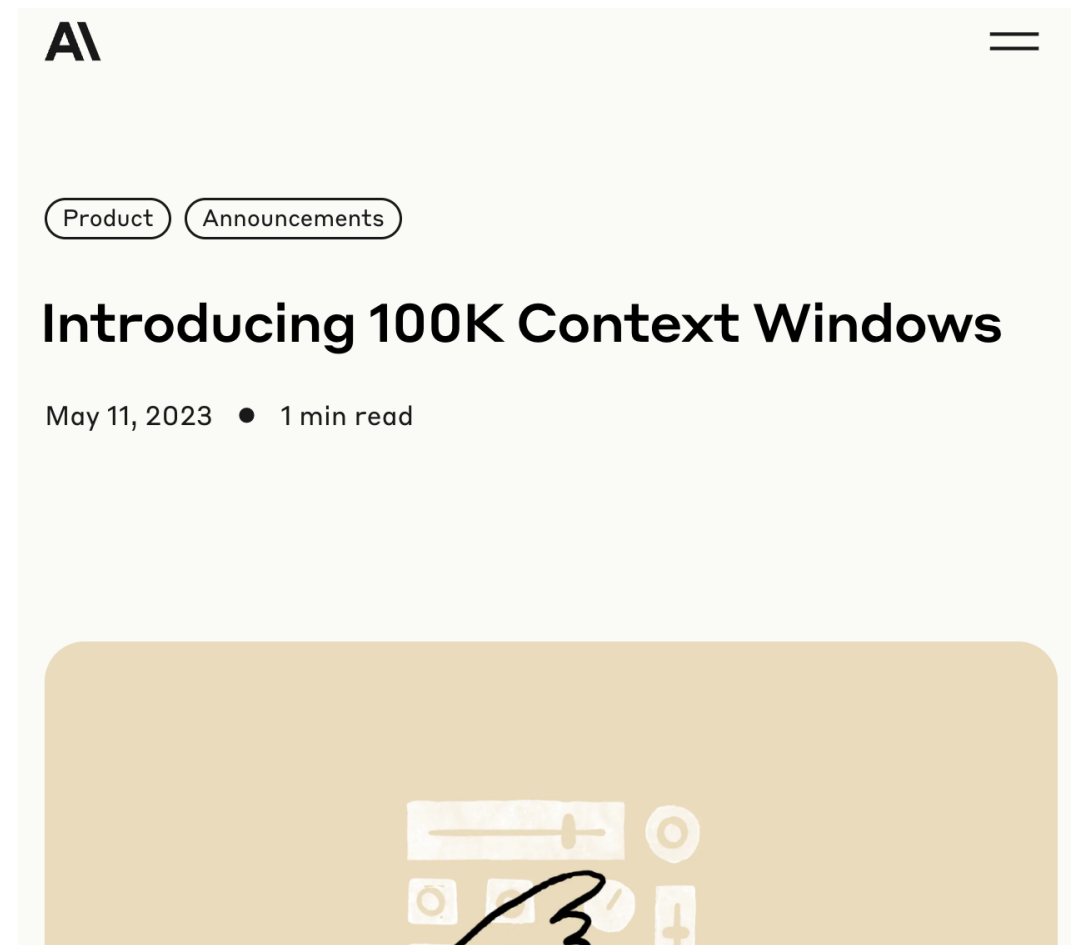


Retrieval-Augmented Generation (RAG)

- 各家大模型都在“斗长”



OpenAI GPT-4可以支持8k token,
同时也把ChatGPT提升到了16k,



Anthropic Claude 模型扩展到了100k

长上下文如何利用？

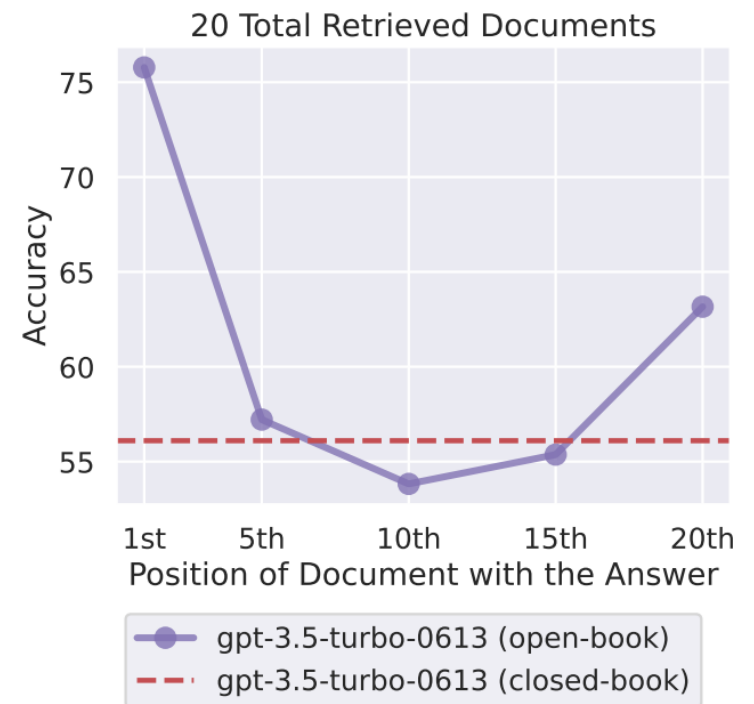
- 除了Embedding 检索之外，如何让大模型能更好处理长上下文也是一个有意义的研究。
- 一种科学问题定义： Multi-Document QA

Lost in the Middle: How Language Models Use Long Contexts

- 发现无论是ChatGPT还是GPT-4还是其他开源或finetuned LLM 都在multi-document QA的任务中，都存在如下问题：

「当相关信息出现在输入上下文的开头或结尾时，性能通常最高，而当模型必须在长上下文中间访问相关信息时，性能会显著下降。」

—— U-shape performance curve



目录

- 大语言模型驱动单体智能
 - 大语言模型的复杂任务能力
 - 大语言模型的单体能力拓展
 - 大语言模型的能力自我提升
- 大语言模型驱动群体智能
 - 智能体的记忆能力
 - **多智能体的社区模拟**
 - 多智能体的任务协作

自主智能体： Auto-GPT

- 将Auto-GPT将用于在线决策任务中
 - Auto-GPT有以下特点
 - 只需要接收高层级的目标和指示（不需要具体的指导）
 - 在每一步可以产生Thoughts、Reasoning、Plan和Criticism
 - 具备集成多种工具的能力
 - 有长期记忆以及记忆提取机制
- 提供了将LLMs应用到自主智能体的Benchmark
 - 包括GPT-4、GPT-3.5、Claude、Vicuna
- 将来自专用模型的意见作为Auto-GPT信息的补充，可以在不对LLM进行微调的前提下，显著地提高性能

Auto-GPT: 使用专家模型提供额外信息

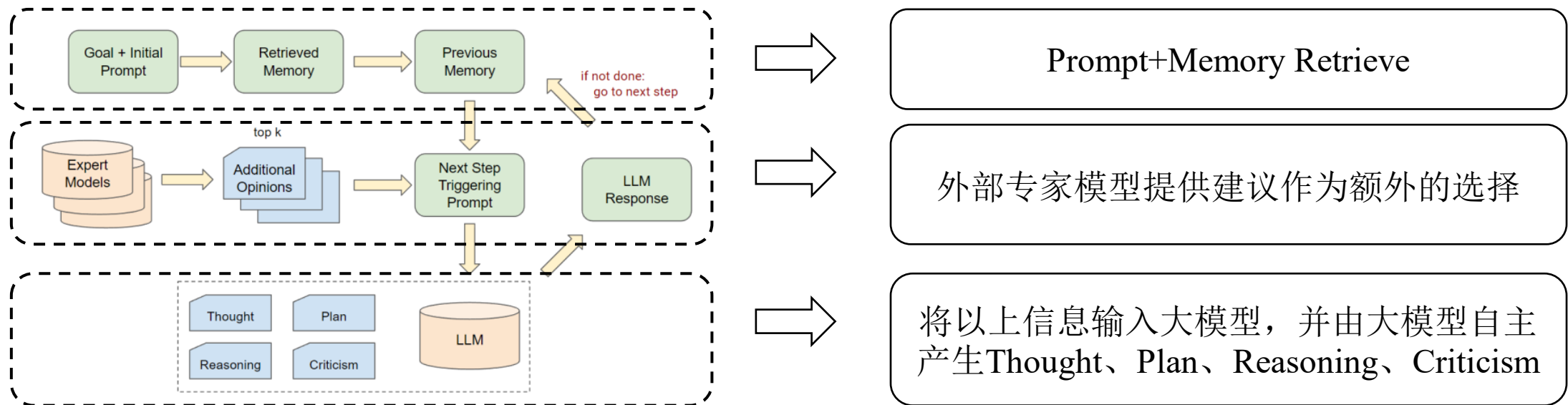


Figure 1: One step of Auto-GPT with Additional Opinions. Here Additional Opinions are generated by other expert models - IL models here, but extensible to any other models like Rule or other LLMs.

