Link to the slides

RAISE Workshop 2025

(A Lightning Introduction to) Foundation Models and Agents in NLP

Yu Zhang

Assistant Professor, Department of Computer Science & Engineering

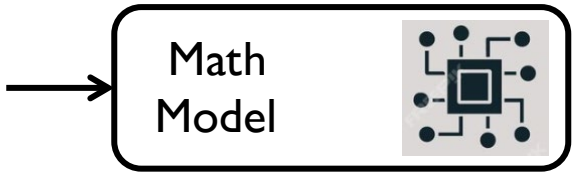
yuzhang@tamu.edu

May 12, 2025

NLP before the Era of Large Language Models

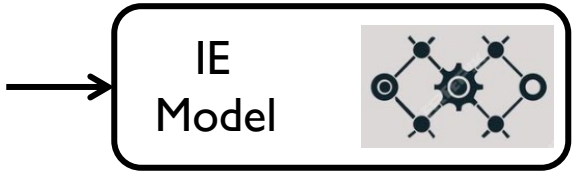
- Given an NLP task, design a task-specific model → annotate task-specific training data → using the annotated data to train the model.

Math: (*John buys 20 cards and 1/4 are uncommon. How many uncommon cards did he get?*, *5*)

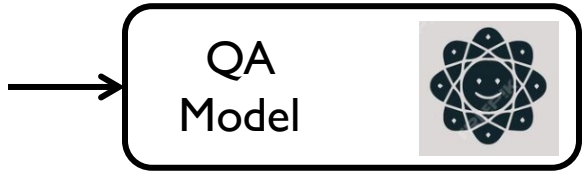


Classification: (*Rapid chromatographic technique for preparative separations with moderate resolution*, *Organic Chemistry*)

Information Extraction: (*in rats, nitrofurantoin causes pulmonary toxicity*, *[pulmonary toxicity, DISEASE]*)



Question Answering: (*Who formulated the zeroth law of thermodynamics?*, *Ralph H. Fowler*)



GPT-3: A Unified Model for Most NLP Tasks

- Only one task – next token prediction – on web-scale corpora

$$L_1(\mathcal{U}) = \sum_i \log P(u_i | u_{i-k}, \ldots, u_{i-1}; \Theta)$$

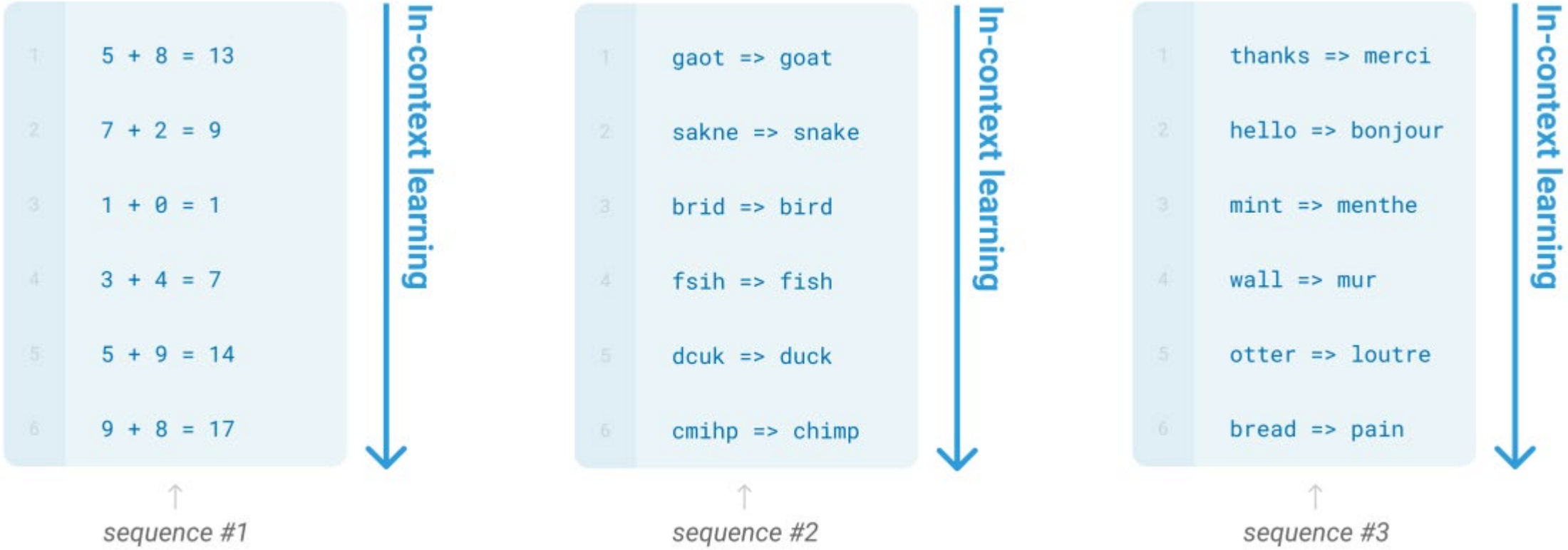
next token | previous tokens | model parameters

- Most NLP tasks can "reduce" to next token prediction.
  - *Math*: John buys 20 cards and 1/4 are uncommon. How many uncommon cards did he get? The answer is 11
  - *Classification*: (paper title) Rapid chromatographic technique for preparative separations with moderate resolution => (label) Organic Chemistry
  - *Information Extraction*: (text) in rats, nitrofurantoin causes pulmonary toxicity. => (entity, type) pulmonary toxicity, disease
  - *Question Answering*: Who formulated the zeroth law of thermodynamics? Ralph H. Fowler

Language Models are Few-Shot Learners. NeurIPS 2020.

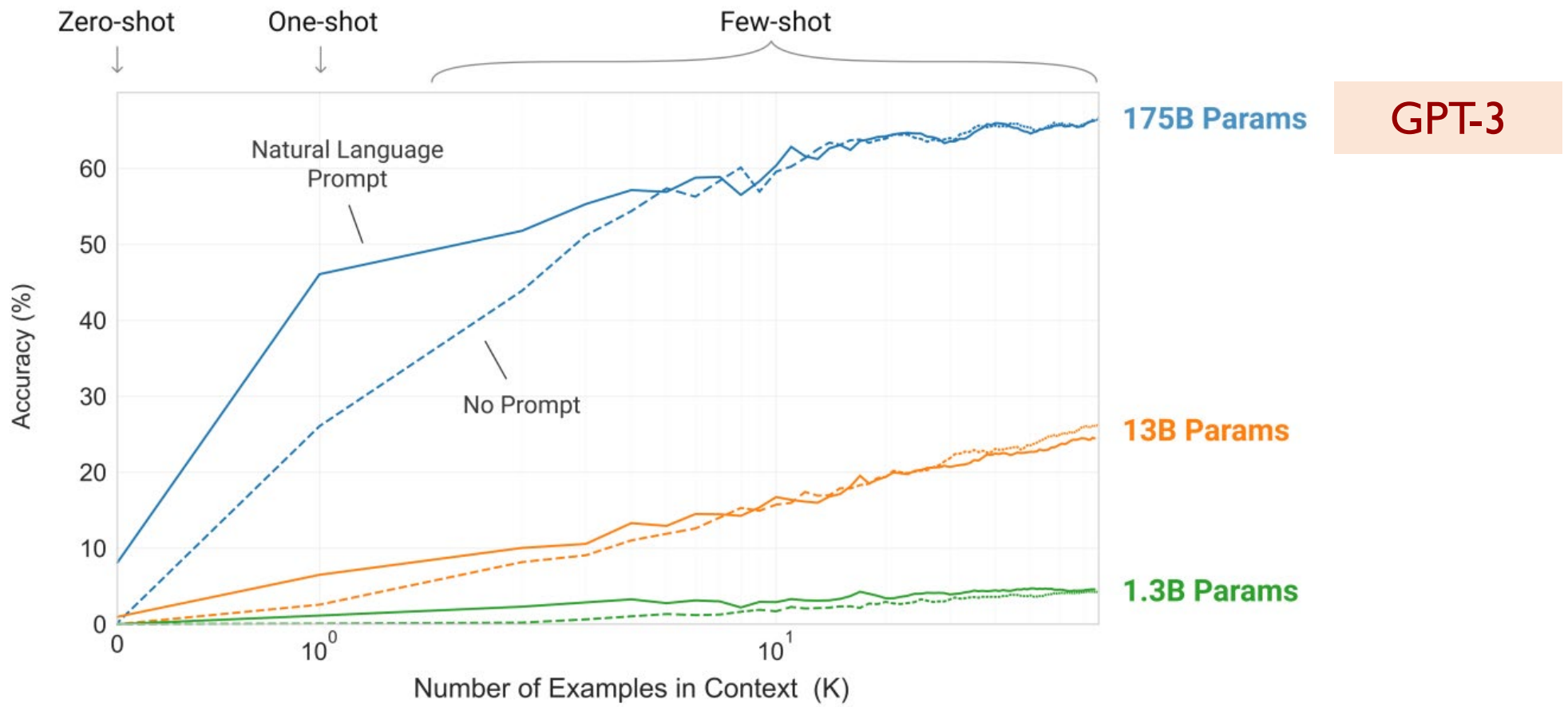
GPT-3: Performing Each Task with Just a Few Examples

