

# LLMs enhanced with Search and Tool Using 融合检索和工具调用的大语言模型

刘群 LIU Qun

华为诺亚方舟实验室 Huawei Noah's Ark Lab

中文信息学会信息检索专委会年会

2023-08-16, 山西太原



NOAH'S ARK LAB



# Content

背景 Background

大语言模型结合搜索引擎 LLMs combined with search engines

大语言模型调用外部工具 LLMs calling external tools

总结 Summary

# Content

## 背景 Background

大语言模型结合搜索引擎 LLMs combined with search engines

大语言模型调用外部工具 LLMs calling external tools

## 总结 Summary

# 大语言模型应用面临的问题

- ▶ 信息更新不及时，时效性不足
- ▶ 信息覆盖面不够
  - ▶ 长尾知识不足
  - ▶ 专业知识不足
- ▶ 信息修改不方便
- ▶ 推理能力不足
  - ▶ 数学计算精度不够
  - ▶ 符号推理能力不够：微积分、解方程……
- ▶ 无法与外界交互
  - ▶ 发邮件
  - ▶ 购物
  - ▶ 画画
  - ▶ 操控设备

# 解决上述问题的可行办法

- ▶ 使用搜索引擎：
  - ▶ 信息可以随时更新，确保永远最新
  - ▶ 可以方便添加大量专业知识和长尾知识，搜索引擎数据规模不受模型参数限制，增加数据规模消耗的算力远小于语言模型训练
  - ▶ 数据可以方便地增加、删除、修改
- ▶ 调用外部工具：
  - ▶ 调用数学计算工具
  - ▶ 调用符号推理工具
  - ▶ 调用环境交互工具（发邮件、画画、操控设备……）

# Content

背景 Background

大语言模型结合搜索引擎 LLMs combined with search engines

大语言模型调用外部工具 LLMs calling external tools

总结 Summary

# Content

大语言模型结合搜索引擎 LLMs combined with search engines

概览 Overview

方法 Methods

项目 Projects

我们的工作 Our Work

# Retrieval-based LMs and Applications (ACL2023 Tutorial)

## ACL 2023 Tutorial: Retrieval-based Language Models and Applications



Akari Asai<sup>1</sup>,

Sewon Min<sup>1</sup>,

Zexuan Zhong<sup>2</sup>,

Danqi Chen<sup>2</sup>

<sup>1</sup>University of Washington, <sup>2</sup>Princeton University

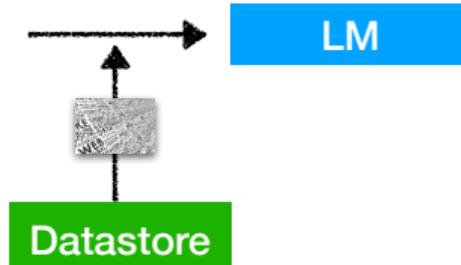
Sunday July 9 14:00 - 17:30 (EDT) @ Metropolitan West

Zoom link available on [Underline](#)

For those who have not registered to ACL: we will release video recordings after the tutorial

<https://acl2023-retrieval-lm.github.io/>

# Retrieval-based LMs and Applications (ACL2023 Tutorial)



vs.



LM	Datastore
# of parameters	# of tokens
kNN-LM (Khandelwal et al., 2020)	250M
NPM (Min et al., 2023)	350M
Atlas (Izacard et al., 2022)	11B
RETRO (Borgeaud et al., 2021)	7B
REPLUG (Shi et al., 2023)	$\leq 175B$

# Retrieval-augmented: LLM vs. Non-LLM

	Retrieval-augmented LMs (ACL2023 tutorial)	Retrieval-augmented LLMs (This talk)
目的	回答问题，通常答案较短，答对关键信息即可	回答问题，通常要生成较长、较完整的回复
模型	规模较小，可以是编码器、编码器-解码器、解码器模型	规模较大（百亿参数以上），通常只考虑解码器模型
评价	答案的准确性，模型的困惑度，自动评价为主	答案的合理性、真实性等，经常要人工评价和自动评价相结合
技术	可以修改语言模型和检索系统的架构，可以分别训练或联合训练	不修改语言模型和检索系统的架构，通过微调和提示训练大语言模型，不联合训练语言模型和检索模型

# Content

大语言模型结合搜索引擎 LLMs combined with search engines

概览 Overview

方法 Methods

项目 Projects

我们的工作 Our Work

# WebGPT

- ▶ 第一个千亿参数级别语言模型（GPT-3）结合搜索引擎的工作
- ▶ 模仿人类搜索的过程
- ▶ 生成带引用的搜索结果，便于溯源
- ▶ 已经被应用于New Bing 和ChatGPT检索插件

Table 2: An answer produced by our 175B best-of-64 model to a randomly-chosen question from the ELI5 test set (not cherry-picked). The full text of the references can be found in Appendix J, along with answers from our human demonstrators and the ELI5 dataset. Further samples are available at <https://openaipublic.blob.core.windows.net/webgpt-answer-viewer/index.html>.

<b>Question</b>	Why did we decide that certain words were "bad" and shouldn't be used in social settings?
<b>Answer</b>	We have all been taught that there are certain words that should not be used in public, at least not without good reason. But why do some words fall into this category while others do not? It turns out that there is not a single answer to this question, as each word has a different history and carries different meanings in different cultures and families [2]. However, there are some general categories that "bad" words tend to fall into. "Bad" words generally relate to parts of life that we don't like talking about in public, like bathroom functions, or negative ways of talking about people's religion, skin color, ability, or gender [3]. Often, people say bad words when they are experiencing strong emotions, and those words are linked to our emotions [3].
<b>References (titles only)</b>	[1, 2, 3] Why Are Some Words 'Bad'?   Vermont Public Radio ( <a href="http://www.vpr.org">www.vpr.org</a> ) [4] On Words: 'Bad' Words and Why We Should Study Them   UVA Today ( <a href="http://news.virginia.edu">news.virginia.edu</a> ) [5] The Science of Curse Words: Why The &@#! Do We Swear? ( <a href="http://www.babbel.com">www.babbel.com</a> )

Nakano et al. “WebGPT: Browser-assisted question-answering with human feedback” , arxiv2112.09332v3.

# WebGPT

- ▶ 数据生产分为两个部分——demonstration（演示）和comparison（对比）
- ▶ demonstration就是按照人类使用搜索引擎时的操作，针对给定的问题，对搜索引擎返回的结果，进行过滤。
- ▶ comparison就是针对某个特定的问题，模型生成两个答案，标注人员先对每个答案从trustworthiness（可信度）、来源对结果的支持程度进行打分。在对两个答案进行对比打分。

Nakano et al. “WebGPT: Browser-assisted question-answering with human feedback” , arxiv2112.09332v3.

# WebGPT

Table 1: Actions the model can take. If a model generates any other text, it is considered to be an invalid action. Invalid actions still count towards the maximum, but are otherwise ignored.

Command	Effect
Search <query>	Send <query> to the Bing API and display a search results page
Clicked on link <link ID>	Follow the link with the given ID to a new page
Find in page: <text>	Find the next occurrence of <text> and scroll to it
Quote: <text>	If <text> is found in the current page, add it as a reference
Scrolled down <1, 2, 3>	Scroll down a number of times
Scrolled up <1, 2, 3>	Scroll up a number of times
Top	Scroll to the top of the page
Back	Go to the previous page
End: Answer	End browsing and move to answering phase
End: <Nonsense, Controversial>	End browsing and skip answering phase

Nakano et al. “WebGPT: Browser-assisted question-answering with human feedback” , arxiv2112.09332v3.

# WebGPT

This is what the app looks like while **browsing**:

The screenshot shows the WebGPT interface with various sections highlighted by red boxes:

- Question:** "Why do spicy foods make you sweat?"
- Flags:** Buttons for "This question does not make sense" and "This question should not be answered".
- Page title:** "Why Does Spicy Food Make You Sweat? (www.epainassist.com)"
- Navigation bar:** Includes a back button, search bar ("Why do spicy foods make you sweat?"), and a "Find in page" button.
- Page text:** A detailed explanation about capsaicin and its effect on the body's heat-sensing receptors.
- Reset buttons:** "Reset without saving" and "Save and reset".
- Quotes section:** A box containing a quote from Livestrong.com: "[1] Why Do People Sweat When They Eat Spicy Foods? | Livestrong.com (www.livestrong.com)".
- Budget section:** Displays "Number of quote tokens left: 381" and "Number of actions left: 16".
- Done quoting! Write an answer:** A button to proceed with the generated response.
- Scroll bar:** A vertical scroll bar on the right side of the main content area.

Nakano et al. “WebGPT: Browser-assisted question-answering with human feedback” , arxiv2112.09332v3.

# WebGPT

This is what the app looks like while **answering**:

The screenshot shows the "Question answering playground" interface. On the left, the "Quoting" section contains two entries: [1] Why Do People Sweat When They Eat Spicy Foods? (Livestrong.com) and [2] Why Does Spicy Food Make You Sweat? (www.epainassist.com). Each entry includes a snippet of text explaining the mechanism of sweating from spicy food. Below these is a "Budget section" with the message "You clicked 'Done quoting'". On the right, the "Answer section" is a large text area with the placeholder "Write down your final synthesis here". At the bottom, there are "Reset buttons" for "Reset without saving" and "Save and reset". A "Lock in answer" button is located in the bottom right corner of the answer area.

Question answering playground

Why do spicy foods make you sweat? **Question**

[This question does not make sense] [This question should not be answered] **Flags** **Quotes section**

**Quotes**

[1] Why Do People Sweat When They Eat Spicy Foods? | Livestrong.com ([www.livestrong.com](http://www.livestrong.com))

According to an Arizona University article entitled "The Capsaicin Receptor: A Pepper's Pathway to Pain," capsaicin is a molecule found in chili peppers that causes the spicy taste. The body also has a capsaicin receptor found on certain nerves that are heat-sensitive. When activated, these nerves send signals to the spinal cord and brain to send a perception of heat-related pain. The brain responds by triggering chemical reactions to cause cooling of the body, such as a sweat response.

[2] Why Does Spicy Food Make You Sweat? ([www.epainassist.com](http://www.epainassist.com))

Coming to the mechanism as to what causes you to sweat when eating spicy food, the answer is that chemical Capsaicin activates receptors in the skin that normally respond to excess heat or in a hot environment. These receptors are known as polymodal nociceptors. These receptors only respond to extreme conditions or intense mechanical stimulation like a burn or a cut. The signals sent by the receptors to the brain due to the chemical makes the brain think that the body is subjected to extreme heat and thus in order to cool down the body the brain acts and perspiration comes out in order to cool down the body.

**Answer section**

Write down your final synthesis here

**Lock in answer**

Finished browsing

Budget section

You clicked "Done quoting"

**Reset buttons**

Reset without saving

Save and reset

Nakano et al. “WebGPT: Browser-assisted question-answering with human feedback” , arxiv2112.09332v3.

# WebGPT

This is what the comparisons website looks like(upper half):

The screenshot shows a web-based comparison tool for a question: "Why is processed food significantly cheaper than unprocessed food?".

**Question:** Why is processed food significantly cheaper than unprocessed food?  
Shouldn't the extra labor drive the cost up, not down?

**Flags:** The question does not make sense. The question is on a controversial topic.

**Instructions:** For this question, we would like you to provide annotations for each option, as well as comparison ratings.