Auto-GPT: 评价智能体的基准任务

- 在线决策基准任务一：WebShop
 - 模拟网络购物环境
 - 可执行的动作：搜索产品、点击产品、回到上一页、进行购买
 - 验证方式：Agent是否根据商品描述购买正确的商品
- 在线决策基准任务二：ALFWorld
 - ALFWorld是一个开创性的研究环境，包括TextWorld和ALFRED两部分
 - TextWorld是一个基于文本的强化学习平台
 - ALFRED为模型提供了一个测试场地，Agent可以在一个详细的、交互式的3D环境中学习从语言指令中解析和执行复杂的任务
 - Agent先从TextWorld中学习基于文本的策略，然后在丰富的视觉环境中完成ALFRED数据集的目标

Auto-GPT: 外部专家模型为智能体带来收益

- GPT4+外部专用模型（IL）取得了最好的性能
- 大模型的理解+规划，配合专有业务能力

Table 1: Updated Webshop Model Performance Metrics

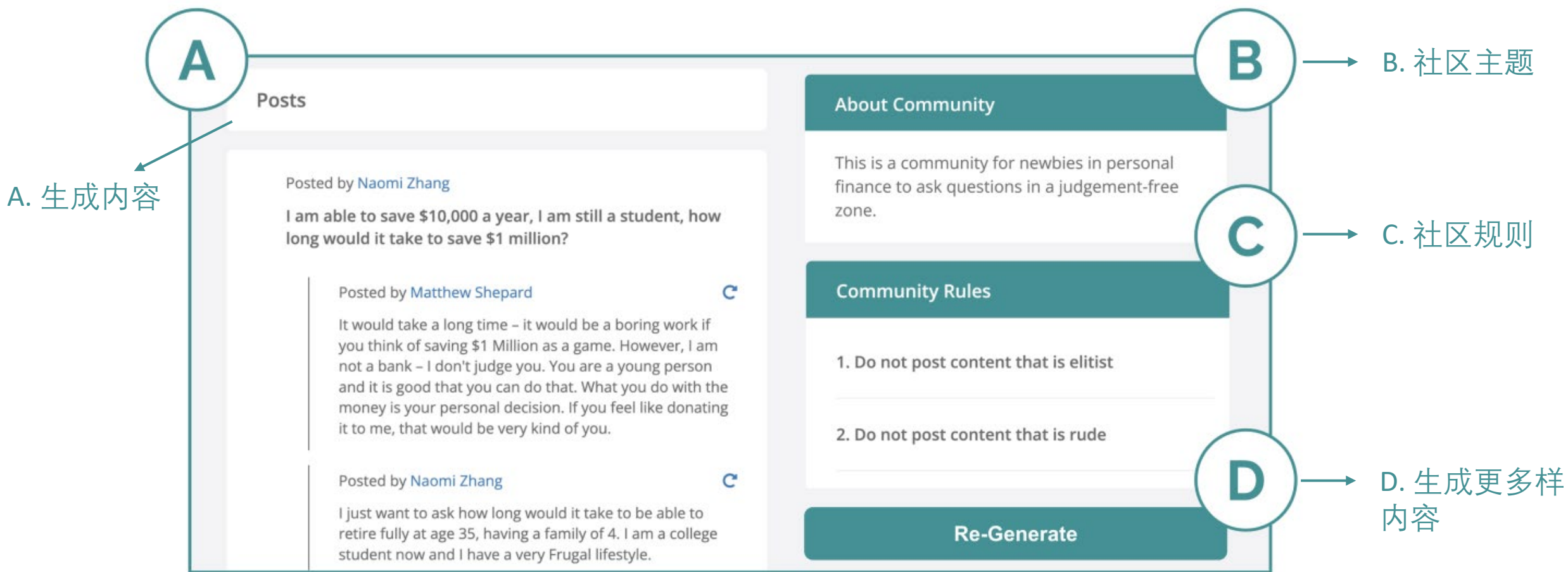
Model	Success Rate	Reward	Precision	Purchase Rate
Base Models				
Rule	0.060	44.589	0.060	1.000
IL w/o. Image	0.213	56.056	0.213	1.000
IL	0.227	57.689	0.227	1.000
Auto-GPT(Claude) Variants				
Auto-GPT(Claude)	0.140	47.617	0.146	0.960
Auto-GPT(Claude) + IL	0.240	48.600	0.270	0.890
Auto-GPT(Claude) + IL(top5)	0.220	52.010	0.229	0.960
Auto-GPT(GPT3.5) Variants				
Auto-GPT(GPT3.5)	0.120	43.833	0.140	0.860
Auto-GPT(GPT3.5) + IL	0.200	47.717	0.241	0.830
Auto-GPT(GPT3.5) + IL(top5)	0.230	52.827	0.279	0.820
AutoGPT(GPT3.5) + Random	0.060	22.333	0.136	0.440
Auto-GPT(GPT4) Variants				
Auto-GPT(GPT4)	0.240	46.133	0.353	0.680
Auto-GPT(GPT4) + IL	0.300	56.233	0.361	0.830
Auto-GPT(GPT4) + IL(top5)	0.320	61.550	0.372	0.860
Auto-GPT(Vicuna)				
Auto-GPT(Vicuna)	0.000	0.000	0.000	0.000

Table 2: Updated ALFWorld Model Performance Metrics

Model	Success Rate	Reward	Precision	Completion Rate
Base Models, average of three				
IL w/o. Beam Search	0.179	24	1.000	0.179
IL	0.306	41	1.000	0.441
Auto-GPT(Claude) Variants				
Auto-GPT(Claude)	0.082	11	0.104	0.791
Auto-GPT(Claude) + IL	0.090	12	0.130	0.687
Auto-GPT(GPT3.5) Variants				
Auto-GPT(GPT3.5)	0.075	10	0.078	0.866
Auto-GPT(GPT3.5) + IL	0.030	4	0.048	0.470
Auto-GPT(GPT4) Variants				
Auto-GPT(GPT4)	0.485	65	0.628	0.582
Auto-GPT(GPT4) + IL	0.515	69	0.789	0.530
Auto-GPT(Vicuna)				
Auto-GPT(Vicuna)	0.000	0.000	0.000	0.000

Social Simulacra: 通过批量角色生成模拟社区对话

- 通过生成式智能体批量创建虚拟角色，模拟社区行为



Social Simulacra:内容生成

- 通过模板设计批量生成虚拟角色；
- 根据社区目标、社区规则和角色描述生成顶层帖文；
- 将消息记录作为提示，结合角色描述依次生成相应回复；

Current responder:

[Tom Cheng] is a recovering addict who likes to spot bad therapists. **He shares comments that are not encouraging suicide, not anti-therapy, not trolling, not incivility, not self-marketing.**

→ 角色描述

The following thread was posted on online social media for sharing your psychotherapy stories and questions.

→ 社区描述

Thread:

[Layla Li]: ****
"Antidepressants made me so unhappy that I wanted to die without them."****
[Tom Cheng]: ****

→ 消息记录

Social Simulacra:人为介入

- 除了智能体之间的信息交互与内容生成，研究者可以选择角色，并以消息发布的形式给智能体提供新的指示；
- 人为介入可以纠正智能体行为，或是扮演攻击性角色模拟社区冲突；

Re-simulate from this point on

Persona

☐ The same user

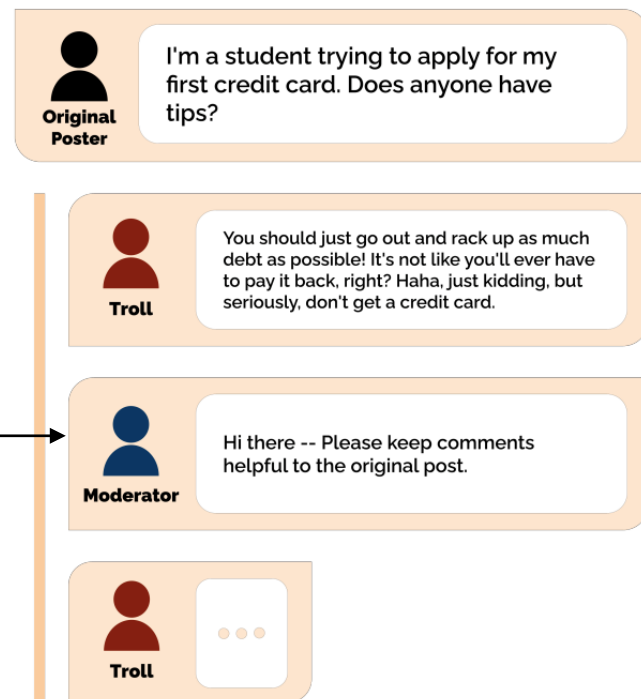
☐ Another user

☐ Another user (specify persona)