- The model may acquire a broad set of skills and pattern recognition abilities during pre-training. It then uses these abilities at inference time to rapidly adapt to or recognize the desired task. – "In-context learning"



Can a model be that "smart"?

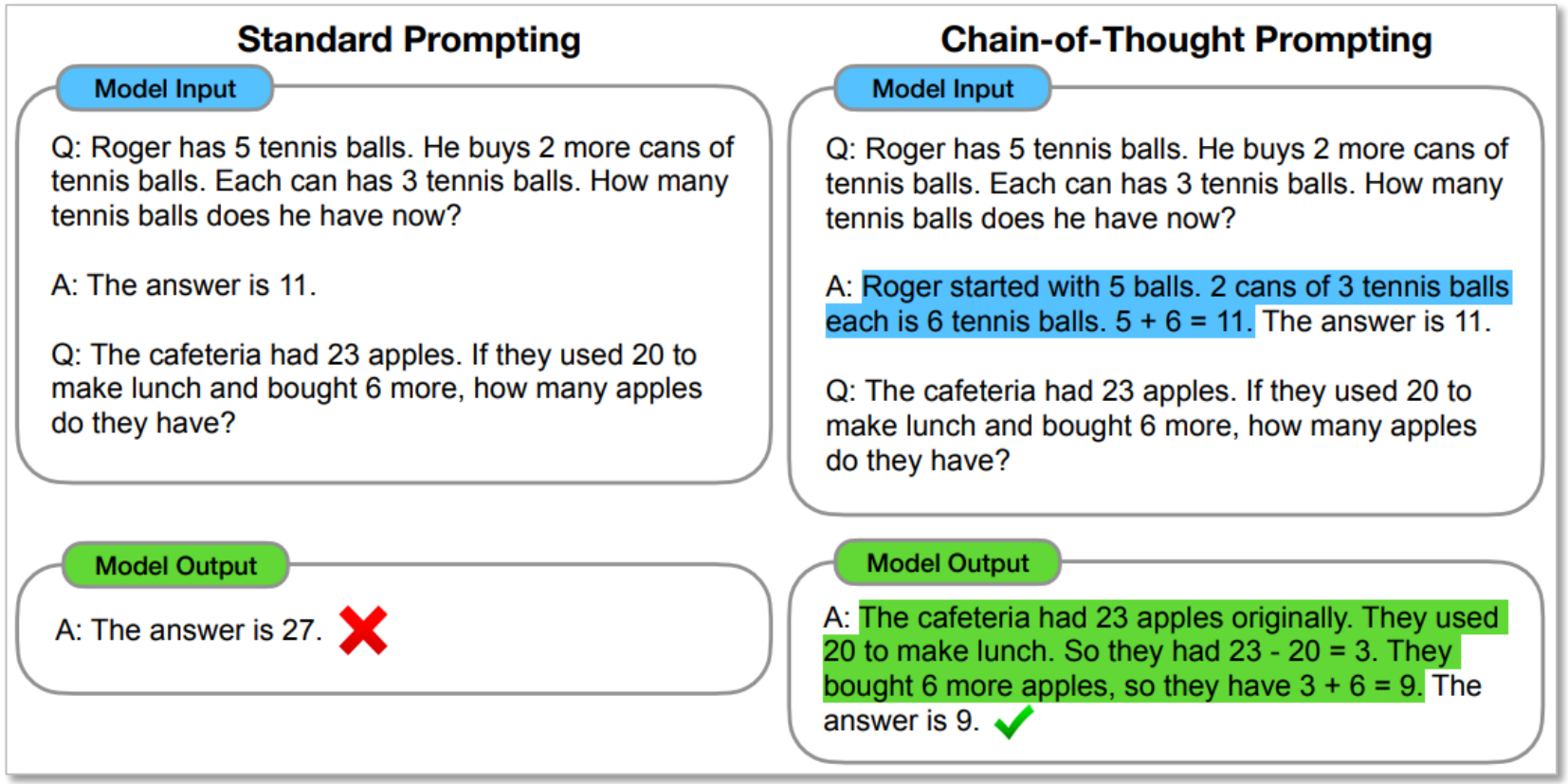
- Only if it is big enough!



Language Models are Few-Shot Learners. NeurIPS 2020.

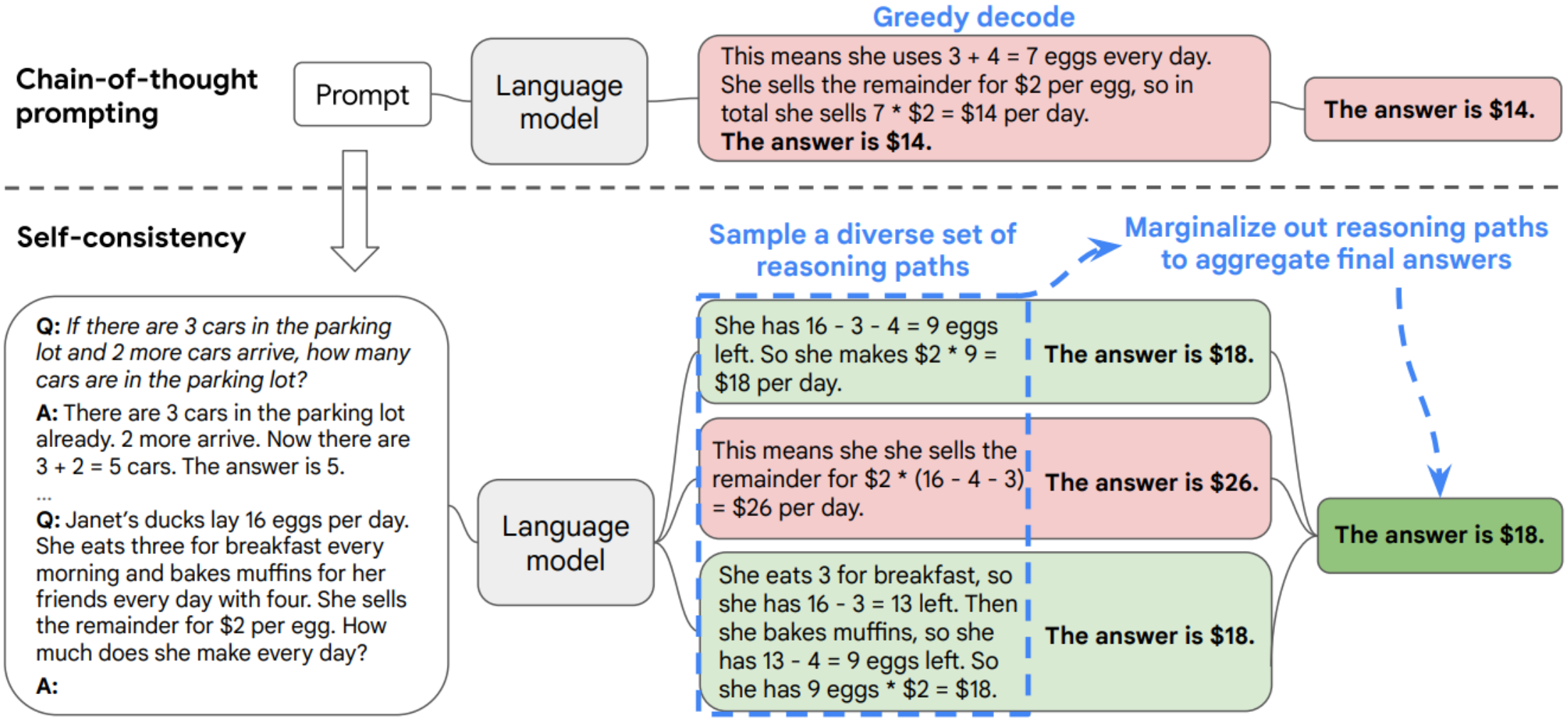
Inference-Time Technique: Chain-of-Thought Prompting