**Tabs:** Option A (selected), Option B, Comparison.

**Answer:** Processed food is cheaper than unprocessed food because the processed food industry is enormously efficient at causing it to produce tasty, cheap food [1][2][3]. The time it takes to grow, harvest, and transport fruits and vegetables to market—as well as ensure they look good and stay fresh until they can be sold—is extremely expensive. [4]

**Annotations:** Label the sources, and use the tools to annotate the answer.

**Annotation tools:**

- Strong support: Core → Processed food is cheaper than unprocessed food
- Weak support: Core → the processed food industry is enormously eff...
- No support: Core → capable of producing tasty, cheap food
- Citation error: Core → grow, harvest
- Core → transport fruits and vegetables to market—as ...

**Sources:** [1] Why Processed Food Is Cheaper Than Healthier Options ([www.npr.org](http://www.npr.org)) Trustworthy ✓  
[2] Why Processed Food Is Cheaper Than Healthier Options ([www.npr.org](http://www.npr.org)) Trustworthy ✓  
[3] Why Processed Food Is Cheaper Than Healthier Options ([www.npr.org](http://www.npr.org)) Trustworthy ✓

**Critique boxes:** Is there anything else that makes this answer useful to the person asking the question? (Optional)

Strengths: Weaknesses:

Nakano et al. “WebGPT: Browser-assisted question-answering with human feedback” , arxiv2112.09332v3.

# WebGPT

This is what the comparisons website looks like(lower half):

Comparison ratings

Ratings

Unsupported information

Look at the information with **no support**:

Option A: **grow harvest** Nothing

How well does the answer avoid unhelpful information? Less information with no support is better.

A much better      A better      Equally good      B better      B much better

Usefulness overall

Overall, how useful is the answer to the person asking the question?

A much better      A better      Equally good      B better      B much better

Additional comments (optional):

Comments box

Submit button

Nakano et al. “WebGPT: Browser-assisted question-answering with human feedback” , arxiv2112.09332v3.

# WebGPT

训练过程：

- ▶ Behavior cloning (BC). 使用用户行为数据对模型进行fine-tune。
- ▶ Reward modeling (RM). 使用BC模型去掉softmax层为起点，根据用户对多个结果的排序，使用交叉熵loss，训练一个奖励模型。
- ▶ Reinforcement learning (RL). 利用奖励模型作为奖励信息，对BC模型进行PPO强化学习训练。把RM的分数作为环境奖励加到每个episode的结尾，并把它作为KL惩罚加到每一个token上以避免过度优化。
- ▶ Rejection sampling (best-of-n). 从BC或者RL模型中采用若干（4/16/64）个结果，选择其中RM打分最高的那个。使用这种方法可以不用对模型进行强化学习RL训练，以减少训练时间，但会增加推理时间。

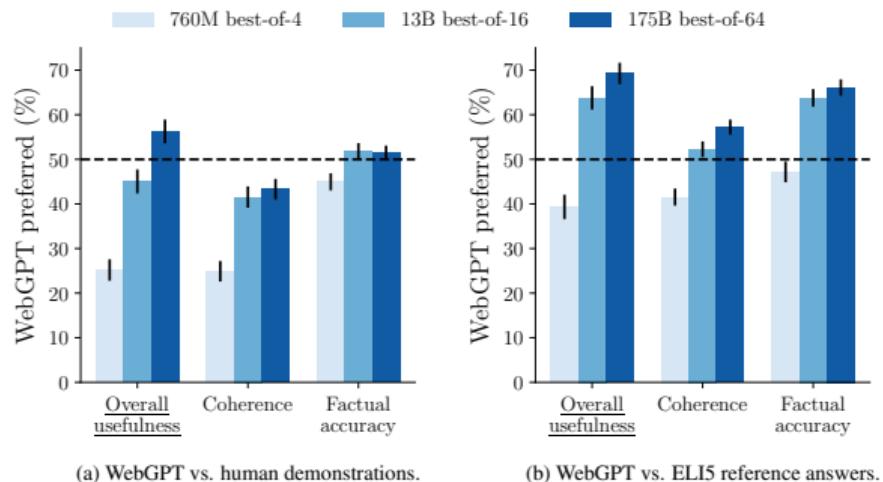
可以看到，这个过程跟OpenAI后来提出的RLHF方法很相似。

Nakano et al. “WebGPT: Browser-assisted question-answering with human feedback” , arxiv2112.09332v3.

# WebGPT

ELI5（问题都来自Reddit的“Explain Like I'm Five”板块）结果：

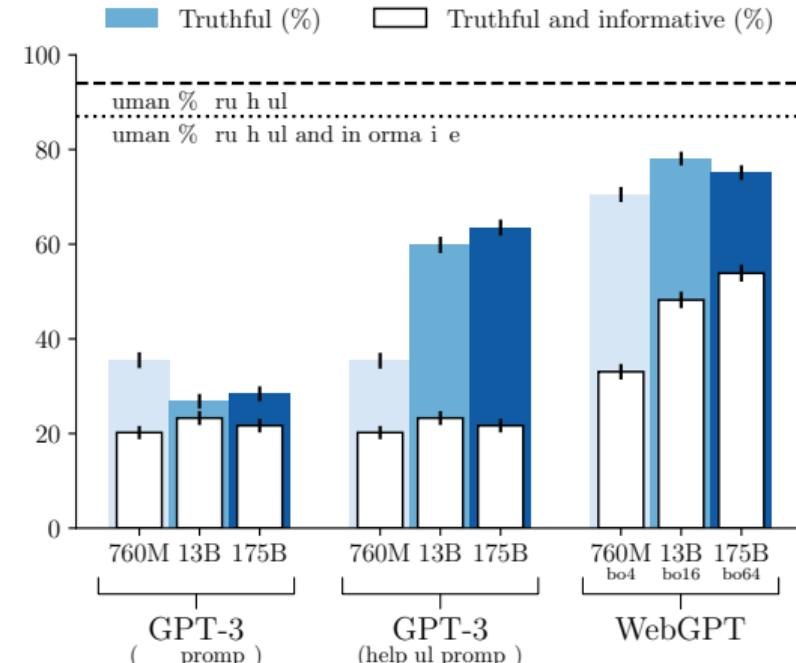
- ▶ 采用训练RM时相同的评价方法，给定系统生成的结果和人工结果，由标注员选择哪个结果更好（可以选择打平）
- ▶ 结果显示，最大的模型（175B best-of-64）跟人工标注员生成的结果相比，在56%的情况下被选为更好的结果；跟ELI5的参考答案相比，69%的情况下被选为更好的结果。



Nakano et al. “WebGPT: Browser-assisted question-answering with human feedback”, arxiv2112.09332v3.

# WebGPT

TruthfulQA (一些容易出现真实性错误的问题) 结果:



- ▶ 可以看到WebGPT无论在“真实”还是“真实且有信息”方面都远好原始的GPT-3模型
- ▶ 但跟人工结果相比仍然有较大差距。

Nakano et al. “WebGPT: Browser-assisted question-answering with human feedback”, arxiv2112.09332v3.

# Content

大语言模型结合搜索引擎 LLMs combined with search engines

概览 Overview

方法 Methods

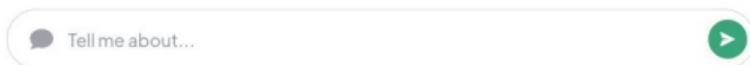
项目 Projects

我们的工作 Our Work

# 项目 Projects

- ▶ Perplexity.ai
- ▶ NeevaAI
- ▶ YouChat (You.com)
- ▶ New Bing
- ▶ Bard
- ▶ ChatGPT with Web Access

# Perplexity.ai



## Ask Anything

Popular Now				
james webb telescope (2.0)	>	perplexity bird sql	>	malaysia airlines flight mh370
elon musk journalists	>	Is a recession coming?	>	de santis anti-vaccine
What's going on with binance	>	@elonmusk	>	elonjet suspension

Perplexity.ai

what does elon musk say about openai chatgpt X | ▶

### PERPLEXITY

[Tweet](#)

ChatGPT is a chatbot created by OpenAI, a company founded by Elon Musk<sup>[1]</sup>. It is based on the GPT-2 algorithm<sup>[2]</sup>. After Musk found out that the chatbot was accessing Twitter's database to train itself, he put a stop to it<sup>[3]</sup>. He has previously called it "scary good"<sup>[4]</sup> and warned that we are not far from dangerously strong AI<sup>[4]</sup>.

Was this result accurate?

### CITATIONS

#### 1. ChatGPT: Elon Musk led OpenAI's new chatbot can ...

<https://www.lifestyleasia.com/ind/gear/tech/openai-new-chatbot-chatgpt>

There is a new chatbot in town. Here is why ChatGPT from OpenAI is revolutionising AI software and has become a viral sensation.

#### 2. AI bot ChatGPT stuns academics with essay-writing skills and ...

<https://www.theguardian.com/technology/2022/dec/04/ai-bot-chatgpt-stuns-academics-with-essay-writing-skills-and-usability>

Latest chatbot from Elon Musk-founded OpenAI can identify incorrect premises and refuse to answer inappropriate requests.

#### 3. Here's What To Know About OpenAI's ChatGPT—What It's ...

<https://www.forbes.com/sites/ariannahjohnson/2022/12/07/heres-what-to-know-about-openais-chatgpt-what-its-disrupting-and-how-to-use-it>

On Sunday, Elon Musk tweeted that he found out OpenAI was accessing Twitter's database to train ChatGPT, so he put an immediate pause on it ...

#### 4. Elon Musk's history with OpenAI—the maker of A.I. chatbot ...

<https://www.yahoo.com/news/elon-musk-history-openai-maker-011005933.html>

This month Musk called OpenAI's ChatGPT "scary good" and warned, "We are not far from dangerously strong AI."

# YouChat

YOU | The latest nobel prize in physics | Log In | Sign Up

All latest nobel prize in physics nobel prize in physics winners what is the nobel prize in physics nobel prize in physics winners

Chat Images Videos News Maps Work For Me Write Code Imagine Study Social Shop Add

1 app and 84,900,000 results

N- The Nobel Prize in Physics 2022 - NobelPrize.org  
nobelprize.org > prizes > physics  
The Nobel Prize in Physics 2022 was awarded jointly to Alain Aspect, John F. Clauser and Anton Zeilinger "for experiments with entangled photons, establishing the violation of Bell inequalities and pioneering quantum..."

N- The Nobel Prize in Physics  
nobelprize.org > prizes > physics  
Physics was the prize area which Alfred Nobel mentioned first in his will from 1895. At the end of the nineteenth century, many people considered physics as the foremost of the sciences, and perhaps Nobel saw it this way as...

Quick Facts  
[https://www.quantamagazine.org/pioneering-quant...](https://www.quantamagazine.org/pioneering-quantum-physics/)

Quantum entanglement - Wikipedia  
Alain Aspect, John Clauser and Anton Zeilinger have won the 2022 Nobel Prize in Physics for groundbreaking experiments with entangled particles. From left: John Clauser, Anton Zeilinger and Alain Aspect.

N- The Nobel Prize in Physics 2021 - NobelPrize.org

YouChat

The latest nobel prize in physics

The latest Nobel Prize in Physics was awarded in 2022 to Alain Aspect, John F. Clauser and Anton Zeilinger "for experiments with entangled photons [1], establishing the violation of Bell inequalities and pioneering quantum information science".  
1. The Nobel Prize in Physics 2022 - NobelPrize.org  
<https://www.nobelprize.org/prizes/physics/2022/summary/>

what are entangled photons?

Entangled photons are two photons that are connected in a way that any change to one will cause a change in the other, regardless of the distance between them. This phenomenon is known as quantum entanglement

This product is in beta and its accuracy may be limited.  
You.com is not liable for content generated.

# Google Bard

Which is the second highest mountain in the world?

The second highest mountain in the world is K2. It is located in the Karakoram Range, on the border between Pakistan and China. K2 is 8,611 meters (28,251 feet) tall, and is considered one of the most challenging mountains to climb.