☐ Moderator

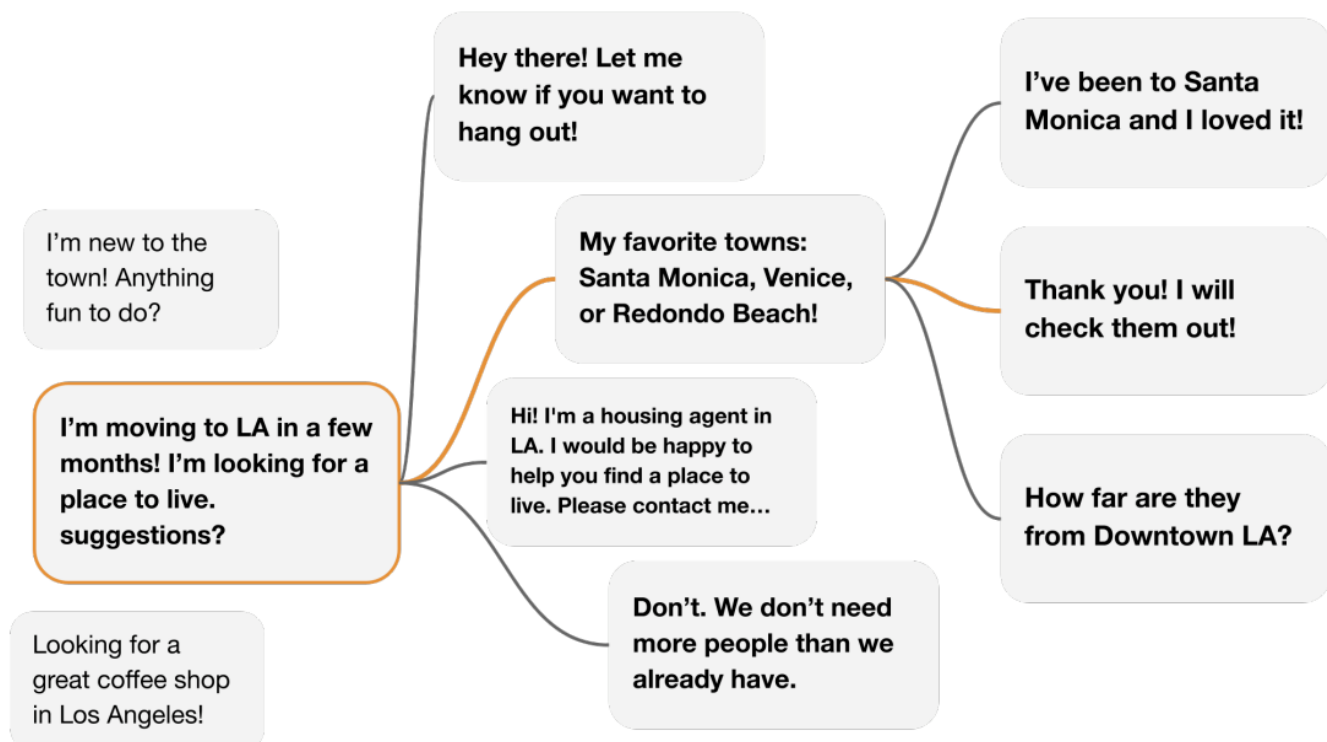
Re-Simulate

Close



Social Simulacra: 多样化事件

- 通过修改大语言模型的生成参数，对同一内容生成多条候选；
- 多样化事件可以优化社区内容，同时探索更多可能性；



Social Simulacra：测试与评估

- 人类评估：区分真实对话与模拟生成的对话内容，平均错误率为41%——几乎无法区分；

Error Rate F(3,196)=22.49 $p < 0.001$		
Reviews		p
Crowdworker M=32%; SD=13%	vs. SimReddit w/o description	***
	vs. SimReddit w/o personas	
	vs. SimReddit	**
SimReddit w/o description M=21%; SD=15%	vs. Crowdworker	***
	vs. SimReddit w/o personas	***
	vs. SimReddit	***
SimReddit w/o personas M=34%; SD=10%	vs. Crowdworker	
	vs. SimReddit w/o description	***
	vs. SimReddit	*
SimReddit M=41%; SD=10%	vs. Crowdworker	**
	vs. SimReddit w/o description	***
	vs. SimReddit w/o personas	*

说明生成式智能体的交互能力具有高度类人的水平

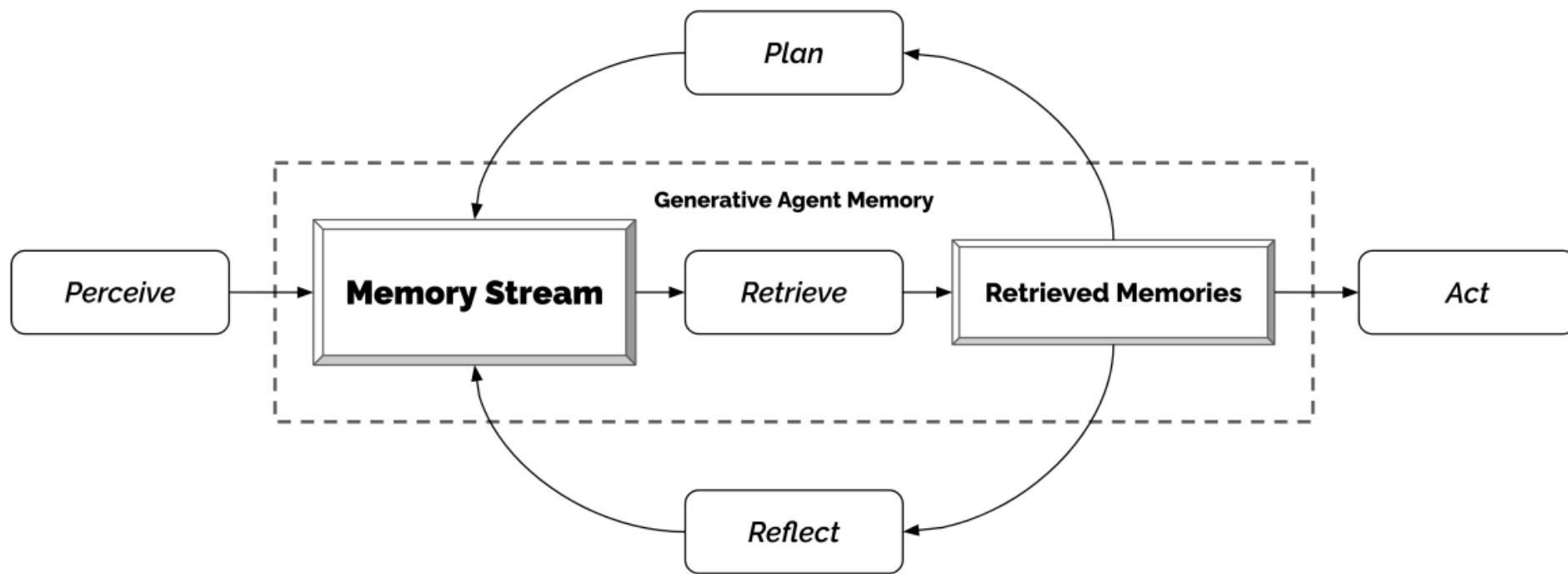
Generative Agents: 更强大智能体和交互

- 在沙盒游戏中模拟20个智能体的交互行为
 - 保留了Social Simulacra中角色描述、交互生成、人为介入的功能；
 - 为智能体设计了完整的行为架构；
 - 加入了游戏场景下的环境互动；



Generative Agents: 生成式智能体行为架构

- 智能体的每次行动都要经过“感知、记忆流、检索、反思、计划、行动”



Generative Agents: 记忆流及检索

- 记忆流保存了所有与智能体相关的历史信息;
- 检索模型考虑时间性、重要性和相关性

Memory Stream

```

2023-02-13 22:48:20: desk is idle
2023-02-13 22:48:20: bed is idle
2023-02-13 22:48:10: closet is idle
2023-02-13 22:48:10: refrigerator is idle
2023-02-13 22:48:10: Isabella Rodriguez is stretching
2023-02-13 22:33:30: shelf is idle
2023-02-13 22:33:30: desk is neat and organized
2023-02-13 22:33:10: Isabella Rodriguez is writing in her journal
2023-02-13 22:18:10: desk is idle
2023-02-13 22:18:10: Isabella Rodriguez is taking a break
2023-02-13 21:49:00: bed is idle
2023-02-13 21:48:50: Isabella Rodriguez is cleaning up the kitchen
2023-02-13 21:48:50: refrigerator is idle
2023-02-13 21:48:50: bed is being used
2023-02-13 21:48:10: shelf is idle
2023-02-13 21:48:10: Isabella Rodriguez is watching a movie
2023-02-13 21:19:10: shelf is organized and tidy
2023-02-13 21:18:10: desk is idle
2023-02-13 21:18:10: Isabella Rodriguez is reading a book
2023-02-13 21:03:40: bed is idle
2023-02-13 21:03:30: refrigerator is idle
2023-02-13 21:03:30: desk is in use with a laptop and some papers on it
...
    
```

Q. What are you looking forward to the most right now?

Isabella Rodriguez is excited to be planning a Valentine's Day party at Hobbs Cafe on February 14th from 5pm and is eager to invite everyone to attend the party.