- Add a series of intermediate reasoning steps in the demonstration examples(s)
- Get the model to explain its reasoning steps before making an answer



Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. NeurIPS 2022.

Inference-Time Technique: Self-Consistency / Majority Voting



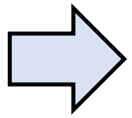
Self-Consistency Improves Chain of Thought Reasoning in Language Models. ICLR 2023.

From Few-Shot to Zero-Shot

Few-shot

In addition to the task description, the model sees a few examples of the task. No gradient updates are performed.

```
1  Translate English to French:        ← task description
2  sea otter => loutre de mer          ← examples
3  peppermint => menthe poivrée        ←
4  plush girafe => girafe peluche      ←
5  cheese =>                           ← prompt
```



Zero-shot

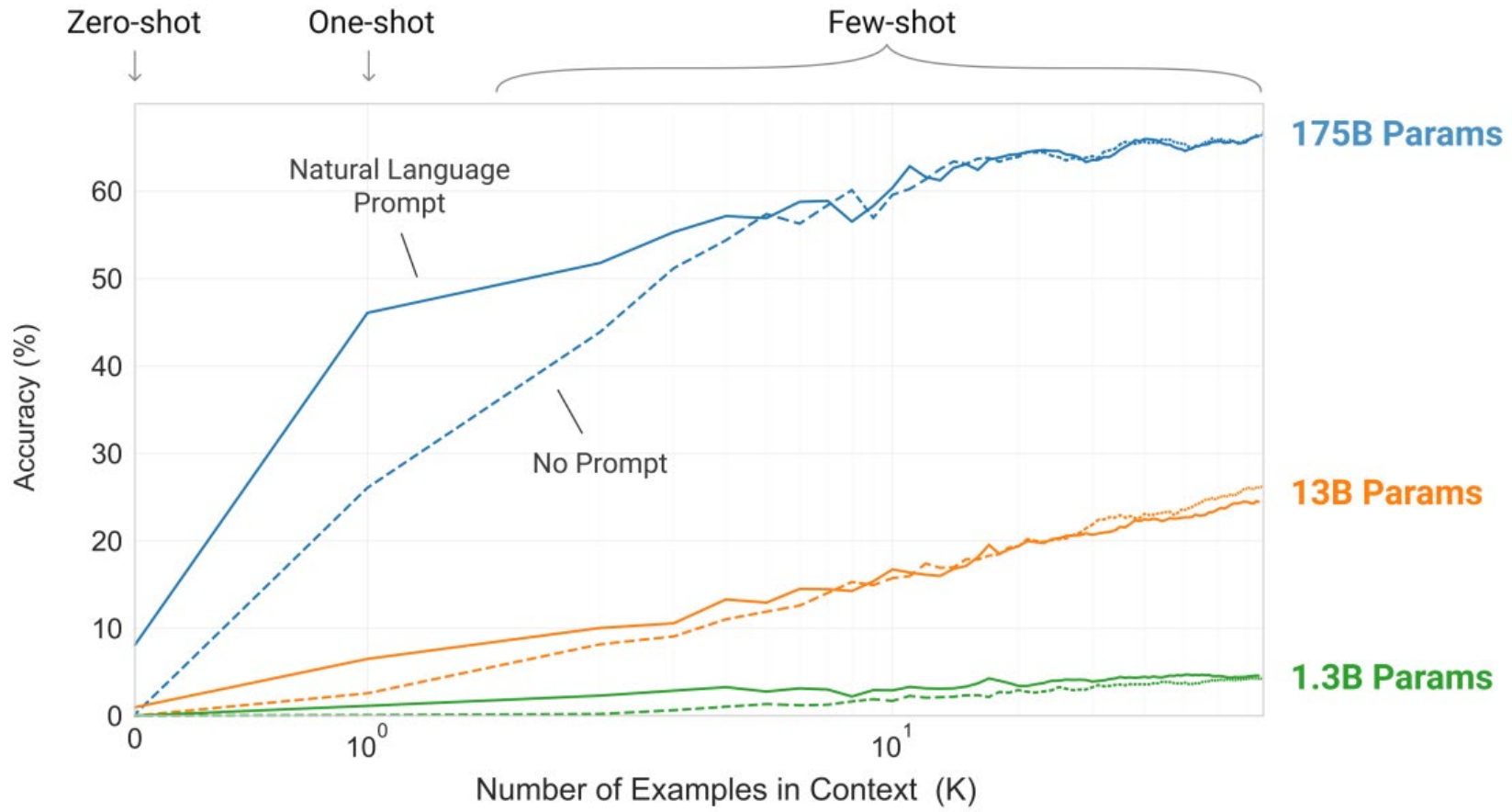
The model predicts the answer given only a natural language description of the task. No gradient updates are performed.

```
1  Translate English to French:        ← task description
2  cheese =>                           ← prompt
```

Is GPT-3 a zero-shot learner?

