Swiss Institute of Bioinformatics

2025 REVIEWS IN QUANTITATIVE BIOLOGY

Introduction to AI Agents and Agentic AI

– foundations for a literature review

Dr Alan Bridge, Co-Director, Swiss-Prot group, SIB

November 21st, 2025

elixir SWITZERLAND

Why review agentic AI?

High stakes, High impact

Agentic AI could reshape science from hypothesis generation to lab automation and paper writing.

The Hype Is Real

Explosion of papers, blogs, and claims that AI agents are "revolutionizing science".

But what's the evidence?

Are these claims scientifically justified or driven by hype? What is the state of the art?

Our Role as Scientists

A duty to critically evaluate performance and understand their ethical & societal impact.

We also have a vested interest in this

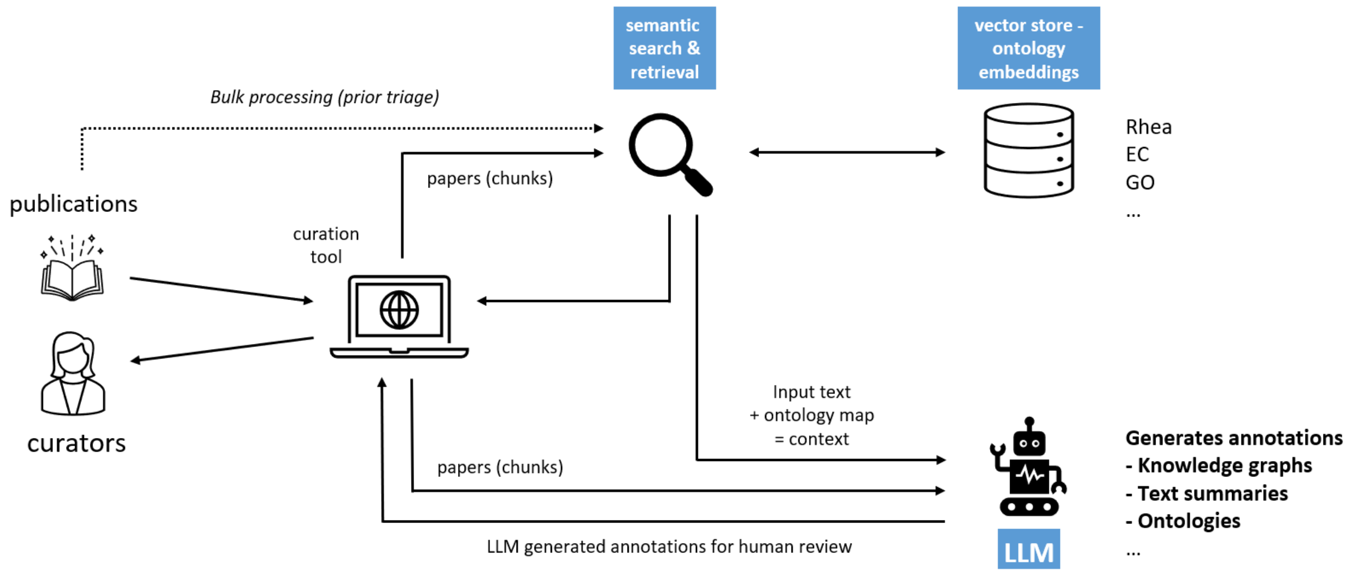
We use AI to extract knowledge from papers for the UniProt knowledgebase.

This is a human in the loop AI assisted curation workflow

- experts have editorial control
- AI performs a limited range of tasks with low autonomy

Q: Should we grant greater agency to our AI systems?

I'd like you to help me decide.



A schematic of our knowledge extraction pipeline at Swiss-Prot.

Quotes from recent papers

Agentic bioinformatics has emerged as a groundbreaking paradigm

Generative AI agents are transforming biology

Agents Achieve Superhuman Synthesis of Scientific Knowledge

What lies behind these claims?
What can agents do, and what have agents achieved?