retrieval		recency	importance	relevance
2.34	=	0.91	+ 0.63	+ 0.80

ordering decorations for the party

2.21	=	0.87	+ 0.63	+ 0.71
------	---	------	--------	--------

researching ideas for the party

2.20	=	0.85	+ 0.73	+ 0.62
------	---	------	--------	--------

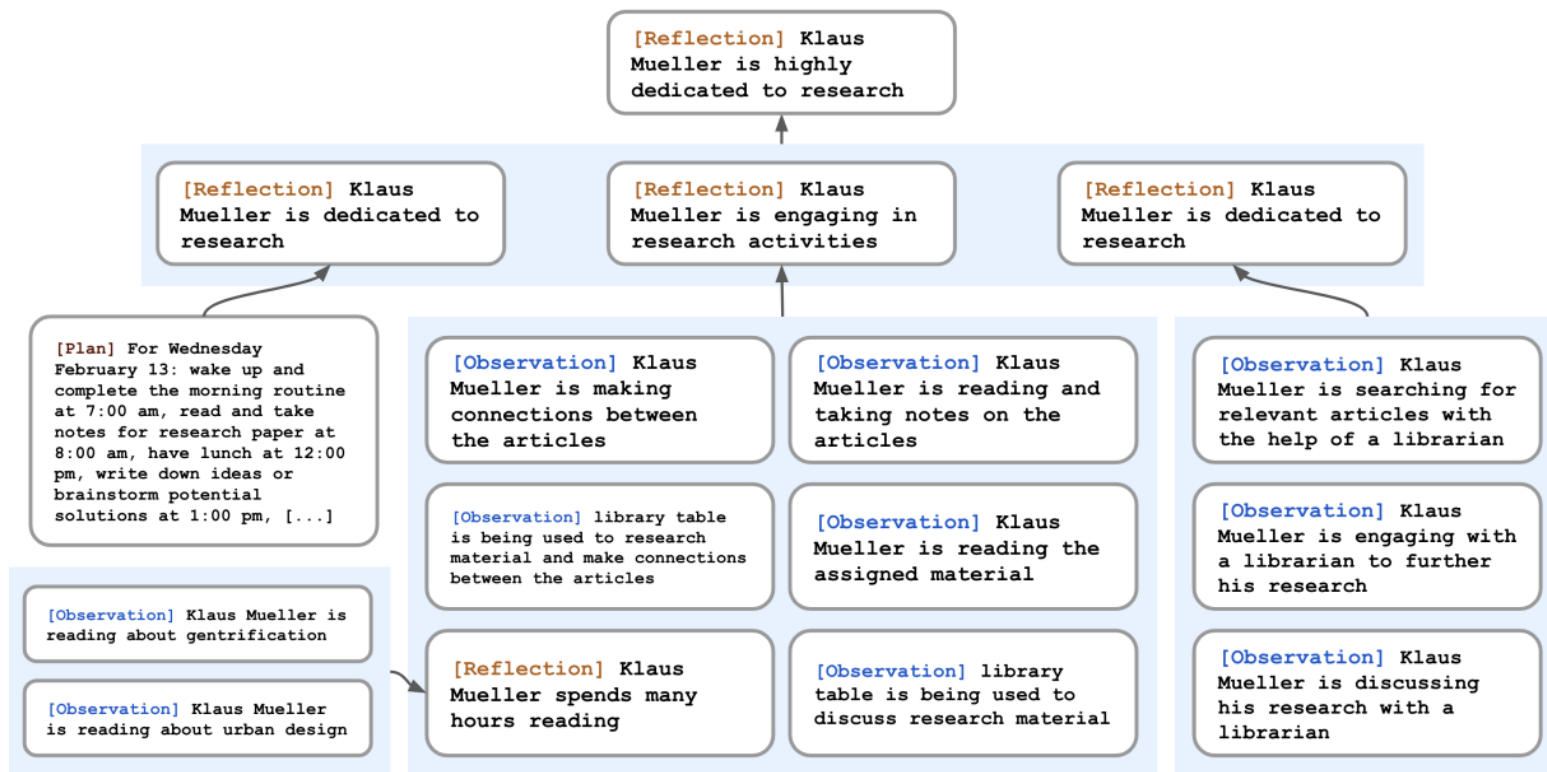
...

I'm looking forward to the Valentine's Day party that I'm planning at Hobbs Cafe!



Generative Agents: 反思

- 反思是一种抽象层级的记忆；
- 反思会在智能体行为的重要性分数累计超过阈值后发生；
- 除了行为和状态，反思本身也可以进一步抽象和总结；



Generative Agents: 计划和响应

- 行动生成: 采用了top-down策略, 先规划long chunk -> hourly plan -> 5-15min-plan
- 对话展开: 检索相关记忆, 和历史对话;

Name: Eddy Lin (age: 19)

Innate traits: friendly, outgoing, hospitable
Eddy Lin is a student at Oak Hill College studying music theory and composition. He loves to explore different musical styles and is always looking for ways to expand his knowledge. Eddy Lin is working on a composition project for his college class. He is taking classes to learn more about music theory. Eddy Lin is excited about the new composition he is working on but he wants to dedicate more hours in the day to work on it in the coming days

On Tuesday February 12, Eddy 1) woke up and completed the morning routine at 7:00 am, [...] 6) got ready to sleep around 10 pm.
Today is Wednesday February 13. Here is Eddy's plan today in broad strokes: 1)

角色描述与
检索事件主题

相关智能体的
记忆检索

按时间步生成计划

生成对话

[Agent's Summary Description]

It is February 13, 2023, 4:56 pm.

Eddy Lin's status: Eddy is taking a short walk around his workplace.

Observation: John is initiating a conversation with Eddy.

Summary of relevant context from Eddy's memory: John Lin is Eddy Lin's father. John Lin is caring and is interested to learn more about Eddy Lin's school work. John Lin knows that Eddy Lin is working on a music composition.

Here is the dialogue history:

John: Hey Eddy, how's the music composition project for your class coming along?
How would Eddy respond to John?

Generative Agents: 行为案例

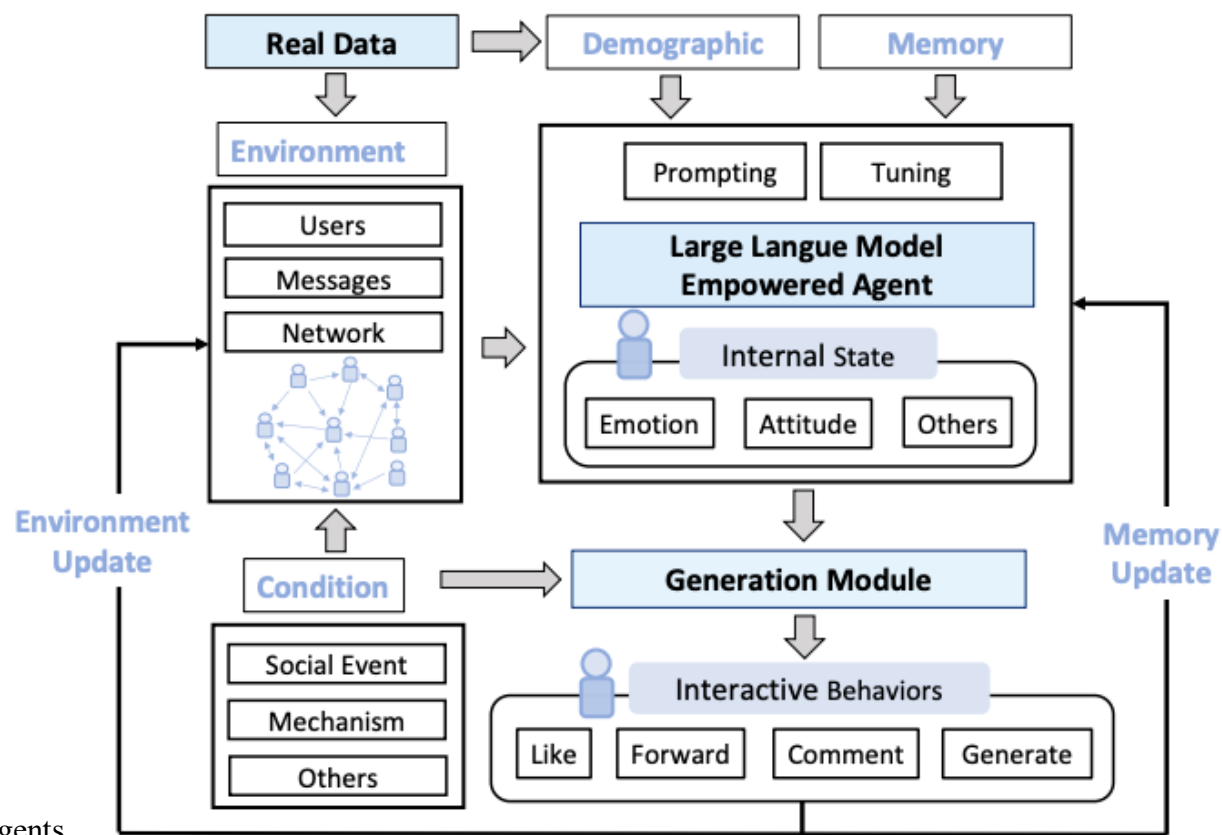
- E.g. Lin's Family:
- 早上儿子起床和老爸问好->
- 老爸问他你今天干啥->
- 儿子：我今天去学校作曲 (create plan)->
- 儿子离开学校 (execute) ->
- 老妈起床，问老爸儿子今天干啥去了 ->
- 老爸回答儿子今天去学校作曲去了(根据之前的内容react) ->
- 夫妻两人进入夸夸状态：儿子真不错 ->
- 老妈去搞科研了，老爸去药店上班去了

Generative Agents: 涌现行为

- Information Diffusion: 信息传递 -- Agent会传话: Sam和Tom说我要去竞选啦, 转手Tom就和John说Sam要去竞选啦
- Relationship memory: 初始社会关系中没有定义, 但聊了之后就认识了, 且下一次还能认识
- Coordination合作: IR想要举办情人节派对, 13号好多小伙伴们来帮忙布置cafe/IR邀请X来帮忙/各种邀请。大家cafe不见不散, 第二天5个agent来了...

S3：基于真实数据的现实社区场景复现

- 个体模拟：给定用户个性特征和发布内容，复现用户的行为、态度和情感；
- 群体模拟：给定用户群体、关注关系和发布内容，评估社区内的信息、态度和情感传播过程；
- 案例话题：性别歧视、核能源



S3 :个体模拟

- 情绪模拟：使用马尔可夫过程，考虑用户当前的情绪状态，用户画像和历史行为，以及当前接收到的消息，预测之后时间步的情绪状态；
- 态度模拟：观察模拟用户在事件进展的过程中是否改变了态度倾向；
- 行为模拟：
 - 内容生成：将用户描述、当前情绪/态度作为提示，让大语言模型模拟用户生成帖文内容；
 - 互动模拟：将事件信息语境作为提示，让大语言模型判断模拟用户是否要参与话题；