Task Instruction Only

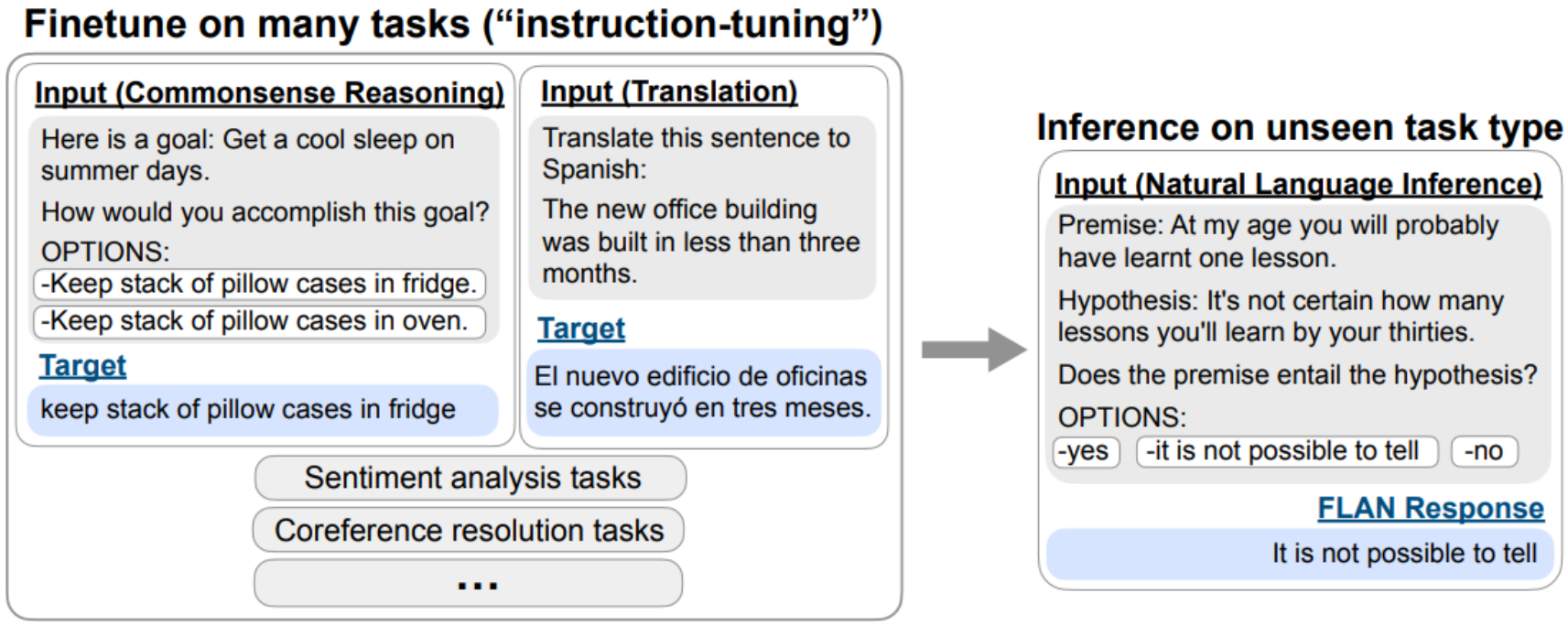
Task Instruction + A Few Examples



Language Models are Few-Shot Learners. NeurIPS 2020.

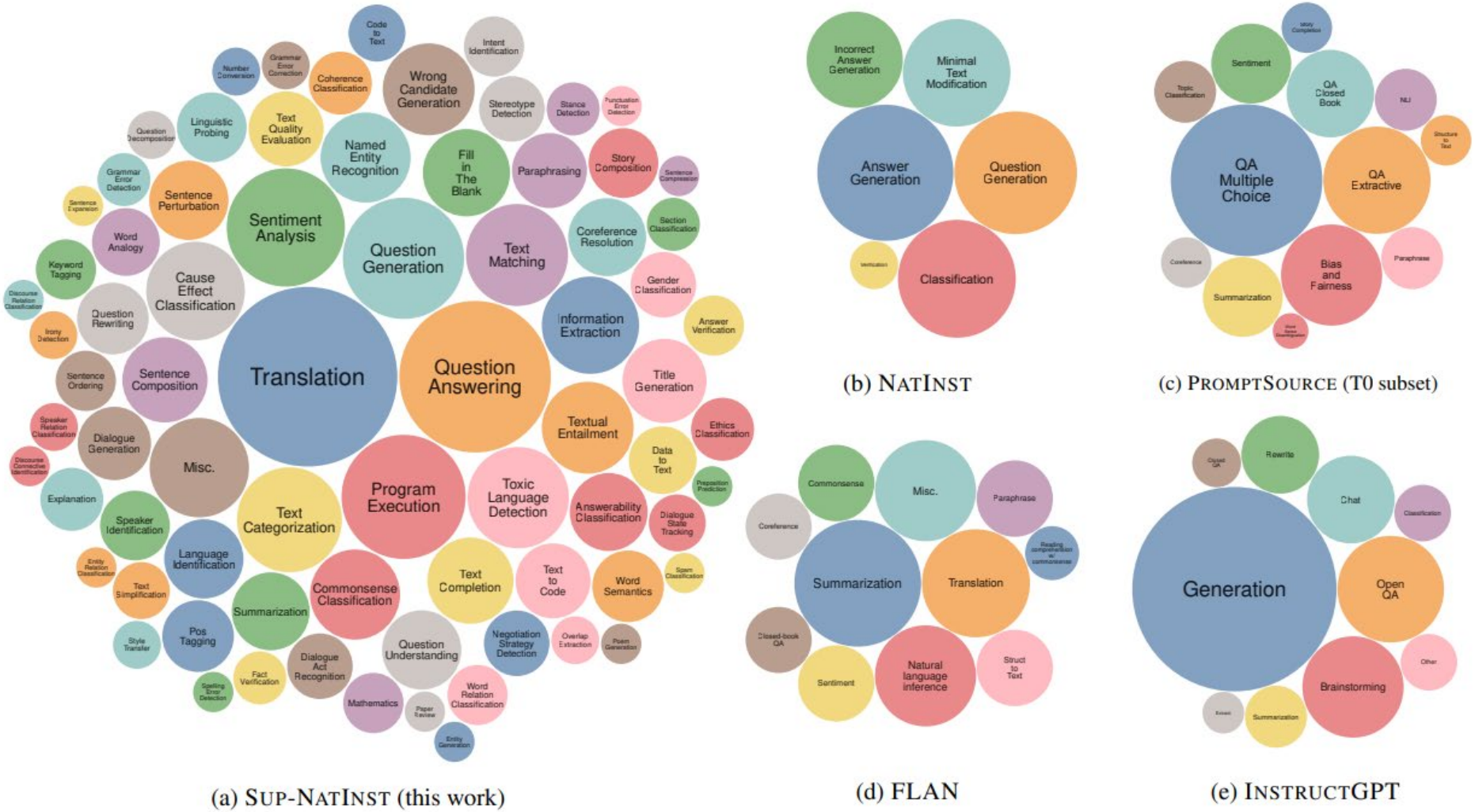
Why is the zero-shot setting hard?

- GPT-3 is not good at following an instruction to perform a new task.
- How to solve this problem?
  - Train the model to follow task instructions! → GPT-3.5



Finetuned Language Models Are Zero-Shot Learners. ICLR 2022.
Training Language Models to Follow Instructions with Human Feedback. NeurIPS 2022.