The name "K2" comes from the Great Trigonometrical Survey of British India, which used the letter "K" to designate mountains in the Karakoram Range. The mountain was originally named "Mount Godwin-Austen" after Henry Godwin-Austen, a British surveyor who explored the Karakoram Range in the 1850s. However, the name "K2" eventually became more popular, and is now the more commonly used name for the mountain.

K2 is a very difficult mountain to climb, and has a high fatality rate. Only about 40% of climbers who attempt to summit K2 are successful. The mountain is known for its steep slopes, unpredictable weather, and avalanches.

The first successful ascent of K2 was made in 1954 by a team of Italian climbers led by Achille Compagnoni and Lino Lacedelli. Since then, there have been over 3,000 attempts to summit K2, and over 800 climbers have died on the mountain.

...

Enter a prompt here

Bard may display inaccurate or offensive information that doesn't represent Google's views. [Bard Privacy Notice](#)

# New Bing

Who was the better baseball player? Ted Williams or Rickey Henderson

✓ Searching for: **Ted Williams vs Rickey Henderson**

✓ Generating answers for you...

That's a tough question, as Ted Williams and Rickey Henderson were very different types of players. Williams was one of the best hitters ever, with a career batting average of .344 and 521 home runs <sup>1</sup>. Henderson was the best basestealer and leadoff hitter ever, with a record 1,406 stolen bases and 2,295 runs scored <sup>2</sup>. Both players won multiple awards and honors, and are in the Hall of Fame <sup>3</sup>.

Some might argue that Williams was the better player because of his superior hitting skills and power, while others might favor Henderson for his speed, defense, and ability to get on base. It depends on what you value more in a baseball player. One way to compare them is to look at their career Wins Above Replacement (WAR), which measures how many more wins a player contributed to his team than a replacement-level player. According to Baseball-Reference.com, Williams had a career WAR of 123.1, while Henderson had a career WAR of 111.2. This suggests that Williams was slightly more valuable than Henderson over their careers, but not by much.

Of course, WAR is not a perfect measure, and there are other factors to consider, such as era, competition, and postseason performance. Ultimately, there is no definitive answer to who was the better player, as both Williams and Henderson were among the greatest left fielders and players of all time.

Learn more: 1. [mlbcomparisons.com](#) 2. [mlb.com](#) 3. [baseball-fever.com](#) 4. [mlb.com](#) 5. [quora.com](#)

Who is the best left fielder ever?

✓ Searching for: **best left fielder ever**

 Ask me anything...

# 生成式搜索引擎的可验证性评估

- ▶ 人工评价了四个受欢迎的生成式搜索引擎：Bing Chat, NeevaAI, perplexity.ai, YouChat
- ▶ 发布了人工审核注释，为进一步开发可信赖的生成式搜索引擎提供了数据支持
- ▶ 定义了引用召回率和引用精度的评价指标，以鼓励开发全面和正确引用的系统
- ▶ 发现1：所生成的回复流畅性较好，而且人类评价的有用性也较高
- ▶ 发现2：所生成的引用信息的准确率和召回率都不高

<i>Fluency (↑)</i>		<i>Perceived Utility (↑)</i>	
	Average Over All Queries		Average Over All Queries
Bing Chat	4.40	Bing Chat	4.34
NeevaAI	4.43	NeevaAI	4.48
perplexity.ai	4.51	perplexity.ai	4.56
YouChat	4.59	YouChat	4.62
Average	4.48	Average	4.50
<i>Citation Recall (%; ↑)</i>		<i>Citation Precision (%; ↑)</i>	
	Average Over All Queries		Average Over All Queries
Bing Chat	58.7	Bing Chat	89.5
NeevaAI	67.6	NeevaAI	72.0
perplexity.ai	68.7	perplexity.ai	72.7
YouChat	11.1	YouChat	63.6
Average	51.5	Average	74.5

Liu, et al., Evaluating Verifiability in Generative Search Engines, arxiv2304.09848

# 让大语言模型生成带引用的文本

- ▶ 该工作研究了一种新的LLM生成范式评测基准ALCE:
  - ▶ 给定一个问题，系统生成文本，同时从一个大型检索语料库中提供引文。每个语句可能包含多个引文
  - ▶ 文章从三个维度构建了自动评价指标：流利性、正确性、引用质量
  - ▶ 文章显示这些自动评价指标与人类评价强相关
- ▶ 文章发现：尽管所有的系统都能够提供流畅和一致的回答，但在正确性和引用质量方面仍有很大的改进空间，例如：在ELI5数据集上，ChatGPT基线有49%的生成没有得到引用段落的完全支持。

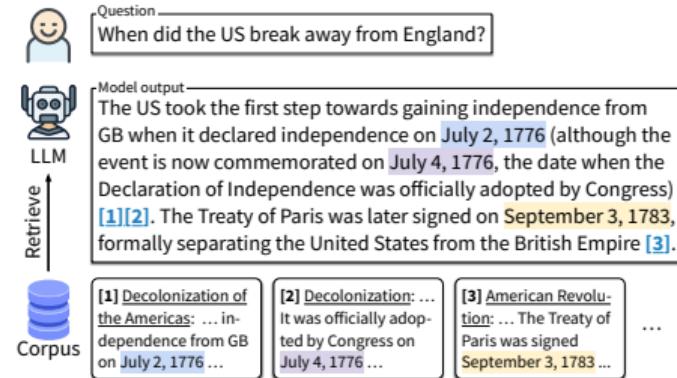


Figure 1: The task setup of ALCE. Given a question, the system generates text while providing *citing passages* from a large retrieval corpus. Each statement may contain multiple citations (e.g., [1][2]).

# Content

大语言模型结合搜索引擎 LLMs combined with search engines

概览 Overview

方法 Methods

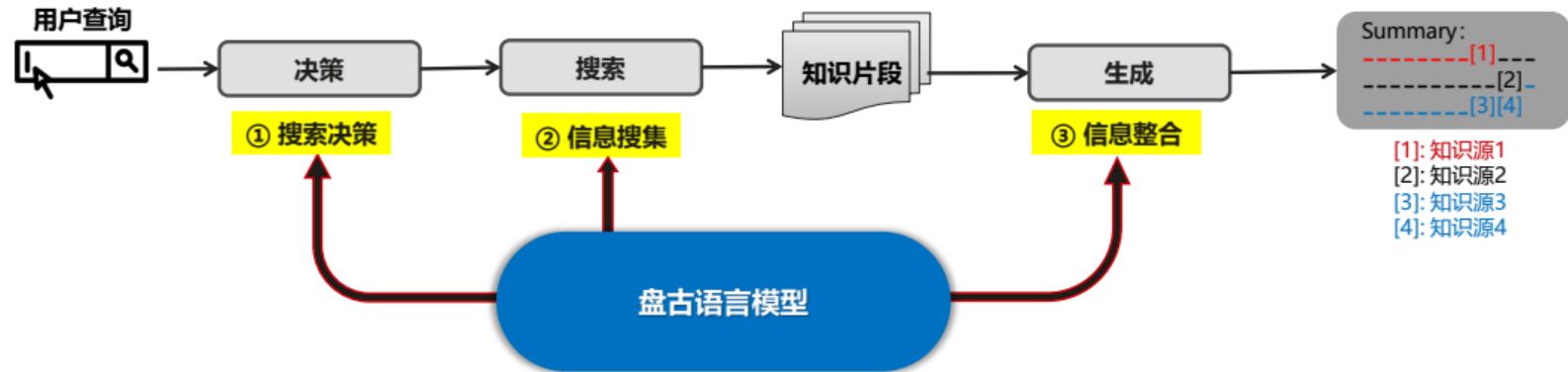
项目 Projects

我们的工作 Our Work

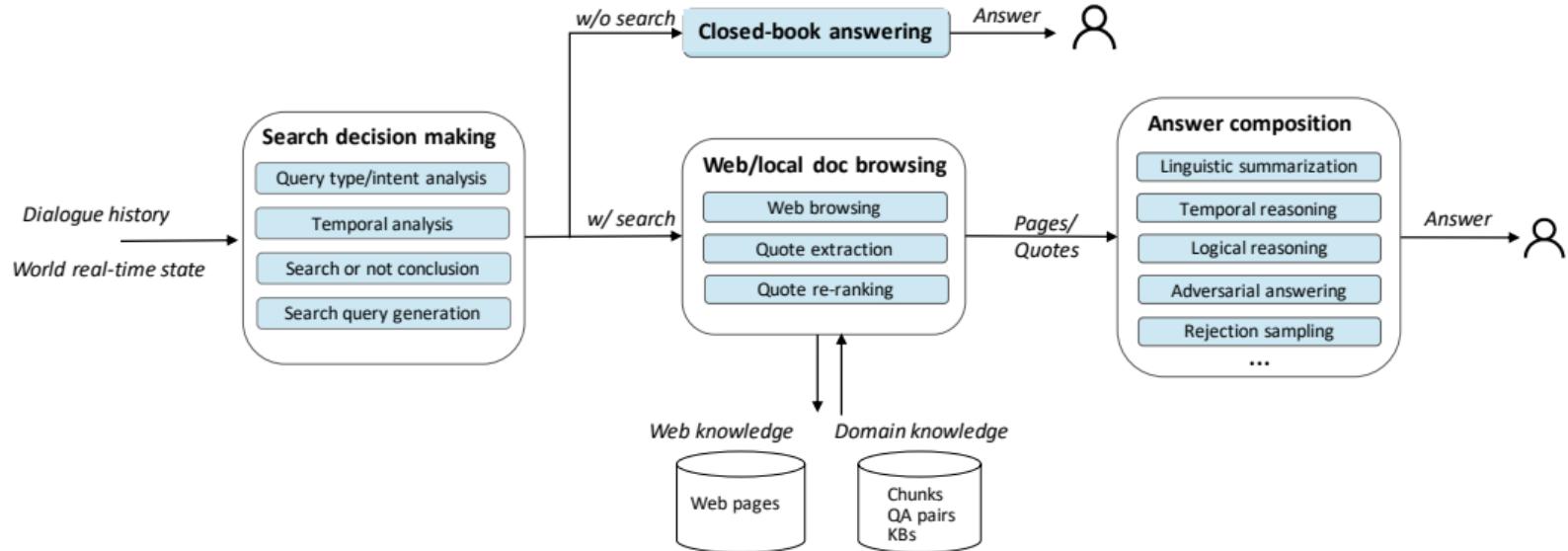
# 盘古基础语言模型

- ▶ 华为诺亚方舟实验室于2021年4月发布了2000亿参数的稠密型的盘古- $\alpha$  中文大语言模型
- ▶ 2023年3月，我们又发布了1万亿参数的采用MoE架构的稀疏型盘古- $\Sigma$  大语言模型
- ▶ 基于盘古- $\Sigma$  大语言模型的一个中文子模型，我们继续开发了盘古基础语言模型：
  - ▶ 大规模数据采集和清洗
  - ▶ 指令微调（Instruct Tuning）
  - ▶ 极致的硬件亲和推理优化
- ▶ 在此基础上，我们开发了盘古语言模型融合搜索引擎和工具调用的技术

# 盘古语言模型融合搜索系统框架



# 盘古语言模型融合搜索系统框架



# 样例：回答实时问题

台风泰利什么时候登陆

2023-07-17 18:57:26

- ✓ 正在搜索： google Search搜索成功， search query: 台风泰利什么时候登陆
- ✓ 正在为您生成答案...