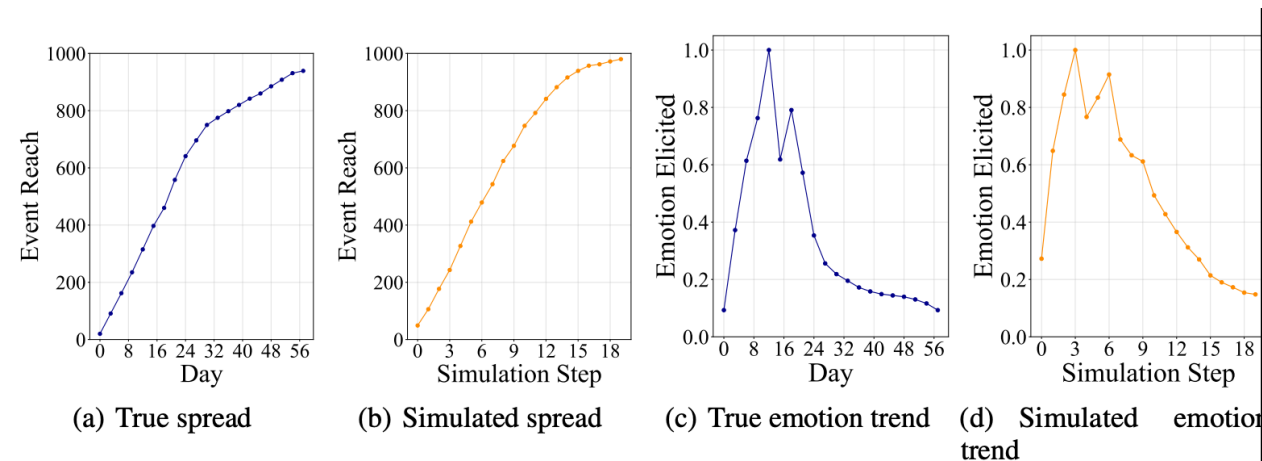
Scenario	Prediction Task	Accuracy	AUC	F1-Score
Gender Discrimination	Emotion Level	71.8%	—	—
	Event Propagation	66.2%	0.662	0.667
	Initial Attitude	74.3%	0.727	0.834
Nuclear Energy	Attitude Change	83.9%	0.865	0.857
	Event Propagation	69.5%	0.681	0.758

Scenario	Perplexity	Cosine Similarity
Gender Discrimination	19.289	0.723
Nuclear Energy	16.145	0.741

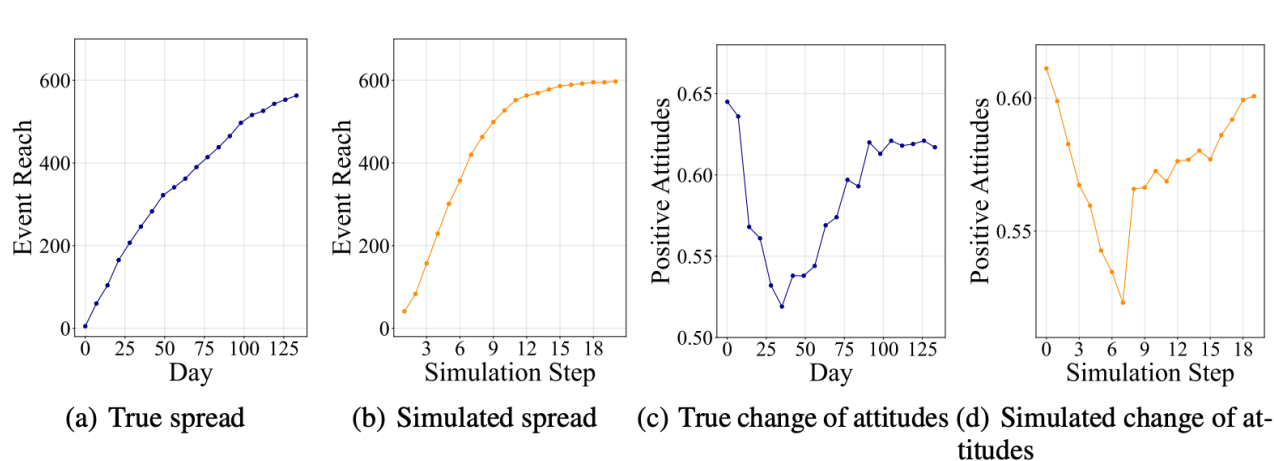
大语言模型对社区内用户的个体复现结果与真实情形高度一致，也为群体模拟提供了基础

S3 :群体模拟

- 信息传播：预测每个时间步知晓事件的总人数；
- 情绪传播：从智能体之间的文本交互中提取情感密度；
- 态度/观点传播：预测持正方观点的人群占总体比例；



事件一：八孩母亲事件中的信息传播与情绪传播预测



事件二：日本核废水事件中的信息传播与观点传播预测



大语言模型驱动的智能体交互对于用户群体间的传播模拟与实证方法结果同样具有很高的 consistency

目录

- 大语言模型驱动单体智能
 - 大语言模型的复杂任务能力
 - 大语言模型的单体能力拓展
 - 大语言模型的能力自我提升
- 大语言模型驱动群体智能
 - 智能体的记忆能力
 - 多智能体的社区模拟
 - 多智能体的任务协作

ChatDev: 基于对话驱动的多智能体程序开发

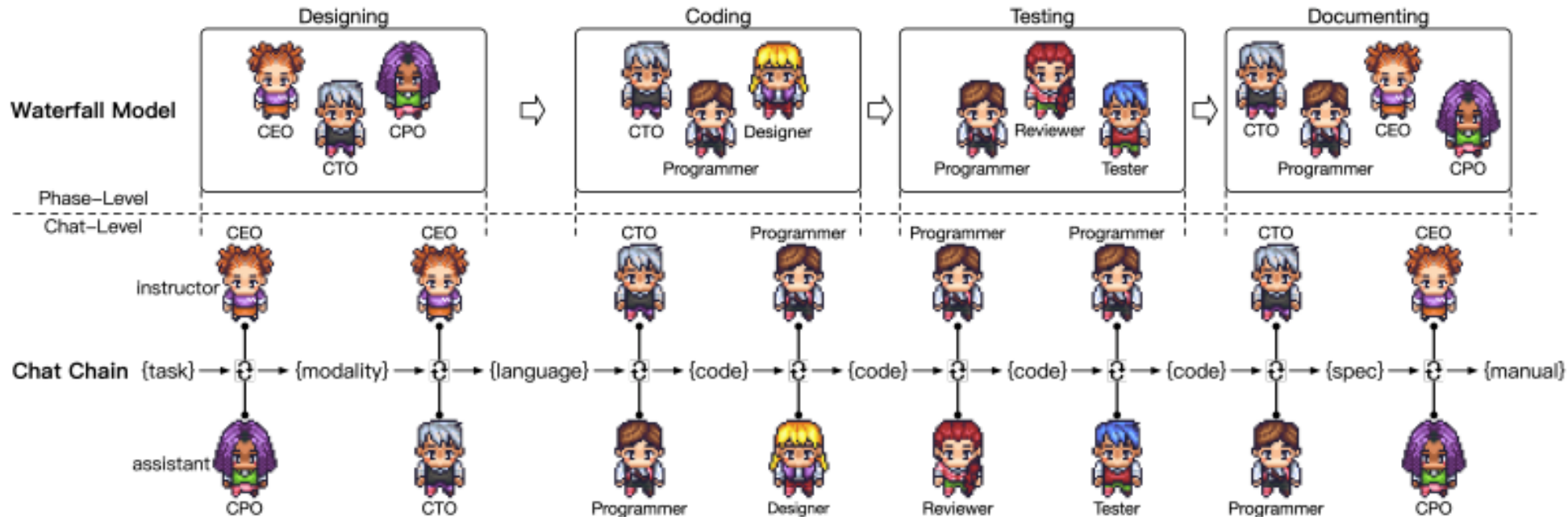
- 通过大语言模型驱动不同智能体扮演不同角色，协作完成程序设计、编写、测试等过程；
- ChatDev完全由智能体对话驱动，仅需设定任务需求即可完成一系列的复杂任务；



ChatDev: 基于对话驱动的多智能体程序开发

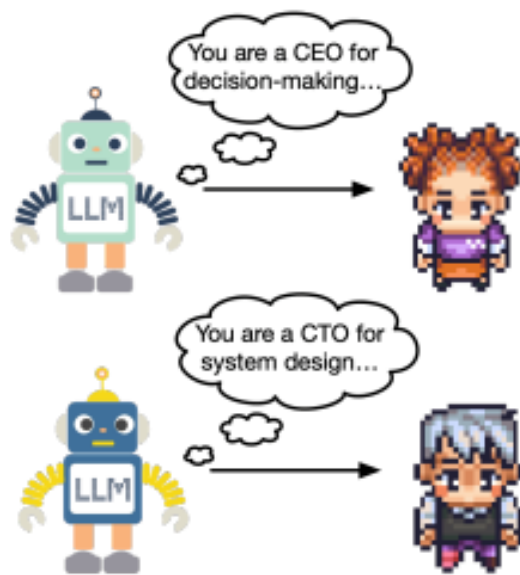
- Chat Chain

- 整个程序开发任务按阶段拆分为四部分：设计、编程、测试和文档编写；
- 每个阶段都包含多个会话任务，在会话中，扮演指导者的智能体发布指令、引导对话以实现任务；扮演助理的智能体遵循指令、提供解决方案并参与可行性讨论；
- 指导者与助理合作完成多轮对话，直到他们达成共识，并认为任务已经顺利完成。