Instruction Tuning: A Competition of Data Collection/Annotation

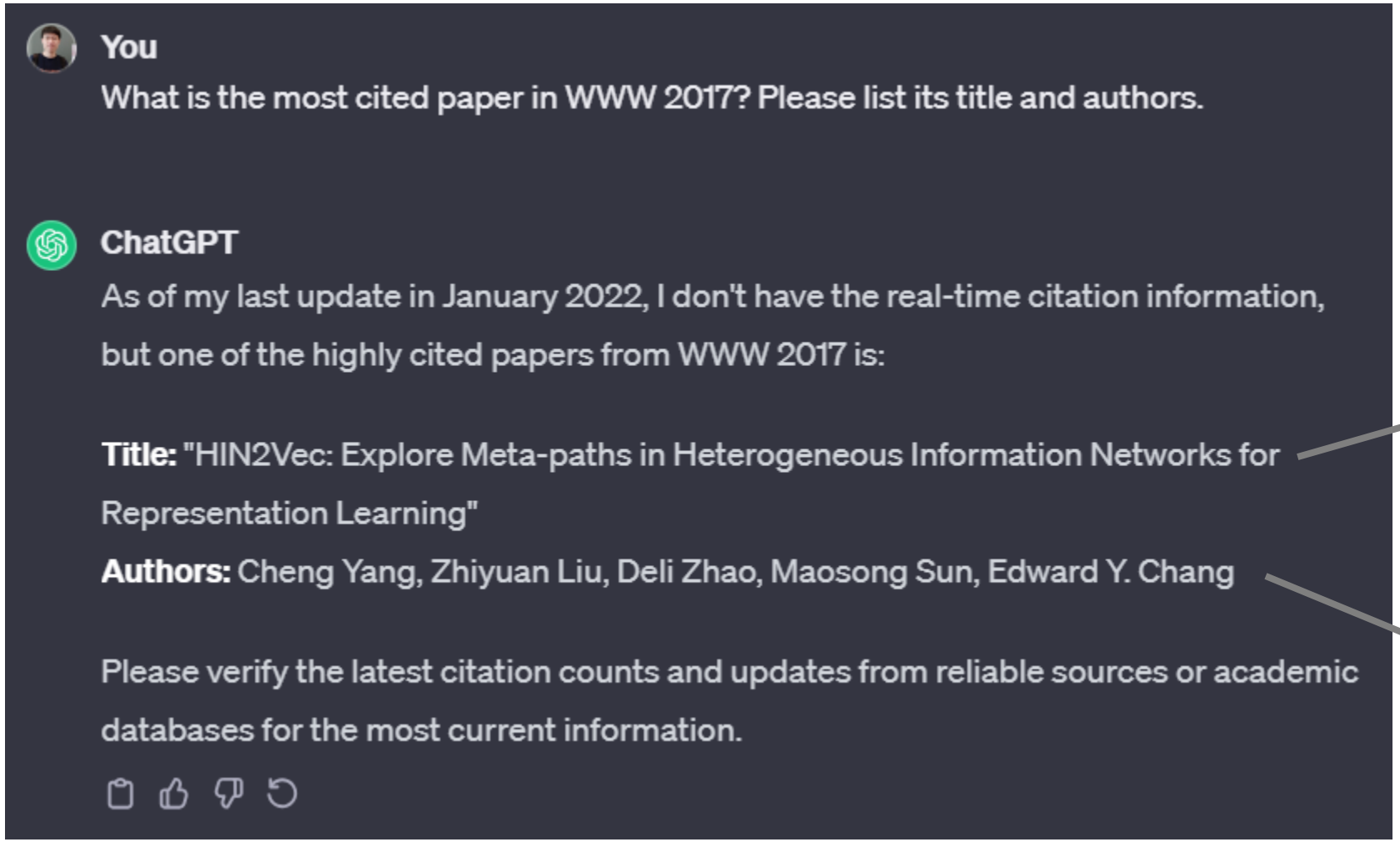


(a) SUP-NATINST (this work) (b) NATINST (c) PROMPTSOURCE (T0 subset)

(d) FLAN (e) INSTRUCTGPT

Super-NaturalInstructions: Generalization via Declarative Instructions on 1600+ NLP Tasks. EMNLP 2022.

What if an LLM does not have the knowledge to answer a question?



HIN2Vec was published in CIKM 2017.

HIN2Vec was written by Tao-yang Fu, Wang-Chien Lee, and Zhen Lei.

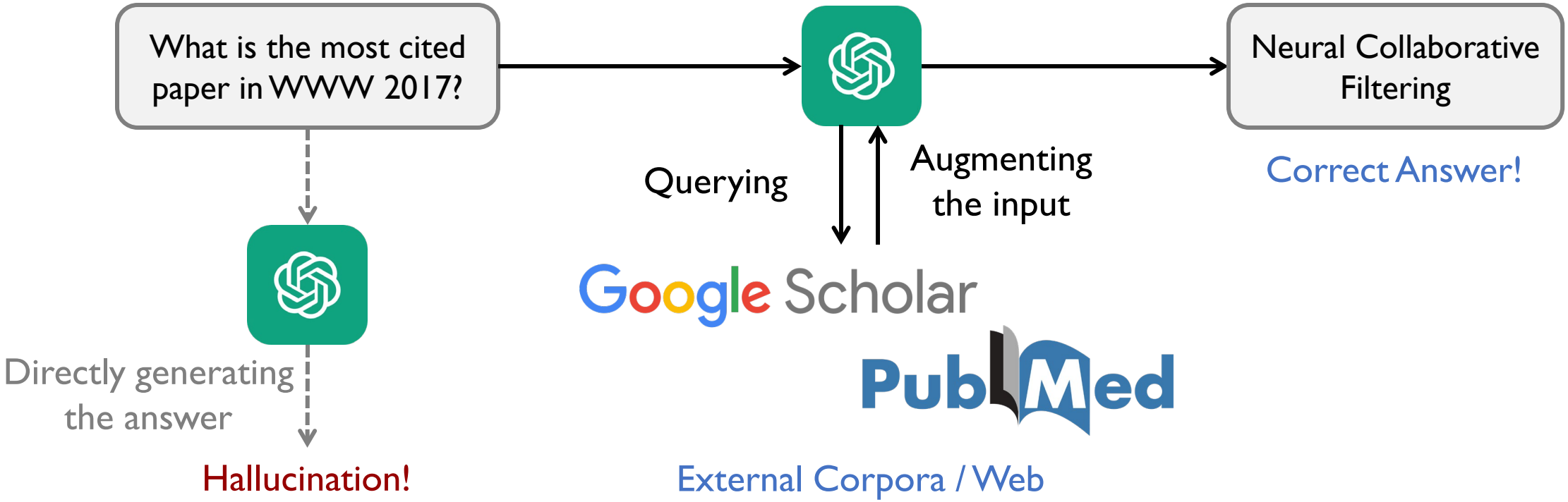
ChatGPT 3.5, queried on January 23, 2024

What if an LLM does not have the knowledge to answer a question?

- Several reasons why this happens
  - Knowledge cutoffs: parameters are usually only updated to a particular time
  - Private data: data stored in private text or data repositories not suitable for training
  - Learning failures: even for data that the model was trained on, it might not be sufficient to get the right answer