台风泰利将于2023年17日夜间在广东阳江到海南文昌一带沿海登陆[2]。

根据中央气象台的最新预报，预计台风泰利将在17日夜间登陆，登陆时可能会达到台风级或强台风级，最大风力为12到14级，最大风速为35到42米/秒[2]。

据统计，近30年来，初台最爱登陆地点为广东[3]。

---

了解详细信息：[m.haikou.bendibao.com](http://m.haikou.bendibao.com) [news.cn](http://news.cn) [cma.gov.cn](http://cma.gov.cn) [m.nongjiao.com](http://m.nongjiao.com) [m.nongjiao.com](http://m.nongjiao.com)

# 样例：回答误导性问题

大连属于山东省哪个地区

2023-07-17 18:47:43

✓ 正在搜索：google Search搜索成功，search query：大连属于山东省哪个地区

✓ 正在为您生成答案...

大连不属于山东省，而是位于中国东北地区。大连旧称旅大市，旧名达尔尼、达里尼、青泥洼<sup>[2]</sup>。大连在战国末期属燕国辽东郡，公元前221年秦统一六国后，置辽东郡，治所在襄平(今辽东半岛)<sup>[3]</sup>。

# 样例：拒绝回答问题

华为Mate 90核心配置有哪些

2023-07-17 19:02:33

- ✓ 正在决策是否搜索： google Search搜索成功， search query： 华为mate90核心配置,华为mate90哪些配置
- ✓ 正在为您生成答案...

抱歉，根据已有信息，无法回答该问题，建议您提供更多相关信息。

了解详细信息：

[tech.sina.cn](#)

[zhihu.com](#)

[post.smzdm.com](#)

[zhuanlan.zhihu.com](#)

[zhihu.com](#)

# 样例：回答长尾问题

TinyBERT和DistillBERT有什么区别

2023-07-17 19:30:38

- ✓ 正在决策是否搜索： google Search搜索成功， search query: tinybert和distillbert的区别,tinybert与distillbert的不同点
- ✓ 正在为您生成答案...

TinyBERT和 DistillBERT都是 BERT的蒸馏版本，但有以下区别：

- TinyBERT是一种更轻量级的版本，具有更少的参数和更快的推理速度[1] [3] [5]。
- DistillBERT是一种更传统的蒸馏方法，将一个较大的模型(如 BERT)的知识蒸馏到一个更小的模型(如 TinyBERT)中[4]。
- TinyBERT在经验上比 DistillBERT更有效，能够超过基线 BERT的96%，容量小7.5倍，推理速度快9.4倍[3]。
- TinyBERT的性能和泛化能力都得到了显著提高，但其参数仍然比 DistillBERT少28%左右[1] [3] [5]。

了解详细信息：[cnblogs.com](#) [zhuanlan.zhihu.com](#) [cnblogs.com](#) [blog.csdn.net](#) [cnblogs.com](#)

# 样例：回答代码问题

从一段字符串中识别出中文，正则写法

2023-07-17 19:43:05

以下是使用 Python 正则表达式识别代码中的中文字符的示例代码[2] [3]：

```
1 import re
2 def extract_chinese_chars(code):
3     chinese_chars = re.findall(r'[\u4e00-\u9fa5]', code)
4     print("中文字符:", chinese_chars)
```

复制

该函数使用 `re.findall` 函数和正则表达式模式 `[\u4e00-\u9fa5]` 来匹配中文字符，并将匹配结果返回。

了解详细信息：

[cloud.tencent.com](#)

[cloud.tencent.com](#)

[cloud.tencent.com](#)

[clayandmore.github.io](#)

[cloud.tencent.com](#)

# Content

背景 Background

大语言模型结合搜索引擎 LLMs combined with search engines

大语言模型调用外部工具 LLMs calling external tools

总结 Summary

# Content

大语言模型调用外部工具 LLMs calling external tools

概览 Overview

方法 Methods

项目 Projects

我们的工作 Our Work

# Tool Learning with Foundation Models

---

## Tool Learning with Foundation Models

---

**Yujia Qin<sup>1</sup>, Shengding Hu<sup>1</sup>, Yankai Lin<sup>2\*</sup>, Weize Chen<sup>1</sup>, Ning Ding<sup>1</sup>, Ganqu Cui<sup>1</sup>,  
Zheni Zeng<sup>1</sup>, Xuanhe Zhou<sup>1</sup>, Yufei Huang<sup>1</sup>, Chaojun Xiao<sup>1</sup>, Chi Han<sup>3</sup>, Yi Ren Fung<sup>3</sup>,  
Yusheng Su<sup>1</sup>, Huadong Wang<sup>1</sup>, Cheng Qian<sup>1</sup>, Runchu Tian<sup>1</sup>, Kunlun Zhu<sup>8</sup>, Shihao Liang<sup>8</sup>,  
Xingyu Shen<sup>1</sup>, Bokai Xu<sup>1</sup>, Zhen Zhang<sup>1</sup>, Yining Ye<sup>1</sup>, Bowen Li<sup>1</sup>, Ziwei Tang<sup>5</sup>, Jing Yi<sup>1</sup>,  
Yuzhang Zhu<sup>1</sup>, Zhenning Dai<sup>1</sup>, Lan Yan<sup>1</sup>, Xin Cong<sup>1</sup>, Yaxi Lu<sup>1</sup>, Weilin Zhao<sup>1</sup>,  
Yuxiang Huang<sup>1</sup>, Junxi Yan<sup>1</sup>, Xu Han<sup>1</sup>, Xian Sun<sup>7</sup>, Dahai Li<sup>7</sup>, Jason Phang<sup>4</sup>,  
Cheng Yang<sup>5</sup>, Tongshuang Wu<sup>6</sup>, Heng Ji<sup>3</sup>, Guoliang Li<sup>1</sup>, Zhiyuan Liu<sup>1\*</sup>, Maosong Sun<sup>1\*</sup>**

<sup>1</sup>Tsinghua University, <sup>2</sup>Renmin University of China, <sup>3</sup>University of Illinois Urbana-Champaign,

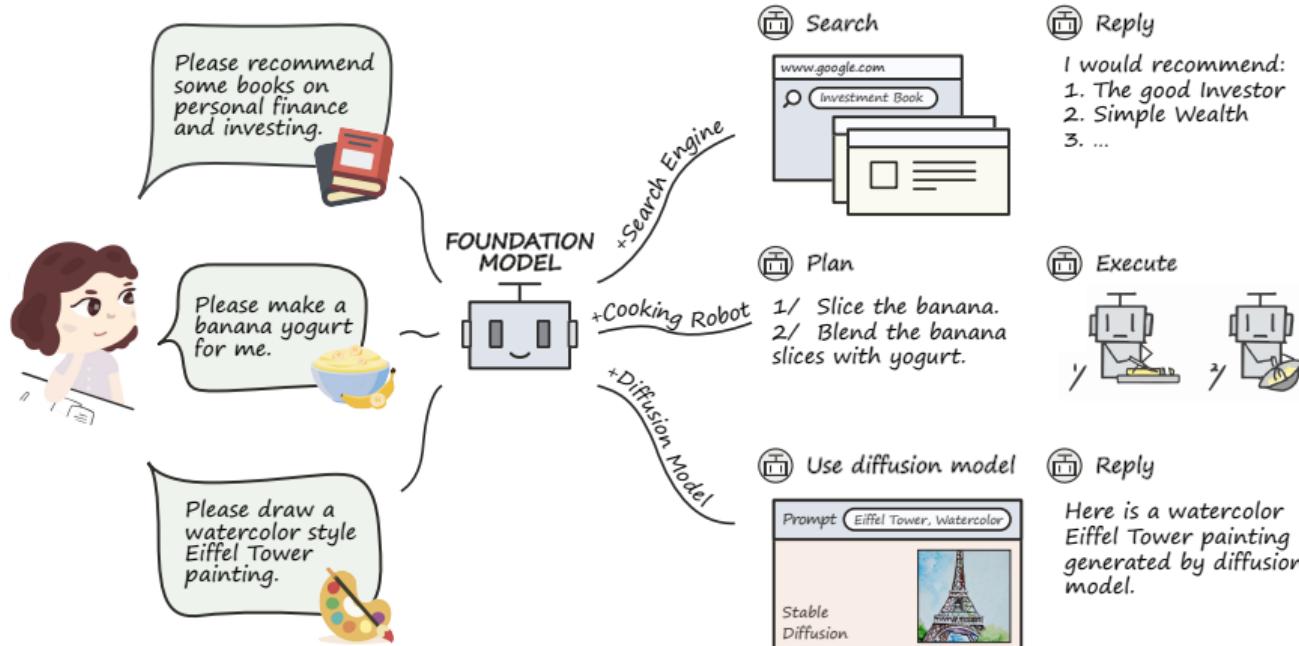
<sup>4</sup>New York University, <sup>5</sup>Beijing University of Posts and Telecommunications,

<sup>6</sup>Carnegie Mellon University, <sup>7</sup>Zhihu Inc., <sup>8</sup>ModelBest Inc.

qyj20@mails.tsinghua.edu.cn

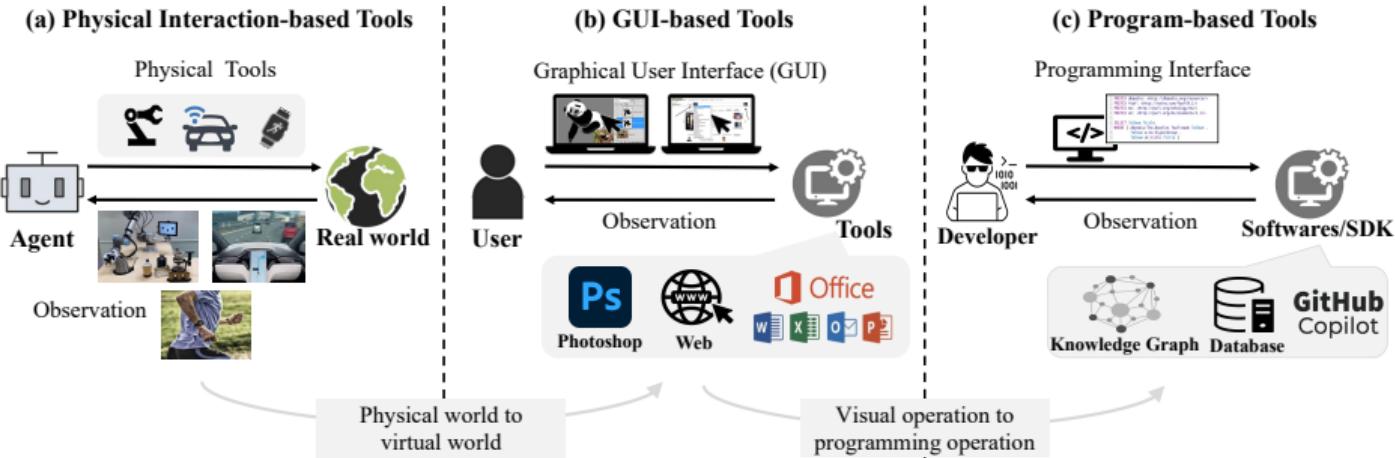
Qin, et al. “Tool Learning with Foundation Models.” arXiv2304.08354.

# Tool Learning with Foundation Models



Qin, et al. “Tool Learning with Foundation Models.” arXiv2304.08354.

# Tool Learning with Foundation Models



Qin, et al. “Tool Learning with Foundation Models.” arXiv2304.08354.