Themes to explore in this literature review

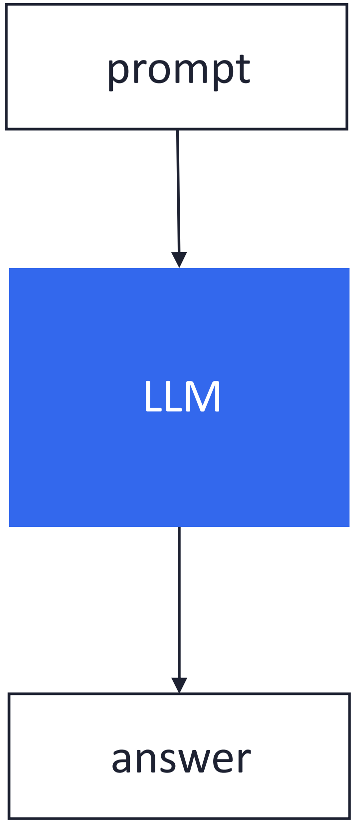
- Definitions of AI agents and Agentic AI, and relation to LLMs
- Agentic AI in biology & bioinformatics
- Evaluation and benchmarks
- Limitations of current agents
- Ethical considerations

These slides and a starting bibliography are attached to

https://lab.dessimoz.org/teaching/rqb/schedule_2025

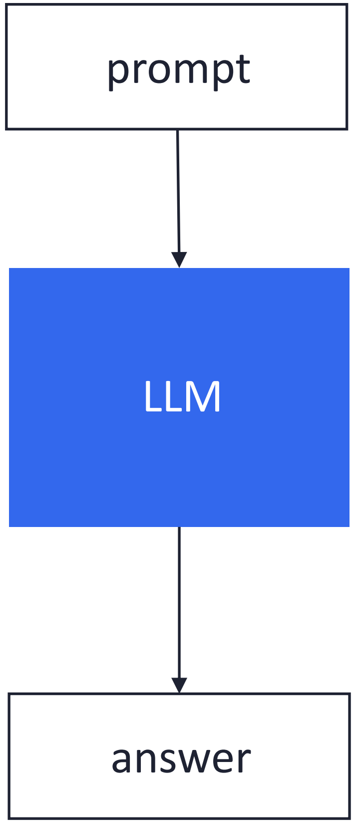
Definitions

LLM

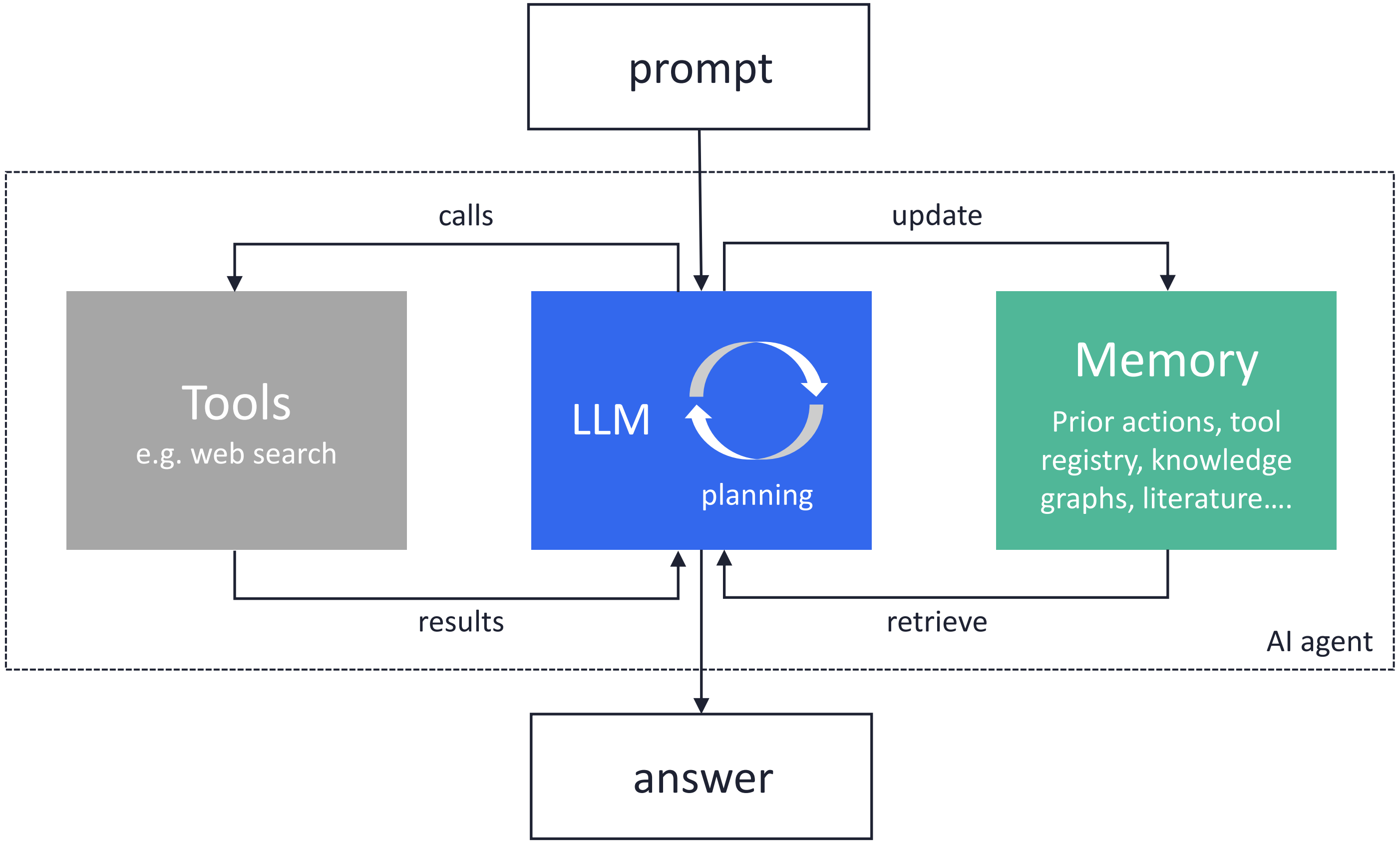


LLMs answer reflexively

LLM -> AI Agent

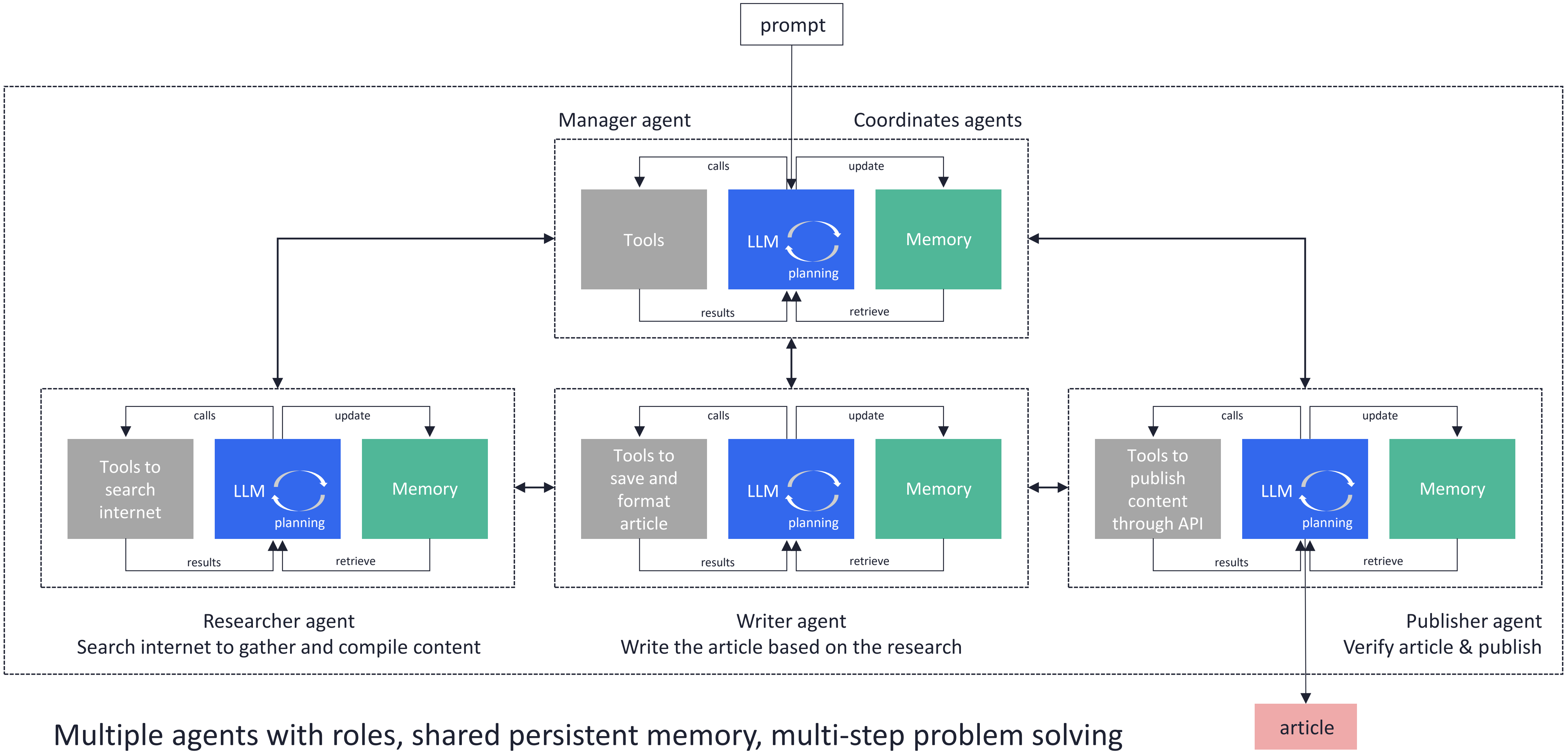


LLMs answer
reflexively



Has capacity to decide what action to take.
Access to tools, memory and planning.