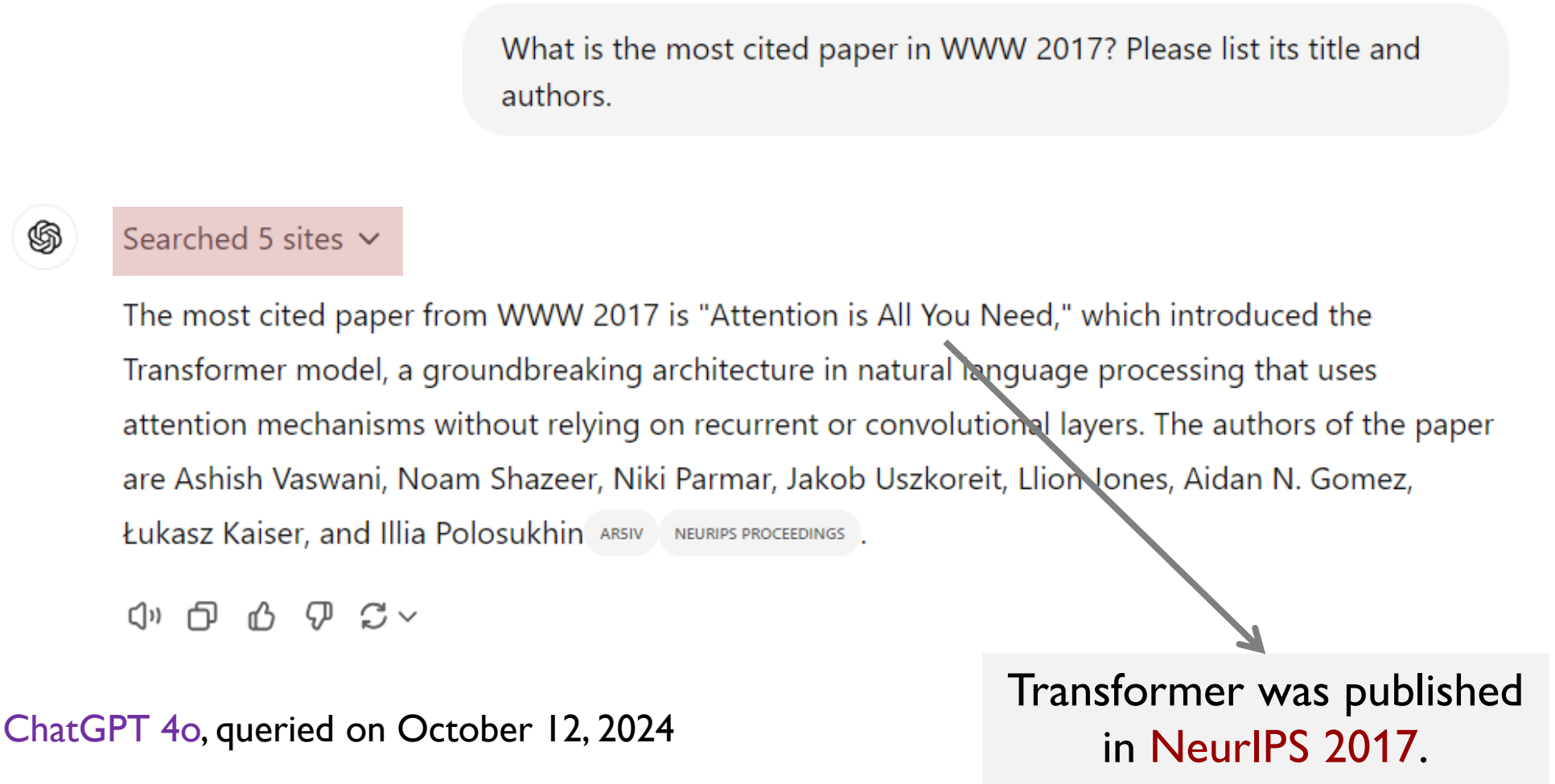
Retrieval-Augmented Generation

- *"Give a man a fish, and you feed him for a day; teach a man to fish, and you feed him for a lifetime."*
- *"Teach an LLM to know one more fact, and you help it answer one question; teach an LLM to explore the external corpora/Web, and you help it answer many questions."*



Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. NeurIPS 2020.

What if retrieval is not enough?



ChatGPT 4o, queried on October 12, 2024

What if retrieval is not enough?

- Retrieval is just one "tool" that can be used by LLMs.
- There are many other "tools".

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.

- Tool use: switching between the text-generation mode and the tool-execution mode

Toolformer: Language Models Can Teach Themselves to Use Tools. NeurIPS 2023.

How to induce tool use?

- Training
- Inference-time prompting

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.

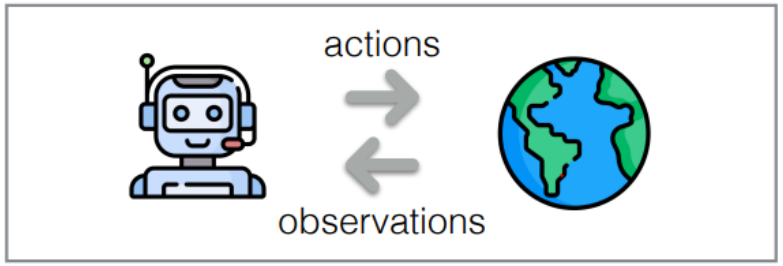
Input: Coca-Cola, or Coke, is a carbonated soft drink manufactured by the Coca-Cola Company.
Output: Coca-Cola, or **[QA("What other name is Coca-Cola known by?")]** Coke, is a carbonated soft drink manufactured by **[QA("Who manufactures Coca-Cola?")]** the Coca-Cola Company.

Input: x
Output:

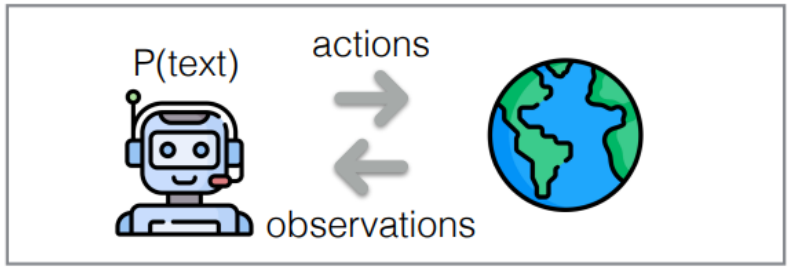
Toolformer: Language Models Can Teach Themselves to Use Tools. NeurIPS 2023.

From LLMs to Agents

- LLMs predict text. P(text)
- AI agents iteratively perform actions in the world.



- LLM agents are agents with an LLM backbone.

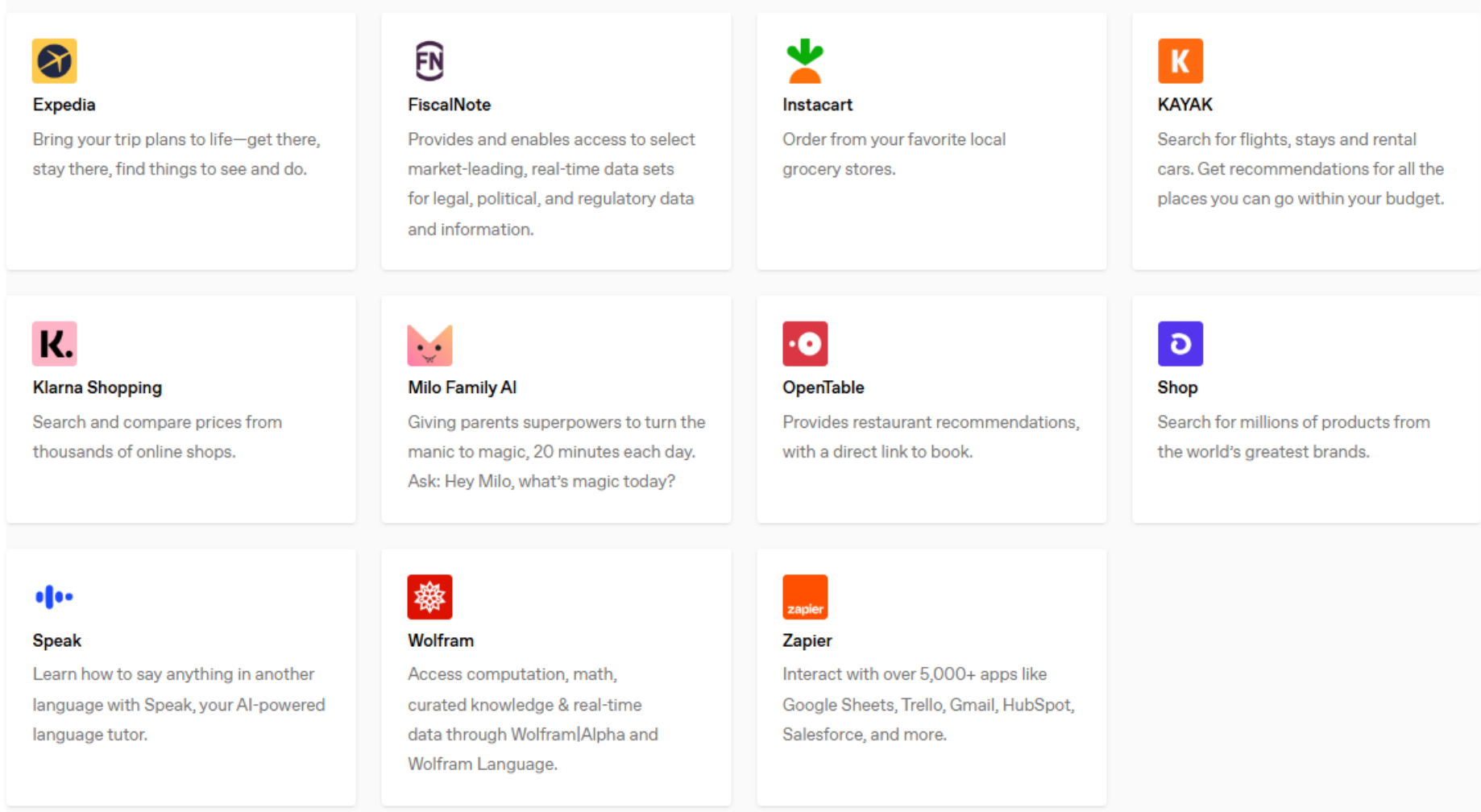


  - *Probably*: Proactive use of tools
  - *Probably*: An iterative, multi-step process
  - *Maybe*: Interaction with the outside world