# Content

大语言模型调用外部工具 LLMs calling external tools

概览 Overview

方法 Methods

项目 Projects

我们的工作 Our Work

# 外部工具调用的能力进阶

- ▶ 在上下文学习（ICL）能力较弱时：
  - ▶ 通过有监督微调（SFT）实现外部工具调用（类似ChatGPT Code Interpreter）
- ▶ 在上下文学习（ICL）能力较强，但思维链（CoT）能力还较弱时：
  - ▶ 通过上下文学习（ICL）实现插件调用（只需提供插件的描述，无需SFT）（插件形式，类似ChatGPT Plug-ins）
- ▶ 在思维链（CoT）能力较强时：
  - ▶ 通过思维链（CoT）进行任务分解，实现外部工具的规划和搜索，以完成复杂任务（Agent形式，类似AutoGPT）

# 通过有监督微调（SFT）调用工具（Code Interpreter）

- ▶ 通过构造指令（Instruction）数据，让模型学会使用工具
- ▶ 需要模型具备初步的代码生成能力
- ▶ 使用工具的能力通过SFT内嵌在模型中
- ▶ 典型工作：
  - ▶ LAMDA (Google): 通过人工标注学会调用计算器、翻译系统和Web搜索等三个工具
  - ▶ ChatGPT Code Interpreter: 用自然语言直接生成Python代码并调用Python引擎执行所生成代码

# ToolFormer: 一种工具调用的SFT数据生成方法

- ▶ 工具调用的SFT需要大量人工标注的训练数据
- ▶ 这篇文章提出了一种工具调用SFT数据的生成方法
- ▶ 利用已有大语言模型的上下文学习（In-Context-Learning）能力，生成指令数据，用于训练另外一个语言模型调用工具的能力：
  - ▶ 每个API需提供少量样例
  - ▶ 把每个API的少量样例，随机组合一个文本片段，生成一个API调用的指令数据
  - ▶ 检查API调用返回的结果，如果返回结果和文本片段的内容吻合，则留下作为指令数据，否则丢弃
  - ▶ 如此可以很容易得到大量的指令数据，用于训练另外一个语言模型调用工具的能力
- ▶ 以数学推理能力为例，通过使用Toolformer构造数据进行微调，仅6.7B的GPT-J大幅超过了远大于它的175B的GPT-3。

Schick, et al. “Toolformer: Language Models Can Teach Themselves to Use Tools.” arXiv2302.04761.

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.

The name derives from "la tortuga", the Spanish word for [MT("tortuga") → turtle] turtle.

The Brown Act is California's law [WikiSearch("Brown Act") → The Ralph M. Brown Act is an act of the California State Legislature that guarantees the public's right to attend and participate in meetings of local legislative bodies.] that requires legislative bodies, like city councils, to hold their meetings open to the public.



Your task is to add calls to a Question Answering API to a piece of text. The questions should help you get information required to complete the text. You can call the API by writing "[QA(question)]" where 'question' is the question you want to ask. Here are some examples of API calls:

**Input:** Joe Biden was born in Scranton, Pennsylvania.

**Output:** Joe Biden was born in [QA("Where was Joe Biden born?")] Scranton. [QA("In which state is Scranton?")] Pennsylvania.

**Input:** Coca-Cola, or Coke, is a carbonated soft drink manufactured by the Coca-Cola Company.

**Output:** Coca-Cola, or [QA("What other name is Coca-Cola known by?")] Coke, is a carbonated soft drink manufactured by [QA("Who manufactures Coca-Cola?")] the Coca-Cola Company.

**Input: x**

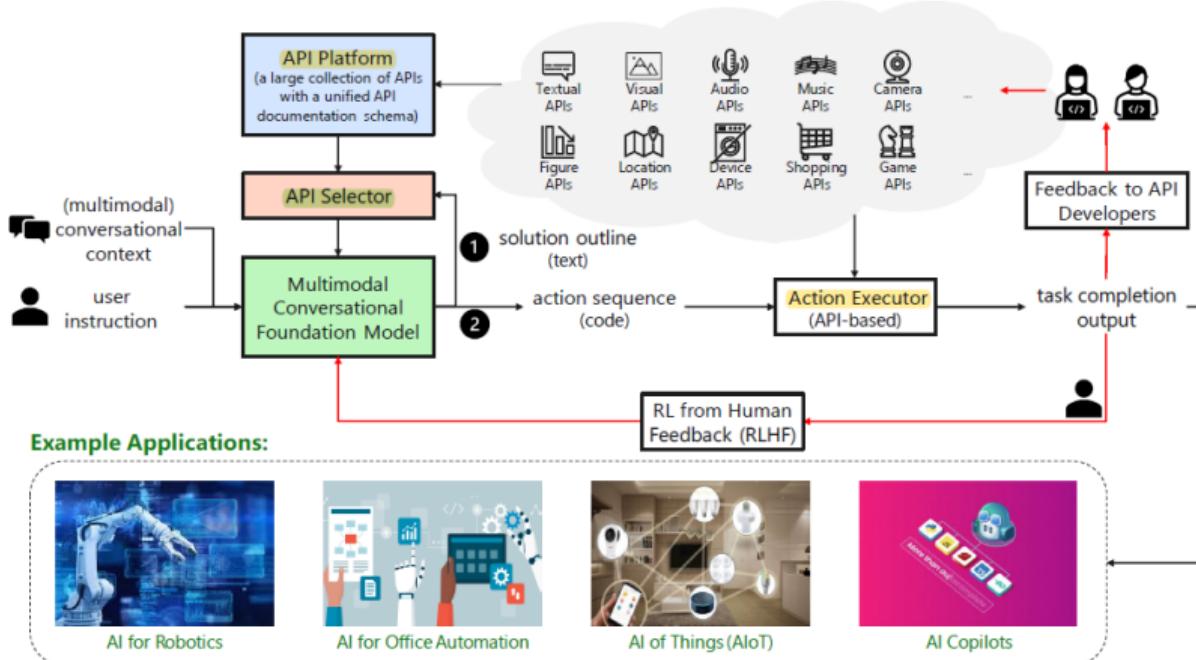
**Output:**

$x_{i-1}^1$  = Pittsburgh is also known as  
 $x_{i-1}^2$  = the Steel City  
 $c_i^1$  = What other name is Pittsburgh known by?  
 $c_i^2$  = Which country is Pittsburgh in?  
 $r_i^1$  = Steel City  
 $r_i^2$  = United States  
 $L_j(c_i^1 \rightarrow \text{Steel City}) < \min(L_i(c_i^1 \rightarrow e), L_j(e))$   
 $L_j(c_i^2 \rightarrow \text{United States}) < \min(L_i(c_i^2 \rightarrow e), L_j(e))$   
 $x^1$  = Pittsburgh is also known as [QA(What ...? → Steel City)] the Steel City.

# 通过上下文学习 (ICL) 调用工具 (Plug-ins)

- ▶ 工具提供商撰写工具描述 (Description) 让模型调用
  - ▶ 工具调用时工具描述 (Description) 和用户查询语句 (Query) 共同组成提示 (Prompt) 送给模型
  - ▶ 模型按照工具描述 (Description)，根据用户查询语句 (Query) 生成相应的工具调用命令 (Function Call)
- ▶ 使用工具的能力通过外挂插件 (Plut-ins) 方式 (而不是内嵌方式) 让模型使用
- ▶ 典型工作：
  - ▶ ChatGPT Plug-ins
  - ▶ TaskMatrix.AI

# TaskMatrix.AI



Liang, et al. "TaskMatrix.AI: Completing Tasks by Connecting Foundation Models with Millions of APIs." arXiv, March 28, 2023. <http://arxiv.org/abs/2303.16434>.

# 通过思维链(CoT)调用工具解决复杂问题(Agent, AutoGPT)

- ▶ 在模型已经具备使用单个工具的能力前提下，借助思维链（CoT）能力调用多个工具解决复杂问题：
  - ▶ 路径规划（Planning）：模型通过思维链将复杂问题分解成多个步骤，或者用户直接指定解决问题的分解步骤
  - ▶ 单步执行（Single Step）：模型依次调用工具完成每个步骤，以解决复杂问题。每个步骤结束后，模型可以通过某种方法验证该步骤执行是否正确：
    - ▶ 如果正确，则继续执行下一步骤（Going Forward）
    - ▶ 如果不正确，则重新生成另一条命令，再次执行当前步骤（Retry）
    - ▶ 如果不正确，而且当前步骤已经没有其他方法可以执行，那么回溯到上一步骤，重新规划路径。（Backtracking, Replanning）
- ▶ 典型工作：
  - ▶ WebShop：通过模仿学习学会访问多个网站完成购物行为
  - ▶ LangChain & AutoGPT

# WebShop

- ▶ WebShop为语言grounded提供了几个挑战，包括理解构成指令、重新表述查询、理解和处理网页上的嘈杂文本，以及进行策略探索。
- ▶ 给定一个文本指令，指定一个产品要求，代理需要遍历多种类型的网页，并发出不同的行动来找到、定制和购买商品。
- ▶ 收集了超过1600个人类演示，并使用强化学习、模仿学习和预训练的图像和语言模型对各种代理进行训练和评估。
- ▶ 最佳模型取得了任务成功率29%，这比基于规则的启发式方法(9.6%)要好，但远低于人类专家表现(59%)。
- ▶ 论文表明，在WebShop上训练的代理在amazon.com和ebay.com上表现出很好的模拟环境到真实环境的迁移能力，表明WebShop在开发可以实用的网页代理方面具有潜在价值。

Yao, et al. “WebShop: Towards Scalable Real-World Web Interaction with Grounded Language Agents.” arXiv2207.01206.

# WebShop

**A**

WebShop search

Instruction:  
I'm looking for a small portable folding desk that is already fully assembled; it should have a khaki wood finish, and price lower than 140.00 dollars

Search

1

2 results

Back to Search  
Page 1 (Total results: 50) [btn] Next [/btn]

MENHG Folding Breakfast Tray Table, Efficient Home Laptop Notebook Computer Desk, Portable Writing Study Desk, Sturdy Home Office Table Workstation \$109.0

KPSP Folding Study Desk Bed Breakfast Serving Tray Table Efficient Home Laptop Notebook Computer Desk Portable Standing Desk for Small Space Bedroom

3

4.1 4.2

Color black khaki white

4.3

MENHG Folding Laptop Table Bed Desk PC Lap Desk with Drawer Book Stand Reading Holder Leg Space Laptop Bed Tray Foldable Lazy Table Breakfast Desk Sofa Small Desk for Small Space

Price: \$109.0 Rating: 4.1 Buy Now 5

Description Overview Reward: 1.0

**B**

HTML mode

Simple mode

Instruction:  
I'm looking for a small portable folding desk that is already fully assembled [...]

[btn] Back to Search [/btn]  
Page 1 (Total results: 50) [btn] Next [/btn]  
[btn] MENHG Folding Breakfast Tray [...] [/btn]  
\$109.0  
[btn] KPSP Folding Study Desk Bed [...] [/btn]

**C**

$\bar{u}$  (Instruction): I'm looking for a small portable...

$\bar{y}$  (Description): MENHG Folding Laptop Table Bed...

$y_{price}$ : \$109.0

$Y_{opt}$  (Options): { black, khaki, white }

$Y_{att}$  (Attributes): { steel pipe, no assembly, portable }

The WebShop environment.

A: An example task trajectory in HTML mode, where a user can

- (1) search a query in a search page,
- (2) click a product item in a results page,
- (3) choose a color option in a item page,
- (4) check item-detail pages and go back to the item page, and
- (5) finally buy the product to end the episode and receive a reward  $r \in [0, 1]$  (§3.2).

B: the results page in simple mode for agent training and evaluation. The blue text indicates clickable actions and bold text indicates an action selected by the agent.

C: The product notation used in §3 with corresponding examples from the product in A. The attributes  $Y_{att}$  are hidden from the task performer.