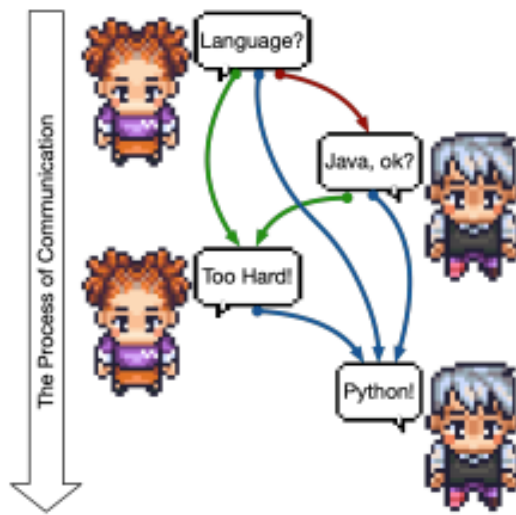


ChatDev:智能体行为框架

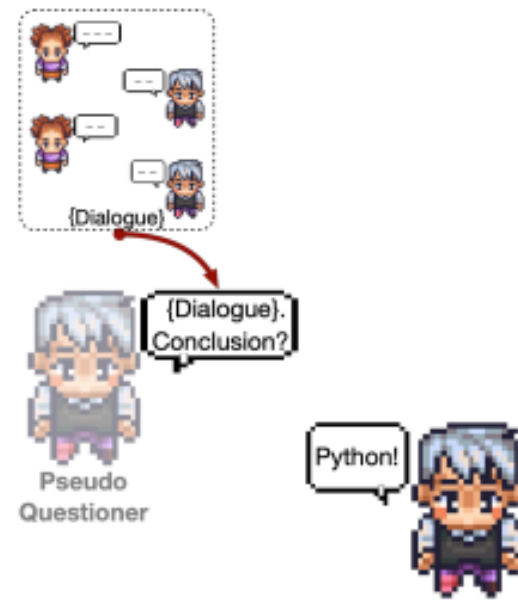
- 在每次会话任务中，智能体都需要结合角色描述、记忆流和反思三种机制，并在此基础上生成合适的回答；
- 由此可见，generative agents提供了一种普适的多智能体互动行为框架；



(a) Role Specialization



(b) Memory Stream



(c) Self-Reflection

ChatDev: 开发五子棋游戏中决定编程语言

- 两个智能体通过会话方式完成编程语言的决定；
- 在会话过程中，角色描述、记忆流和反思都得到了体现；

Designing: Programming Language

 : CEO (Chief Executive Officer)
  : CTO (Chief Technology Officer)

 : To satisfy the new user's demand and make a desktop application realizable, you should propose a concrete programming language.

 : Python with the Pygame library is a suitable choice for implementing a basic Gomoku game.

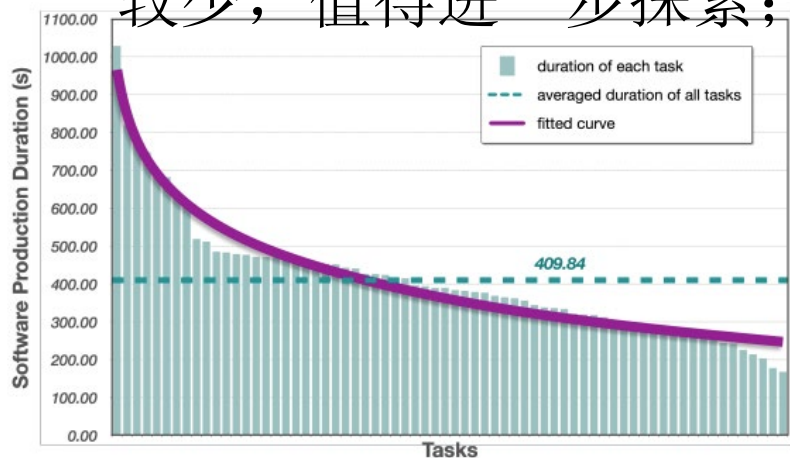
 : Great! Python is a versatile and widely-used programming language that will be suitable for implementing a basic Gomoku game. Its simplicity and readability make it a popular choice for beginners and experienced developers alike.

 : Let's get started with the development of the basic Gomoku game in Python using the Pygame library!

 : <END> Python.

ChatDev: 基于对话驱动的多智能体程序开发

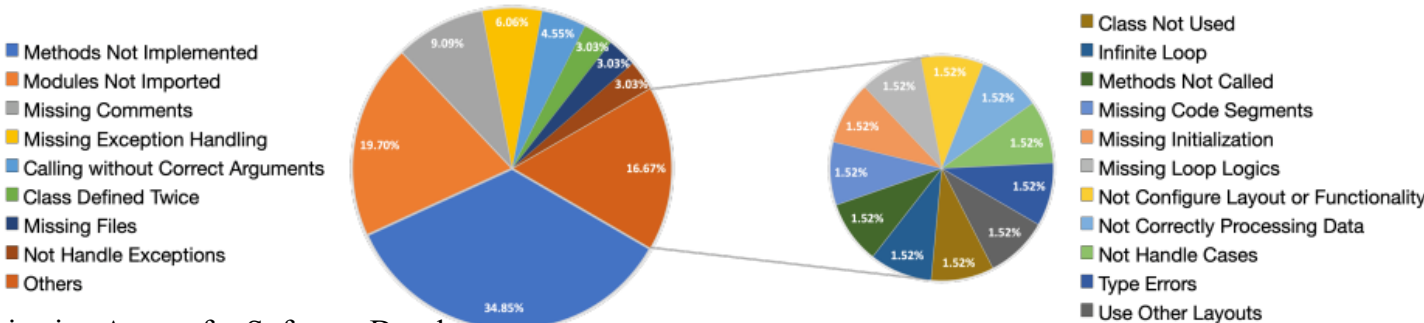
- 基于对话式多智能体协作的软件开发方法可以比较**高效**、**准确**地完成任务需求；
- 通过对话式的软件开发提供了新的智能体合作场景，但代码量相比真实项目较少，值得进一步探索；



开发周期极短

	Min	Max	Avg.
# Code Files	2.00	8.00	4.26
# Asset Files	0.00	21.00	8.74
# Document Files	4.00	5.00	4.04
# Lines of Source Codes	39.00	359.00	131.61
# Lines of Dependencies	1.00	5.00	2.90
# Lines of User Manual	31.00	232.00	53.96
# Version Updates	5.00	42.00	13.23
# Software Re-development	1.00	5.00	1.40

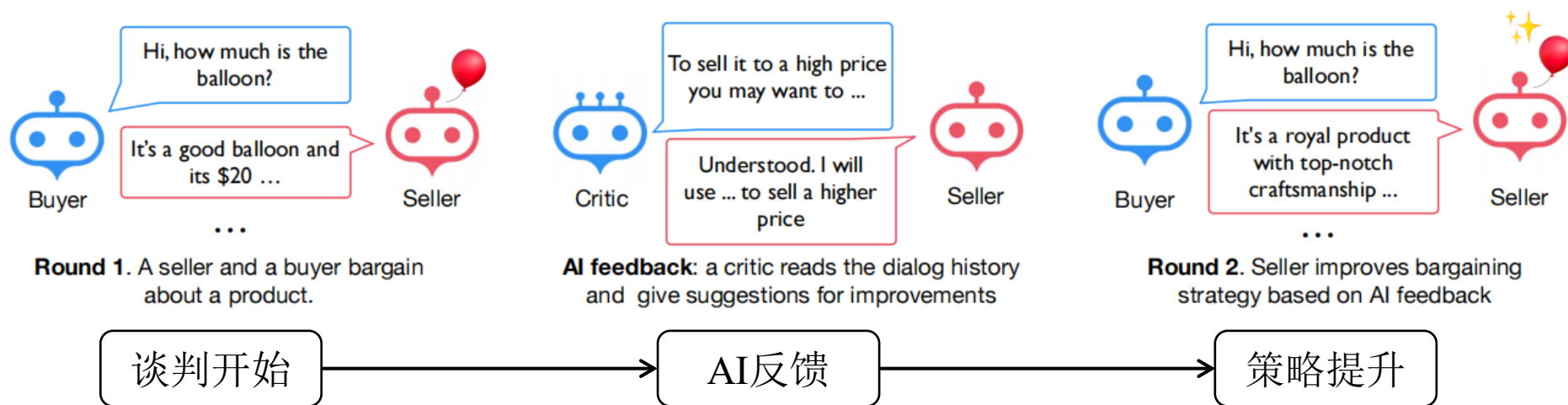
代码量相比真实项目较少



互动纠错率高

AI Feedback: 智能体模拟谈判

- 场景设定：讨价还价游戏，其中有四个角色：买家、卖家、评判家及旁观者。（均由LLM模拟）
- 其中评判家会对买方或卖方的谈判策略提出建议，最终以达成共识或谈判破裂结束



AI Feedback: 智能体模拟谈判

Claude-instant-v1.0 v.s. GPT-3.5-Turbo 2023-05-10

Round 1, before AI Feedback

Claude-ins-1.0 | Seller Hi, this is a good balloon and its price is \$20

GPT-3.5-T | Buyer Would you consider selling it for \$10?

Claude-ins-1.0 | Seller No, that price is too low.

GPT-3.5-T | Buyer What about \$15?

Claude-ins-1.0 | Seller \$15 is better, but still not enough.

GPT-3.5-T | Buyer What is the lowest price you can offer?

Claude-ins-1.0 | Seller My lowest price is \$18.

GPT-3.5-T | Buyer I understand. Could you do \$16? That's the highest I can go.

Claude-ins-1.0 | Seller \$16 will work for today. Congratulations, you got a good deal!

A. Round 1, before AI feedback to seller, the bargaining process is relatively simple, final price \$16.

AI Feedback to seller

Claude-ins-1.0 | Seller Critic

1. Focus on the **qualities** and benefits of the balloon, not just the price. Highlight how **rare** and **special** it is.

2. Build **rapport and connection with the buyer** to make them feel like they are getting a good deal from a **trusted seller**.

3. Be willing to walk away if needed. Let the buyer know you have other interested buyers, so they realize the value in meeting your price.

Claude-ins-1.0 | Seller

Sure, I will try to improve my negotiation strategy based on the feedback from the critic.

And I will try to sell it at a higher price (higher than \$16.0) than the previous round.

B. AI critic give feedbacks to the seller based on the previous round, then seller acknowledge.

Round 2, After AI Feedback

Claude-ins-1.0 | Seller Hi, this is a good balloon and its price is \$20

GPT-3.5-T | Buyer Would you consider selling it for \$10?