Slide credit: Prof. Graham Neubig (CMU) https://www.phontron.com/class/anlp-fall2024/

From LLMs to Agents

- Example of Tool Use: ChatGPT plugins



Reasoning within Agents

- The correct answer (finally…):

DeepSeek-R1 671B, queried on May 8, 2025

Scratchpad of DeepSeek-R1

Reasoning
Started thinking...

Okay, so I need to find out the most cited paper from the WWW 2017 conference. Let me start by recalling that WWW stands for the International World Wide Web Conference, which is a major venue for research related to the web. The 2017 edition would have been the 26th conference.

First, I should figure out how to determine the most cited paper from that specific year. Citation counts can be tracked through databases like Google Scholar, Semantic Scholar, or the ACM Digital Library since ACM often sponsors these conferences. However, citation counts can vary over time, so the most cited paper might change as newer papers accumulate citations. But since the user is asking about the current most cited, I need up-to-date information.

I'll start by checking the proceedings of WWW 2017. The conference usually publishes its proceedings, and sometimes the awards or notable papers are mentioned. If I can find the list of accepted papers, I can then look up each of them on Google Scholar to check their citation counts. But that's time-consuming. Alternatively, maybe there's a known paper from that year that's highly cited. For example, in recent years, transformer-based models like BERT or attention mechanisms have been highly cited, but 2017 might be a bit early for those.

Wait, 2017 is the year when the "Attention Is All You Need" paper was published, but that was at NIPS, not WWW. So maybe that's not relevant here. Let me think. In the realm of web conferences, topics like graph neural networks, recommendation systems, network analysis, and so on are common.

Another approach is to search for "most cited papers from WWW 2017" on Google or academic search engines. Let me try that. Searching for that exact phrase, I find some resources. For instance, there's a page on the WWW 2017 conference website that lists accepted papers, but I need to check if they have any awards or most cited mentions. Alternatively, Semantic Scholar might have a list sorted by citations.

Let me check Semantic Scholar. Going to their website, I can search for papers from the WWW 2017 conference. Applying the filters for the year 2017 and the venue WWW, then sorting by citations. Let's see. The top result there might be the one. Alternatively, Google Scholar can be used by searching for the conference name and year, then sorting by citations.

Upon doing a quick search, I find that the paper "Inductive Representation Learning on Large Graphs" by William L. Hamilton, Rex Ying, and Jure Leskovec is highly cited. Wait, but that was published at NIPS 2017, right? Wait, no, maybe I'm confused. Let me check the details. The Hamilton et al. paper on GraphSAGE was indeed presented at NIPS 2017. So maybe that's not the one.

Alternatively, another paper from WWW 2017 that's highly cited could be related to graph convolutional networks or something similar. Let me check the proceedings. For example, "Graph Convolutional Matrix Completion" by Rianne van den Berg, Thomas N. Kipf, and Max Welling was published in 2017, but I think that's at a different conference.

Another possibility is "Neural Collaborative Filtering" by Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. Wait, but that paper was published in WWW 2017? Let me confirm. According to Google Scholar, that paper was indeed published in the Proceedings of the 26th International Conference on World Wide Web in 2017. It's possible that this paper is highly cited as it introduces neural networks into collaborative filtering, a popular approach in recommendations.

Checking the citation counts: As of now, "Neural Collaborative Filtering" has over 5,000 citations on Google Scholar, which is very high. Another paper, "DeepFM: A Factorization-Machine based Neural Network for CTR Prediction" has over 2,000 citations. There might be others. Wait, but I need to confirm which one has the highest.

Alternatively, "Inductive Representation Learning on Large Graphs" (GraphSAGE) by Hamilton et al. has over 6,000 citations, but that's from NIPS 2017, not WWW. So perhaps the most cited from WWW 2017 is "Neural Collaborative Filtering". Let me cross-verify.

Looking at the WWW 2017 accepted papers list, "Neural Collaborative Filtering" is indeed there. The authors are Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. That paper is foundational in combining neural networks with collaborative filtering, which explains its high citation count.

Alternatively, another paper from WWW 2017 that's highly cited is "StarSpace: Embed All The Things!" by Ledell Wu, Adam Fisch, Sumit Chopra, Keith Adams, Antoine Bordes, and Jason Weston. This paper also has a substantial number of citations, but I think it's less than "Neural Collaborative Filtering".

To be sure, checking Google Scholar for "Neural Collaborative Filtering" shows over 5,000 citations as of 2023. "DeepFM" has around 2,000. Another paper, "Metapath2vec: Scalable Representation Learning for Heterogeneous Networks" by Yuxiao Dong, Nitesh V. Chawla, and Ananthram Swami, which was also presented at WWW 2017, has over 1,500 citations.

Therefore, it's likely that "Neural Collaborative Filtering" by He et al. is the most cited paper from WWW 2017. However, I should also check if there's any other paper with higher citations. For example, "Attention Is All You Need" is from 2017 but not WWW. "Dynamic Routing Between Capsules" by Hinton et al. is also from 2017 but a different conference.

Another possible candidate is "Semi-Supervised Classification with Graph Convolutional Networks" by Kipf and Welling, but that was published at ICLR 2017, so not WWW.

Therefore, after checking the citation counts and cross-referencing with the conference proceedings, "Neural Collaborative Filtering" by Xiangnan He and colleagues seems to be the most cited paper from WWW 2017. However, I should verify once more. Let me check another source.

On Semantic Scholar, searching for WWW 2017 papers sorted by citations: the top result is "Neural Collaborative Filtering" with 5,442 citations as of October 2023. The next is "Metapath2vec" with 2,035 citations. Then "DeepFM" with 1,821. So yes, "Neural Collaborative Filtering" is the most cited.

Thus, the answer should be that paper.

Scratchpad of DeepSeek-R1

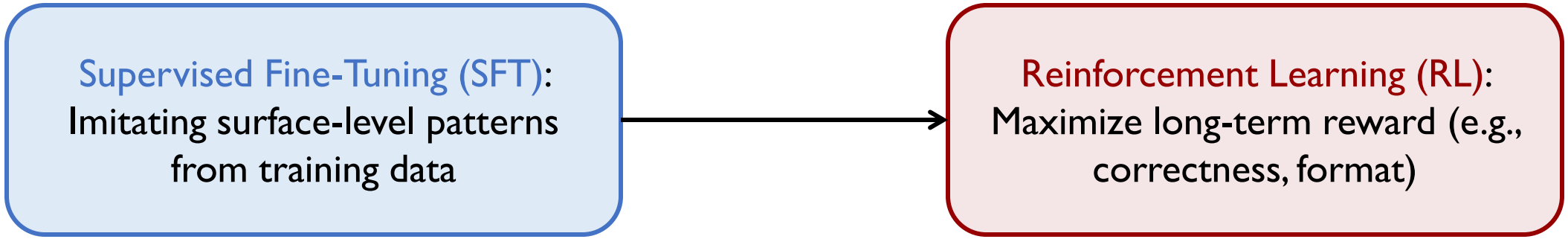
Another approach is to search for "most cited papers from WWW 2017" on Google or academic search engines. Let me try that. Searching for that exact phrase, I find some resources. For instance, there's a page on the WWW 2017 conference website that lists accepted papers, but I need to check if they have any awards or most cited mentions. Alternatively, Semantic Scholar might have a list sorted by citations.