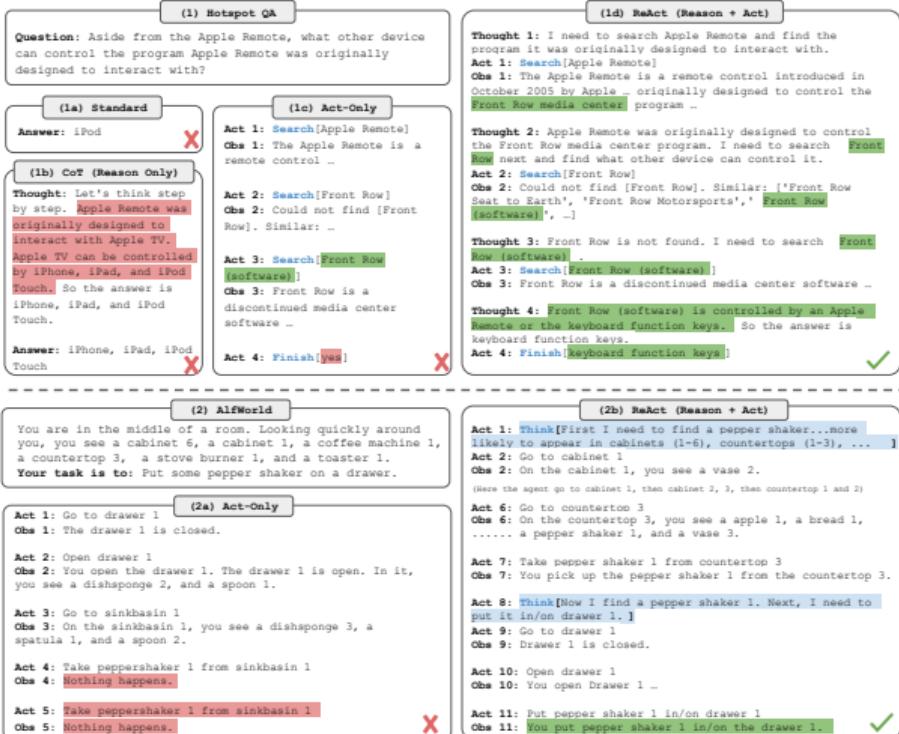
Yao, et al. “WebShop: Towards Scalable Real-World Web Interaction with Grounded Language Agents.” arXiv2207.01206.

# ReAct

- ▶ ReAct是一个将推理和动作与语言模型结合起来的通用范式，用于解决语言推理和决策任务
- ▶ 考虑一个agent与环境互动以解决任务的一般设置。在时间步 $t$ ，agent从环境中收到一个观察值 $o_t \in \mathcal{O}$ ，并按照一些策略 $\pi(a_t|c_t)$ 采取动作，其中 $c_t = (o_1, a_1, \dots, o_t)$ 是agent的背景。
- ▶ ReAct将agent的动作空间扩大到 $\dot{\mathcal{A}} = \mathcal{A} \cup \mathcal{L}$ ，其中 $\mathcal{L}$ 是语言空间。语言空间中的动作 $\hat{a}_t \in \mathbb{L}$ ，我们将其称为思想或推理轨迹，不影响外部环境，因此不会导致观察反馈。相反，一个思想 $\hat{a}_t$ 的目的是通过对当前上下文 $c_t$ 的推理来组成有用的信息，并更新上下文 $c_{t+1} = (c_t, \hat{a}_t)$ 以支持未来的推理或动作。
- ▶ ReAct使用语言模型以交错的方式产生与任务相关的语言推理轨迹和动作，这使得模型能够进行动作态推理，以创建、维护和调整动作的高级计划（推理到动作），同时也与外部环境（例如维基百科）互动，将额外信息纳入推理（动作到推理）。
- ▶ ReAct在四个不同的基准上和最先进的基准进行了实证评估：问答（HotPotQA）、事实验证（Fever）、基于文本的游戏（ALFWorld）和网页导航（WebShop）。
- ▶ 对于HotPotQA和Fever，通过访问模型可以交互的维基百科API，ReAct的性能优于普通的动作生成模型，与思维链推理（CoT）可比。总体而言，最好的方法是ReAct和CoT的结合，允许在推理过程中使用内部知识和外部获得的信息。
- ▶ 在ALFWorld和WebShop上，通过一到两个样例，ReAct提示能够胜过用103~105个任务实例训练的模仿或强化学习方法，成功率的绝对值分别提高了34%和10%。

Yao, et al. “ReAct: Synergizing Reasoning and Acting in Language Models.” arXiv2210.03629.

# ReAct



(1) Comparison of 4 prompting methods,  
(a) Standard,  
(b) Chain-of-thought(CoT), Reason Only,  
(c) Act-only, and  
(d) ReAct (Reason+Act),  
solving a HotpotQA question.

(2) Comparison of

(a) Act-only and  
(b) ReAct prompting  
to solve an AlfWorld game.

In both domains, we omit in-context examples in the prompt, and only show task solving trajectories generated by the model (Act, Thought) and the environment (Obs).

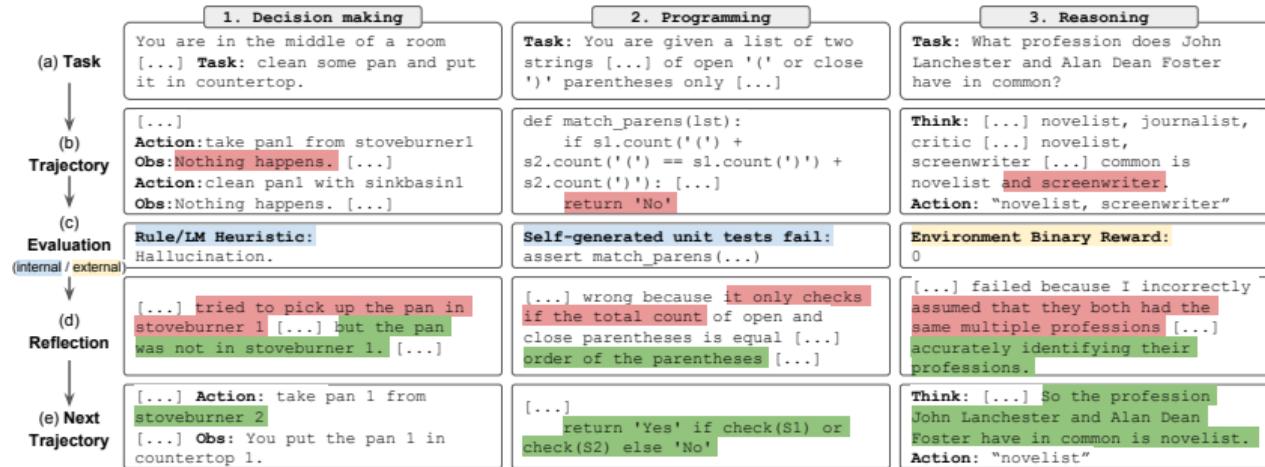
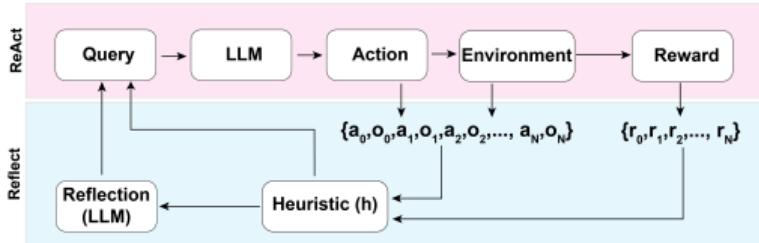
Yao, et al. "ReAct: Synergizing Reasoning and Acting in Language Models." arXiv2210.03629.

# Reflexion

- ▶ 基于ReAct方法，提出了一种反思(Reflexion)框架，通过语言反馈来强化语言代理，而不是更新权重
- ▶ 具体来说，反思代理以文字形式对任务反馈信号进行反思，然后在一个连续记录的内存缓冲区中维护自己的反思文本，以在后续试验中诱导更好的决策
- ▶ Reflexion方法的自我反思过程受到启发性函数的监督。在这个过程中，模型会在有限的次数内尝试修正错误，从而促使其产生更多具有创新性的解决方案
- ▶ Reflexion应用性很强，可以提高决策和知识密集型任务的试错性能，只需要依赖一个二元的奖励模型。它应用在更复杂任务中，可以帮助智能体学会新想法，探索更大不可见的状态空间，并通过在过去环境中的经验形成更准确的行动规划
- ▶ Reflexion可以集成各种类型(数值或自由文本)和来源(外部的或内部模拟的)的反馈信号，并在各种任务上获得显著的改进(顺序决策制定、编码、语言推理)
- ▶ Reflexion在HumanEval编码基准上实现了91%的pass@1准确度，超过了之前的最先进的GPT-4(80%)

Shinn, et al. “Reflexion: Language Agents with Verbal Reinforcement Learning.” arXiv2303.11366.

# Reflexion



Shinn, et al. “Reflexion: Language Agents with Verbal Reinforcement Learning.” arXiv2303.11366.

# Content

大语言模型调用外部工具 LLMs calling external tools

概览 Overview

方法 Methods

项目 Projects

我们的工作 Our Work

# LangChain

- ▶ LangChain是一个用于开发由语言模型驱动的应用程序的框架。
- ▶ LangChain主要拥有2个能力：
  - ▶ 可以将 LLM 模型与外部数据源进行连接
  - ▶ 允许与 LLM 模型进行交互
- ▶ Langchain已经成为目前使用大语言模型最热门的框架之一。

LANGCHAIN 🦜 🌐

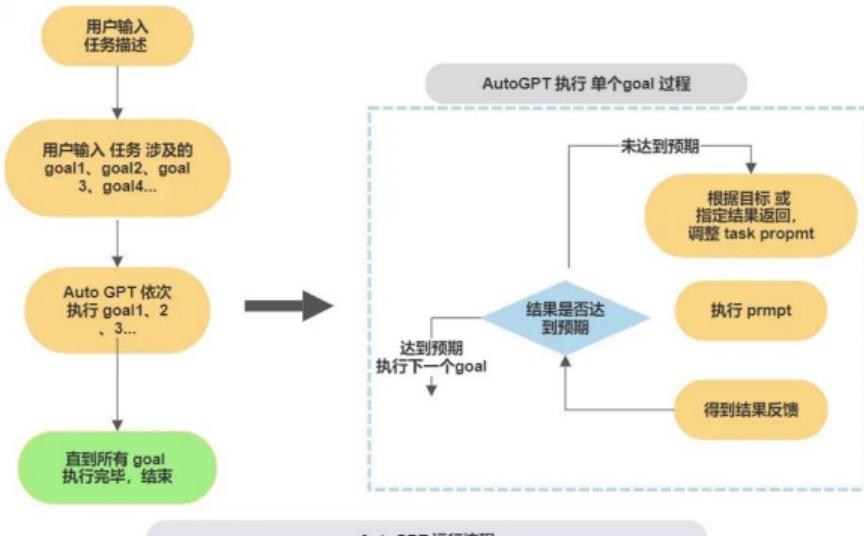


<https://datasciencedojo.com/blog/understanding-langchain/>

# AutoGPT

- ▶ Auto-GPT是一个开源应用程序，它使用LLM（如OpenAI的大型语言模型GPT-4）来自动执行多步骤项目，来完成复杂任务。
- ▶ 它允许LLM自主行动，而无需人工代理来提示其每项行动。
- ▶ Auto-GPT可以生成类似人类的文本、回答问题、翻译语言、总结文本并提供建议等任务。