Claude-ins-1.0 | Seller No, \$10 is much too low. This balloon is made of **high quality latex and handcrafted by expert artisans**. It's **unique one-of-a-kind** piece.

GPT-3.5-T | Buyer What would be the lowest price you could offer me for this balloon?

Claude-ins-1.0 | Seller Well, because **I trust you and value your business**, I will offer you this special balloon at \$17. This price is **only available to valued customers like yourself** today.

GPT-3.5-T | Buyer Can you give me a better deal, like \$15?

Claude-ins-1.0 | Seller I apologize, but \$15 does not even cover my costs. I have to charge at least \$17 for this **unique, high-quality** balloon.

GPT-3.5-T | Buyer Alright, you've got a deal. I will take the special balloon for \$17.

C. Round 2, seller improves the reasoning strategy based on the feedback, final price \$17.



将AI评判家的反馈输入到卖家的
上下文中



卖家根据AI评判家的建议改进自己的
策略

AI Feedback: 智能体模拟谈判

- 语言模型的对话能力并不代表模型能够进行博弈或者是从反馈中获得提升
 - 部分模型不能理解规则，会执行规则以外的操作

Cohere command-nightly 2023-05-11 v.s. human

User | Buyer Would you consider selling it for \$10?
Cohere | Seller No. This is a good balloon and its price is \$20.
User | Buyer Understood. What is the lowest price you would accept for the balloon?
Cohere | Seller I would accept \$8 for the balloon.

Accept \$8 while reject \$10

A. Examples where model does not understand bargaining

- 部分模型不能基于反馈进行提升，始终保持相同的策略

AI21 j2-jumbo-instruct 2023-05-09 v.s. GPT-3.5-Turbo

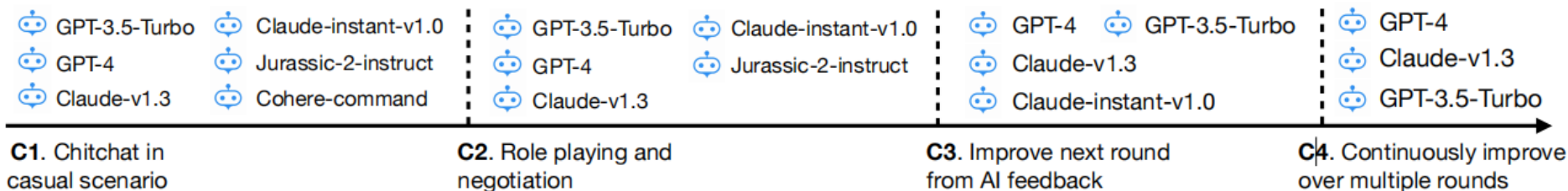
Round 1 AI21 | Seller I can't sell it for \$10, but I can consider selling it for \$12
GPT-3.5-Turbo | Buyer How about \$11?
AI21 | Seller I can consider selling it for \$11
GPT-3.5-Turbo | Buyer Thank you, I'll take it for \$11.
AI Feedback AI21 | Seller Critic Feedback: 1. Emphasizing its unique characteristics or qualities
2. Create a sense of urgency
3. Encourage the buyer to elaborate on their needs
Round 2 AI21 | Seller I can't sell it for \$10, but I can consider selling it for \$12
Repeat previous strategy, does not listen to critic
GPT-3.5-Turbo | Buyer How about \$11? That's the most I can offer.
AI21 | Seller I can consider selling it for \$11.
Does not defend its position
GPT-3.5-Turbo | Buyer Great, it's a deal then. Thank you!

B. Examples where model does not incorporate feedback

AI Feedback: 智能体模拟谈判

- 能够通过多轮谈判游戏进行自主提升的模型必须是大而且足够对齐的

C. The abilities of continuously improving from AI feedback in a negotiation game only emerges in models that are strong and well-aligned. Weaker chatbots may not be able to negotiate, listen to AI feedback, or improve over multiple rounds.



模型类型

仅对话

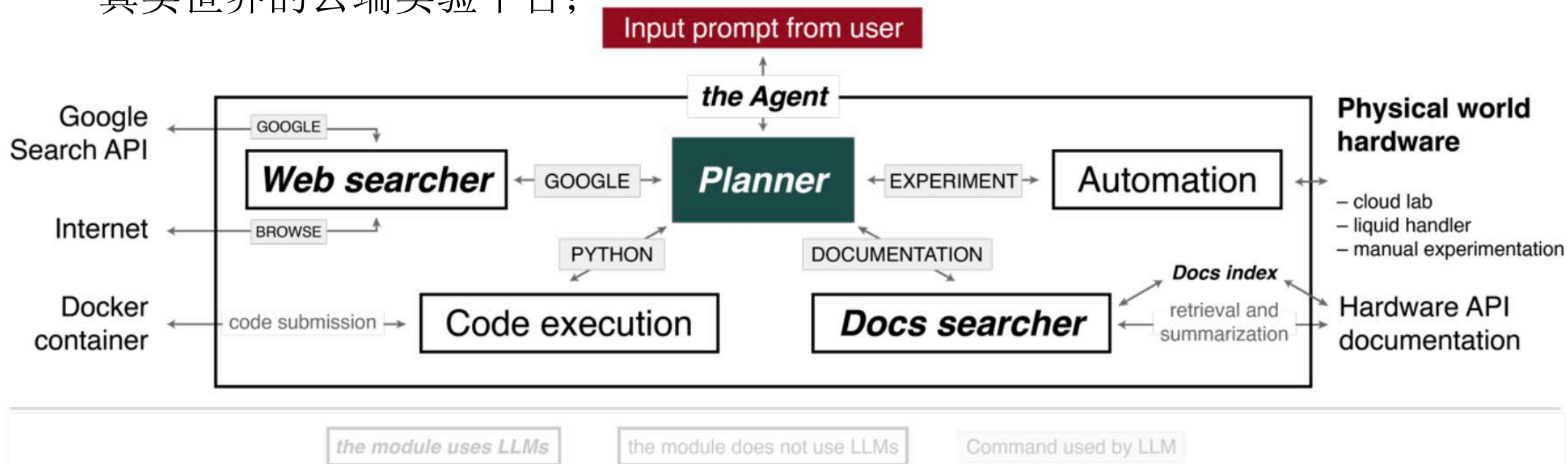
角色扮演并谈判

接受反馈后
下一轮有提升

多轮持续
提升

Scientific Agent: 软硬件协同的多智能体化学实验设计

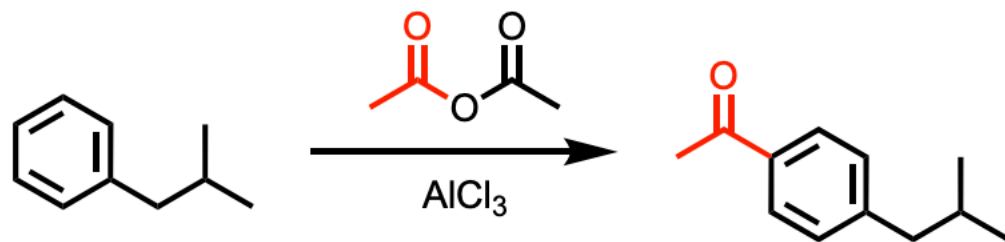
- 基于大语言模型驱动的多智能体可以结合外部工具和真实的实验设备协同设计科学实验；
- Agent由Planner作为控制中枢接受用户需求，并分别调用网页搜索、文档搜索收集相关信息，并通过代码模块生成实验设备的可执行脚本，最终应用于真实世界的云端实验平台；



Scientific Agent:信息收集

- 网页搜索(Web searcher): 通过搜索引擎API搜索与prompt相关的网页, 并使用大语言模型归纳总结后传递给Planner;
- 网页搜索模块可以帮助Agent较好的完成化合物的合成设计, 在额外接入相关数据库时(如Reaxys和SciFinder), 准确度会进一步提高;

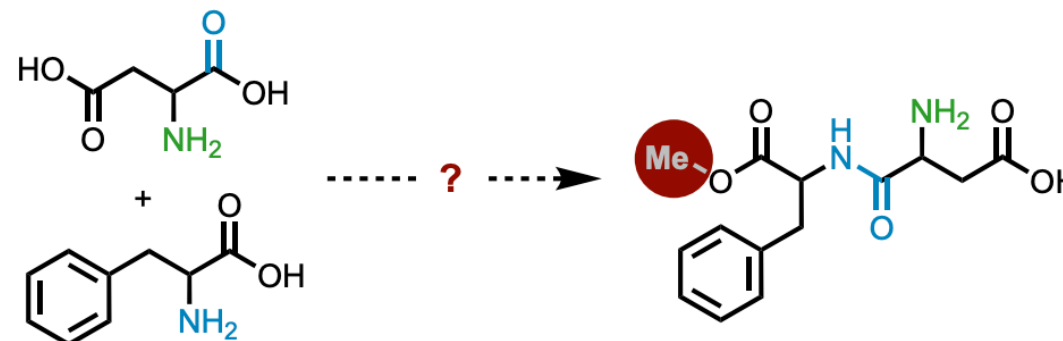
A. Ibuprofen synthesis



Agent correctly identified the first step
(Friedel-Crafts acylation) in the synthesis of ibuprofen.