Let me check Semantic Scholar. Going to their website, I can search for papers from the WWW 2017 conference. Applying the filters for the year 2017 and the venue WWW, then sorting by citations. Let's see. The top result there might be the one. Alternatively, Google Scholar can be used by searching for the conference name and year, then sorting by citations.

Checking the citation counts: As of now, "Neural Collaborative Filtering" has over 5,000 citations on Google Scholar, which is very high. Another paper, "DeepFM: A Factorization-Machine based Neural Network for CTR Prediction" has over 2,000 citations. There might be others. Wait, but I need to confirm which one has the highest.

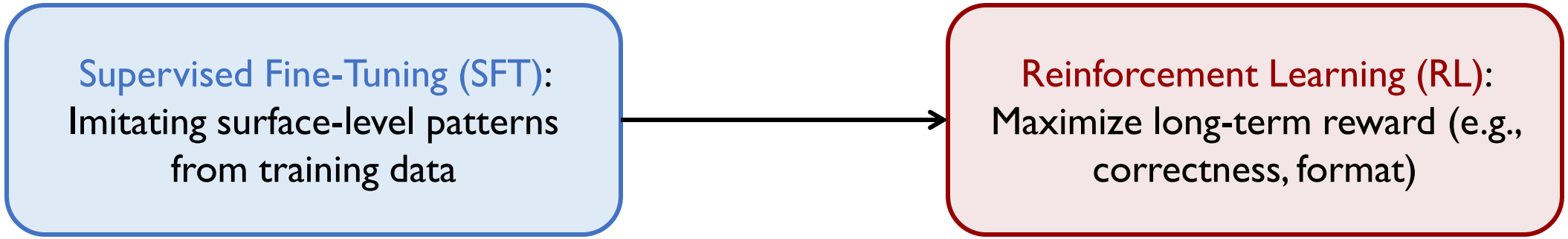
Alternatively, "Inductive Representation Learning on Large Graphs" (GraphSAGE) by Hamilton et al. has over 6,000 citations, but that's from NIPS 2017, not WWW. So perhaps the most cited from WWW 2017 is "Neural Collaborative Filtering". Let me cross-verify.

How to force LLMs to reason?



- Supervised Fine-Tuning (SFT): Given a question, maximize the likelihood that the LLM generates the reference answer.
  - If the reference answer is 0.5, then generating 1/2 is wrong.
  - If the LLM is asked to write code for a specific function, only an exact match with the reference answer is considered fully correct. A program that differs from the reference by just one token (even if it does not run at all) is regarded as better than an alternative implementation that correctly achieves the same functionality.

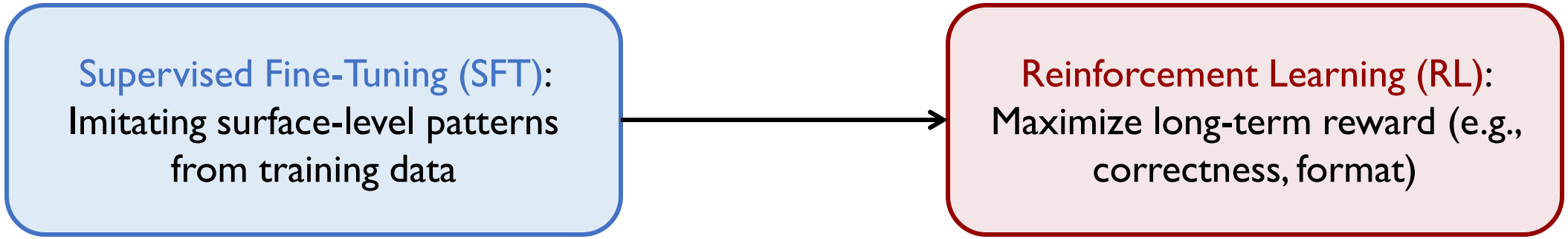
How to force LLMs to reason?



- Reinforcement Learning (RL): When the LLM does something correct according to our rubrics, it gets certain reward. The model is trained to maximize the reward.
  - Format Reward:
    - Has <think> and </think> tokens in its output; puts its intermediate reasoning steps between these two tokens.
    - Has <answer> and </answer> tokens in its output; puts its final answer between these tokens
    - …

DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning. arXiv 2025.

How to force LLMs to reason?

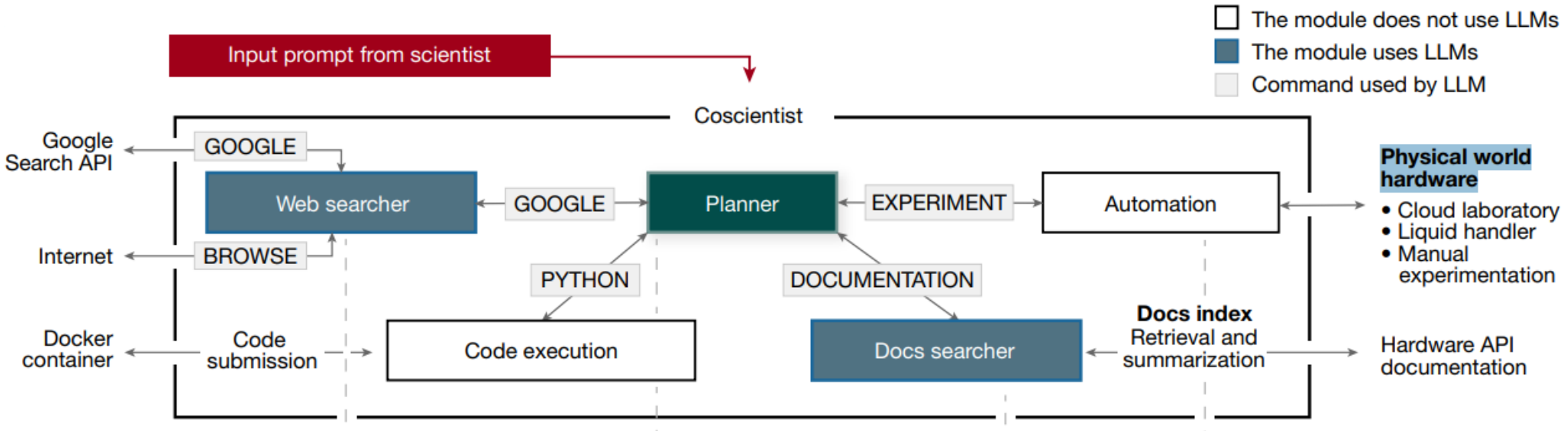


- Reinforcement Learning (RL): When the LLM does something correct according to our rubrics, it gets certain reward. The model is trained to maximize the reward.
  - Correctness Reward:
    - The answer (i.e., tokens between <answer> and </answer>) should be "equivalent" to the reference answer (according to a verifier).
  - If the model does not generate the correct answer, it will NOT know the answer after this iteration; it will only know that the reward is low.
    - This question can be used repeatedly until the model generates a good answer.

DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning. arXiv 2025.

Summary: LLM-based Agents for Science

- Lots of heterogeneous resources associated with scientific research
  - Web, data analysis software, physical world hardware, …
  - Natural language should serve as the "glue" that binds them together.
  - Equipping LLMs with corresponding tools



- Can we use the idea of RL to further enhance their scientific reasoning and planning abilities?

Autonomous Chemical Research with Large Language Models. Nature 2023.

Thank You!