 AUTOGPT <https://autogpt.net/>



<https://aitechtogether.com/python/101823.html>

# Content

大语言模型调用外部工具 LLMs calling external tools

概览 Overview

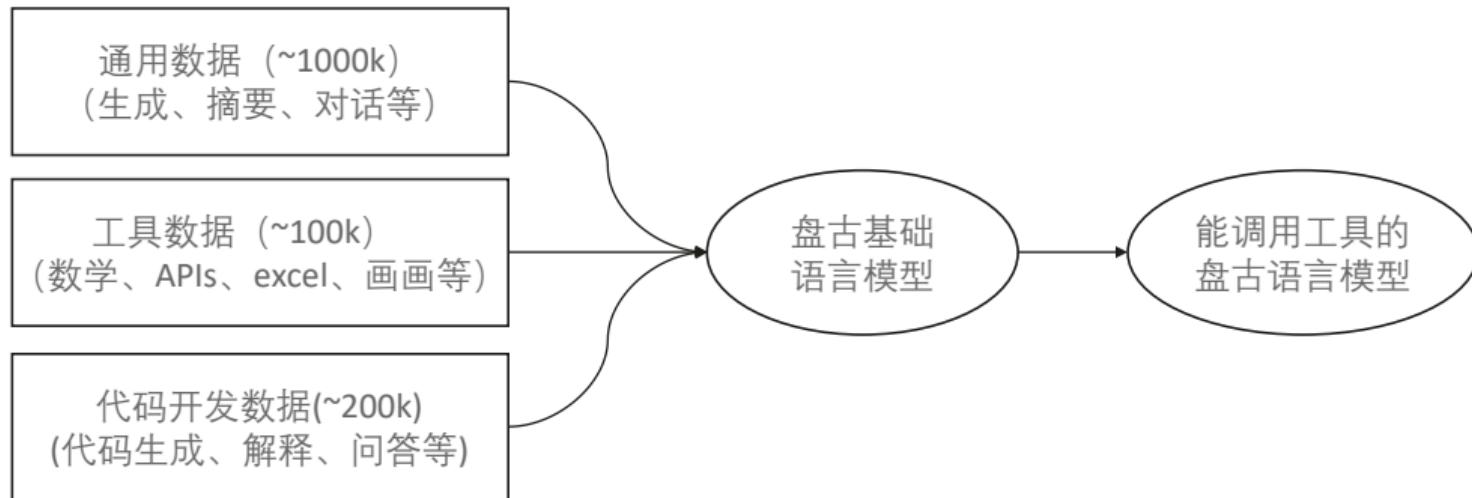
方法 Methods

项目 Projects

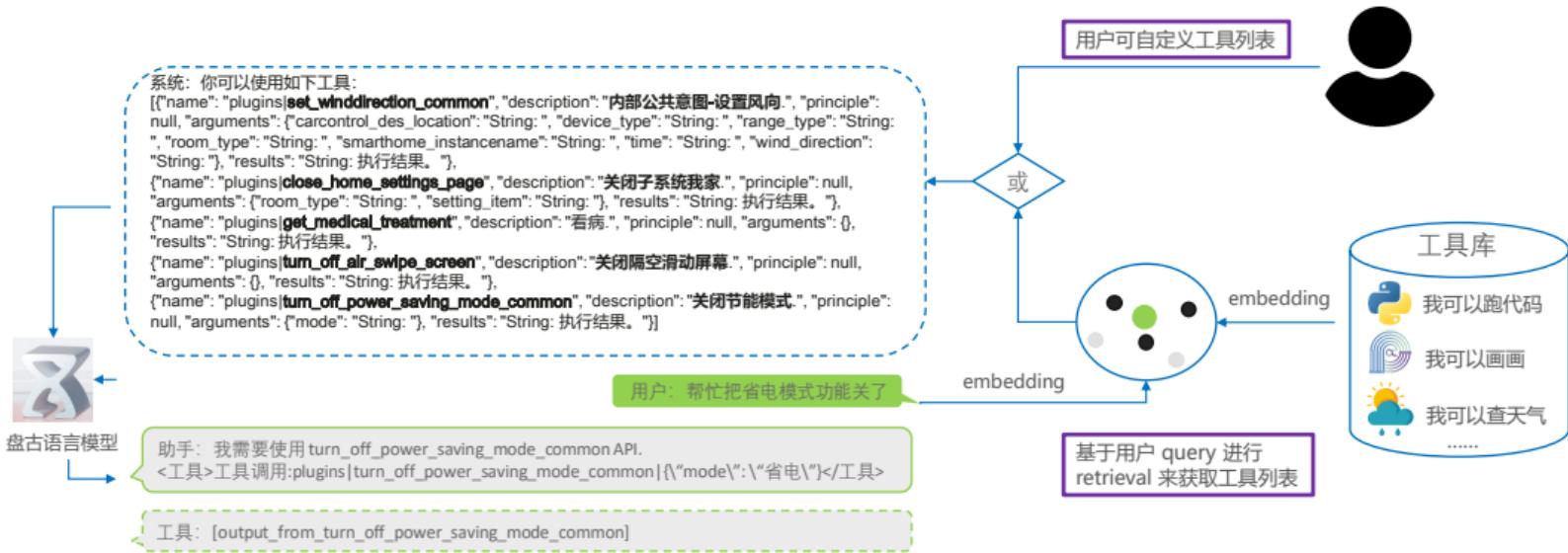
我们的工作 Our Work

# 通过有监督微调SFT实现外部工具调用

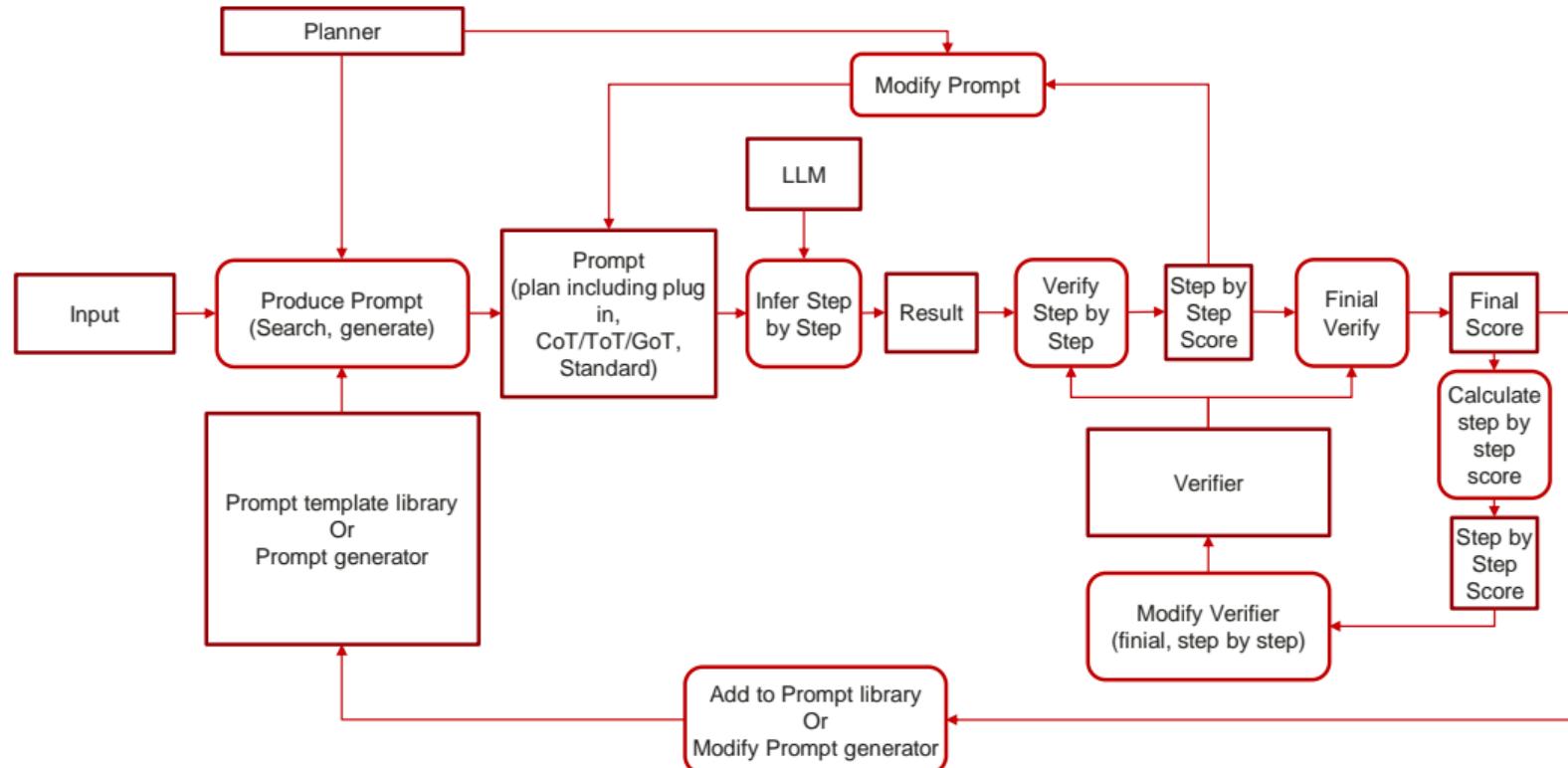
- > 通用数据进行SFT，维持基础通用能力
- > 代码开发数据SFT，夯实工具调用基础
- > 多种工具数据SFT，打造办公生活助手



# 通过上下文学习ICL实现外部工具调用

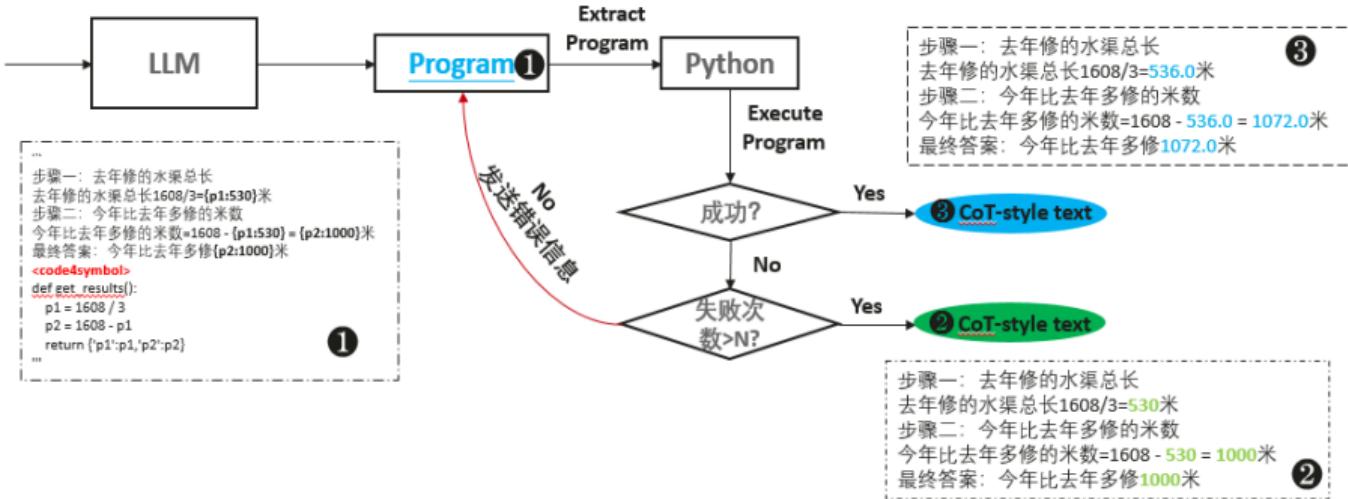


# 通过思维链CoT实现外部工具调用的自动规划和搜索



# 数学推理

Q: “老5屯今年修了4条水渠，总长1608米，等于去年修的3倍。今年比去年多修多少米？”