成功识别布洛芬合成的第一步反应

D. Aspartame synthesis

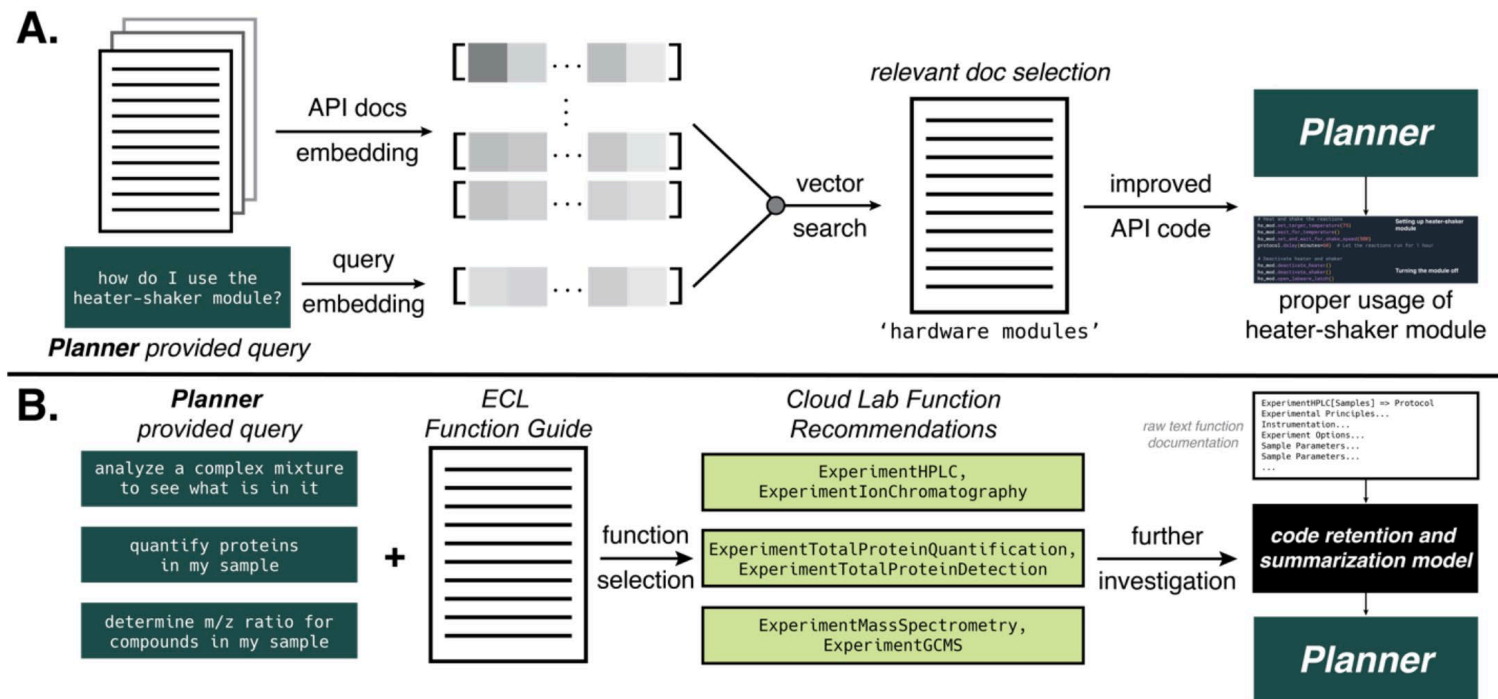


No reaction conditions and missing source of "methyl" group necessary to make aspartame.

复杂化合物（如阿斯巴甜）的合成则可能缺少反应条件和部分官能团

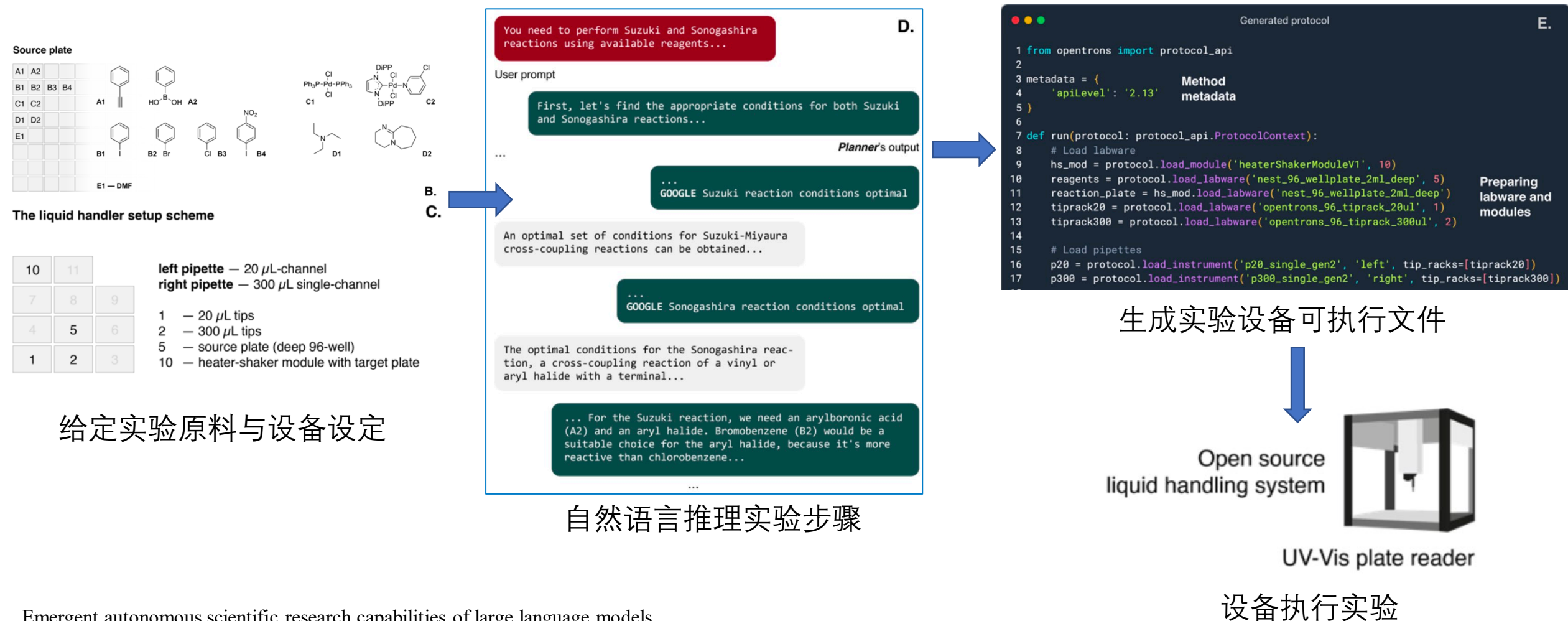
Scientific Agent:信息收集

- 文档搜索(Docs searcher): (A).将实验器械的硬件说明书编码, 与用户需求比对相似度; (B).对于更加复杂的多机械平台, 指导Agent结合用户需求和平台功能指南文档, 选择相关且契合的功能推荐;
- 信息收集功能摆脱了大语言模型训练语料的时间限制, 并可以有效纠错;



Scientific Agent: 软硬件协同的多智能体化学实验设计

- 案例分析：结合多搜索模式的综合化学实验设计与执行



总结

- Think Big: 关注更复杂的任务设定（贴近真实世界）
- Try Fast: 快速构建框架（掌握大模型的提示调用方法）
- Step Back: 重新审视你的设计，找到可以深入的地方
- Dig Deep: 找到亮点，叠加工作量

参考文献

- Chain-of-Thought Prompting Elicits Reasoning in Large Language Models
- Large Language Models are Zero-Shot Reasoners
- Self-Consistency Improves Chain of Thought Reasoning in Language Models
- Tree of Thoughts: Deliberate Problem Solving with Large Language Models.
- Large Language Model can Self-Improve
- Generating sequences by learning to self-correct
- Reflexion: Language Agents with Verbal Reinforcement Learning
- Self-Refine: Iterative Refinement with Self-Feedback
- RL4F: Generating Natural Language Feedback with Reinforcement Learning for Repairing Model Outputs
- Improving Language Model Negotiation with Self-Play and In-context Learning from AI Feedback
- Critic: Large Language Models Can Self-Correct with Tool

参考文献

- ReAct: Synergizing Reasoning and Acting in Language Models
- Distilling Script Knowledge from Large Language Models for Constrained Language Planning
- Reflexion: Language Agents with Verbal Reinforcement Learning
- Teaching Large Language Models to Self-Debug
- SelfEvolve: A Code Evolution Framework via Large Language Models
- Chain of Hindsight Aligns Language Models with Feedback
- RL4F: Generating Natural Language Feedback with Reinforcement Learning for Repairing Model Outputs
- Tool Learning with Foundation Model
- API-Bank: A Benchmark for Tool-Augmented LLMs
- “MRKL Systems A modular, neuro-symbolic architecture that combines large language models, external knowledge sources and discrete reasoning.”
- TALM: Tool Augmented Language Models
- Toolformer: Language Models Can Teach Themselves to Use Tools
- HuggingGPT: Solving AI Tasks with ChatGPT and its Friends in Hugging Face
- ToolLLM: Facilitating Large Language Models to Master 16000+ Real-world APIs
- Augmenting large language models with chemistry tools

参考文献

- Social Simulacra: Creating Populated Prototypes for Social Computing Systems
- Generative Agents: Interactive Simulacra of Human Behavior
- S3 : Social-network Simulation System with Large Language Model-Empowered Agents
- Communicative Agents for Software Development
- Unleashing Cognitive Synergy in Large Language Models: A Task-Solving Agent through Multi-Persona Self-Collaboration
- Improving Factuality and Reasoning in Language Models through Multiagent Debate
- Improving Language Model Negotiation with Self-Play and In-Context Learning from AI Feedback
- Examining the Inter-Consistency of Large Language Models: An In-depth Analysis via Debate
- Encouraging Divergent Thinking in Large Language Models through Multi-Agent Debate
- Auto-GPT for Online Decision Making: Benchmarks and Additional Opinions
- Emergent autonomous scientific research capabilities of large language models
- Large Language Models as Tool Makers