# 数学推理

## ▶ 方法步骤：

1. 生成文本代码混杂的结果，其中`<code4symbol>` tag用来区别其他任务，这个tag会触发系统调用python解释器；
2. 调用python解释器执行代码，执行成功，计算结果填回文本；
3. 如果执行出错，使用模型生成的缺省结果；
4. 多次执行特性（开发中）：如果执行出错，python将错误信息发送给大模型再次生成，达到自动修复的目标。

## ▶ 实验结果：

- ▶ 加入python解释器，在数学类任务上模型得到了显著提升（35%→86.7%）。

# 数学推理：样例

Q1:  $(100+(200/5)-99) / 4$

依据题意得:  $(100+(200/5)-99)/4$

调用Python代码解释器:

```
1 |     def get_result():
2 |         return{"p1":(100+(200/5)-99)/4}
```

故结果为10.25

Q2: 一个圆锥体，它的底半径是3厘米，高是9厘米，它的体积是多少？

依据题意得:  $(1/3)*3.14*3**2*9$

调用Python代码解释器:

```
1 |     def get_result():
2 |         return{"p1":(1/3)*3.14*3**2*9}
```

故结果为84.78

# 数学推理：样例

Q3: 小红买了20.5米的布，做衣服用了10.5米，请问还剩下多少厘米的布？

依据题意得： $(20.5-10.5)*100$

调用Python代码解释器：

```
1 |     def get_result():
2 |         return{"p1":(20.5-10.5)*100}
```

故结果为1000

Q4: 小胡每小时走10公里，小红每小时走5公里，小红在前，小胡在后，两人相距20公里，同时出发，请问小胡多长时间能追上小红？

依据题意得： $20/(10-5)$

调用Python代码解释器：

```
1 |     def get_result():
2 |         return{"p1":20/(10-5)}
```

故结果为4

# 数学推理：样例

Q5:有5筐苹果的重量相等，如果从每筐中取出10kg，那么剩下的苹果相当于原来3筐的重量，原来每筐苹果重多少千克？

依据题意得： $10*5/(5-3)$

调用Python代码解释器：

```
1 |     def get_result():
2 |         return{"p1":10*5/(5-3)}
```

故结果为25

Q6:二十加五的和乘四除以十等于

依据题意得： $(20+5)*4/10$

调用Python代码解释器：

```
1 |     def get_result():
2 |         return{"p1":(20+5)*4/10}
```

故结果为10

# 表格处理

## 表格增删改查复杂操作

- 基于盘古语言模型代码能力生成pandas代码完成表格操作



## 数据集多元透视分析

- 生成代码借助matplotlib/seaborn等工具进行数据分析



# 表格处理：样例

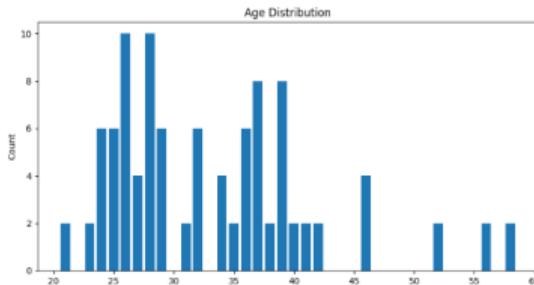
First Name	Last Name	Gender	Country	Age	Date	Id
Dulce	Abril	Female	United States	32	15/10/2017	1562
Mara	Hashimoto	Female	Great Britain	25	16/08/2016	1582
Philip	Gent	Male	France	36	21/05/2015	2587
Kathleen	Hanner	Female	United States	25	15/10/2017	3549
Nereida	Magwood	Female	United States	58	16/08/2016	2468
Gaston	Brumm	Male	United States	24	21/05/2015	2554
Etta	Hurn	Female	Great Britain	56	15/10/2017	3598
Earlean	Melgar	Female	United States	27	16/08/2016	2456
Vincenza	Weiland	Female	United States	40	21/05/2015	6548
Fallon	Winward	Female	Great Britain	28	16/08/2016	5486
Arcelia	Bouska	Female	Great Britain	39	21/05/2015	1258
Franklyn	Unknow	Male	France	38	15/10/2017	2579
Sherron	Ascencio	Female	Great Britain	32	16/08/2016	3256
Marcel	Zabriskie	Male	Great Britain	26	21/05/2015	2587
Kina	Hazelton	Female	Great Britain	31	16/08/2016	3259
Shavonne	Pia	Female	France	24	21/05/2015	1546
Shavon	Benito	Female	France	39	15/10/2017	3579

上一步下步

pengu-file 2023-07-18 10:17:12

## 第一步：上传表格

绘制图片如下：



```
1 import pandas as pd
2 import matplotlib.pyplot as plt
3 df = pd.read_excel('data.xlsx')
4 #统计年龄分布
5 age_count = df['Age'].value_counts()
6 #绘制年龄分布图
7 plt.figure(figsize=(10,5))
8 plt.bar(age_count.index, age_count.values)
9 plt.xlabel('Age')
10 plt.ylabel('Count')
11 plt.title('Age Distribution')
12 plt.savefig('output_0.png')
```

## 第二步：表格分析：年龄分布图

# 日历查询：样例

2023年1月1日已经过去多少天了？

我们可以通过调用工具得到结果。

```
1  from datetime import datetime
2  def get_result():
3
4
5      #今天的日期
6      today = datetime.today()
7
8      #2023年1月1日的日期
9      past_date = datetime(2023,1,1)
10
11     #计算两个日期之间的天数
12     days_passed =(today - past_date).days
13
14     return f"2023年1月1日已经过去{days_passed}天"
```

复制

2023年1月1日已经过去197天

# 图片生成：样例

请画一张清明上河图风格的图片

好的，我将为您生成一张描述清明上河图的图片。请稍等，正在调用盘古画画2.0-艺术进阶版进行作画。



好的，我已经生成了一张描述清明上河图的图片。请查看并确认是否需要进行修改。

# 数学推理中的多轮工具调用：样例

北京到深圳总共2032公里，火车的速度是212公里每小时，飞机的速度是650公里每小时，请问哪个快？快的比慢的节省多少时间？

这是一个应用题，我们可以通过对计算火车和飞机的行驶时间来比较两者的速度。

火车的速度是212公里每小时，所以它需要的时间是行驶距离除以速度，即 $2032\text{公里} \div 212\text{公里/小时} =$

```
1 def get_result():
2     return (*x*: 2032/212)
```

9.58小时。

飞机的速度是650公里每小时，所以它需要的时间是行驶距离除以速度，即 $2032\text{公里} \div 650\text{公里/小时} =$

```
1 def get_result():
2     return (*x*: 2032/650)
```

3.13小时。

所以，飞机比火车快，快的时间是火车的时间减去飞机的时间，即 $9.58\text{小时} - 3.13\text{小时} =$

```
1 def get_result():
2     return (*x*: 9.58-3.13)
```

6.45小时。

所以，飞机比火车快6.45小时。

**第一步：精确算出火车时间。**

**第二步：精确算出飞机时间。**

**第三步：精确算出飞机比火车节省的时间。**

# 基于工具反馈的自我修复：样例

$$2/(\sqrt{7}+\sqrt{5})$$

依据题意得：

```
1 | def get_result():
2 |     return {"p1":2/(\sqrt{7}+\sqrt{5})}
```

复制

代码出错了，错误信息为：

```
"run_python", line 3
    return {"p1":2/(\sqrt{7}+\sqrt{5})}
               ^
SyntaxError: invalid character '\u221a' (U+221A)
```

语法错误，应该是：

```
1 | def get_result():
2 |     return {"p1": 2/(math.sqrt(7)+math.sqrt(5))}
```

复制

计算结果为:0.41

# 基于工具反馈的例外处理：样例

2023年2月30日是星期几？

我们可以调用工具得到结果。

```
1 from datetime import datetime, timedelta
2 def get_result():
3
4     #设置年份
5     year = 2023
6
7     #设置月份
8     month = 2
9
10    #设置日期
11    day = 30
12
13    #创建日期对象
14    date_obj = datetime(year, month, day)
15
16    #获取星期几
17    weekday = date_obj.weekday()
18
19    #输出星期几
20    weekdays = ["星期一", "星期二", "星期三", "星期四", "星期五", "星期六", "星期日"]
21    return weekdays[weekday]
22
23    </code> </pre> 代码出错了，错误信息为：
24
25    "run_python", line 15, in get_result
26    ValueError: day is out of range for month
27
根据错误信息，可以得知，2023年2月只有28天或29天，没有30天。因此，2023年2月30日是不存在的，也就无法确定星期几。
```

# 模型知识与工具调用结合：样例

非洲总面积月3020万平方千米，是全球仅次于亚洲的面积第二大洲，它和亚洲面积相差多少？

这是一个简单的减法问题。非洲的面积是3020万平方千米，**亚洲的面积是4450万平方千米**。所以，非洲的面积是亚洲面积的约76%。我们可以通过以下公式计算**亚洲面积减去非洲面积**：

亚洲面积-非洲面积=亚洲面积-3020万平方千米

第一步：模型得到亚洲面积。

```
1 | def get_result():
2 |     return {"x": 4450-3020}
```

复制

通过计算得到，亚洲面积减去非洲面积的差是1430万平方千米。

第二步：调用工具得到结果。

# 模型知识与工具调用结合：样例

问题：100以内的质数之和是多少？

这是一个关于数学的问题。首先，我们需要知道什么是质数。质数是指大于1的自然数，除了1和它本身，没有其他的因数。所以，2、3、5、7、11、13、17、19、23、29、31、37、41、43、47、53、59、61、67、71、73、79、83、89、97是前100个质数。

然后，我们需要计算这些质数的和。这是一个简单的加法问题，可以直接将这些质数相加。

2+3+5+7+11+13+17+19+23+29+31+37+41+43+47+53+59+61+67+71+73+79+83+89+97=

```
1 | def get_result():
2 |     return {"x":sum([2,3,5,7,11,13,17,19,23,29,31,37,41,43,47,53,59,61,67,71,73,79,83,89,97])}
```

复制

1060。

所以，前100个质数的和是1060。

# Content

背景 Background

大语言模型结合搜索引擎 LLMs combined with search engines

大语言模型调用外部工具 LLMs calling external tools

总结 Summary

# 总结 Summary

- ▶ 大语言模型具备强大的语言能力，但在实际应用中，仍然面临知识不足、更新不及时、推理能力差、与外部环境无法交互等缺陷，而融合搜索和调用工具是解决这些问题的有效方法；
- ▶ 介绍了大语言模型融合搜索引擎的主要方法；
- ▶ 结合搜索引擎可以较好地解决模型知识不足和更新不及时等问题，但真实性问题还无法彻底解决，也还面临很多新问题亟待研究；
- ▶ 介绍了大语言模型调用工具的三种进阶方法；
- ▶ 调用外部工具可以大大增强大语言模型的各种能力，通过智能体方式调用多种工具甚至可以解决一些复杂问题，这一领域还有很大探索空间；
- ▶ 介绍了华为诺亚方舟实验室在这两方面的工作。

# Thank you!

把数字世界带入每个人、每个家庭、  
每个组织，构建万物互联的智能世界。

Bring digital to every person, home and organization  
for a fully connected, intelligent world.

Copyright©2018 Huawei Technologies Co., Ltd.  
All Rights Reserved.

The information in this document may contain predictive statements including, without limitation, statements regarding the future financial and operating results, future product portfolio, new technology, etc. There are a number of factors that could cause actual results and developments to differ materially from those expressed or implied in the predictive statements. Therefore, such information is provided for reference purpose only and constitutes neither an offer nor an acceptance. Huawei may change the information at any time without notice.