LLM -> AI Agent -> Agentic AI (sample architecture, lit review agent)



Multiple agents with roles, shared persistent memory, multi-step problem solving

Delving into agents

LLM basics → AI Agent basics: tools and reasoning → AI Agents in biology → AI Agent evaluation → AI Agent Safety & ethics

Delving into agents

LLM basics

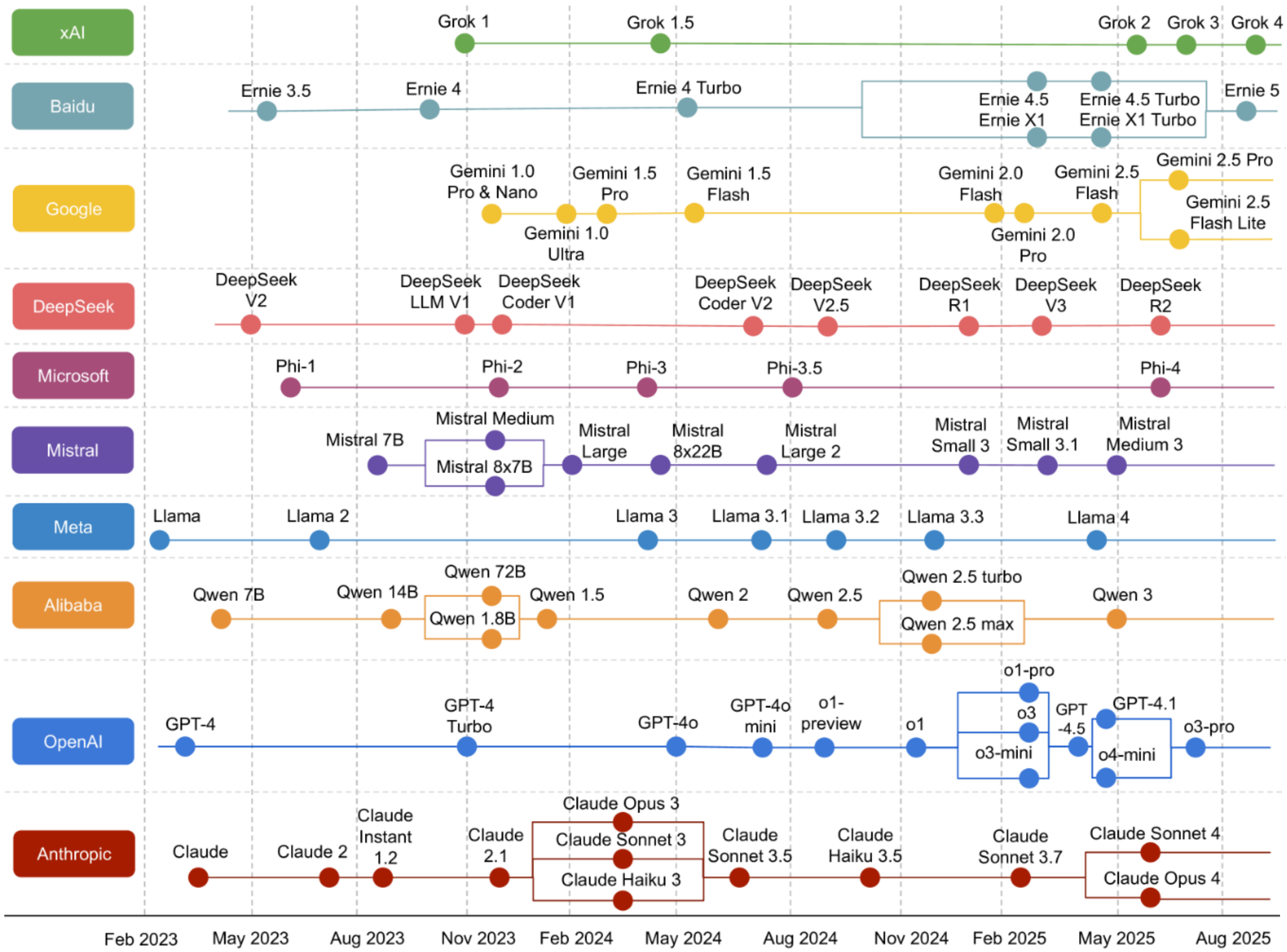
AI Agent basics: tools and reasoning

AI Agents in biology

AI Agent evaluation

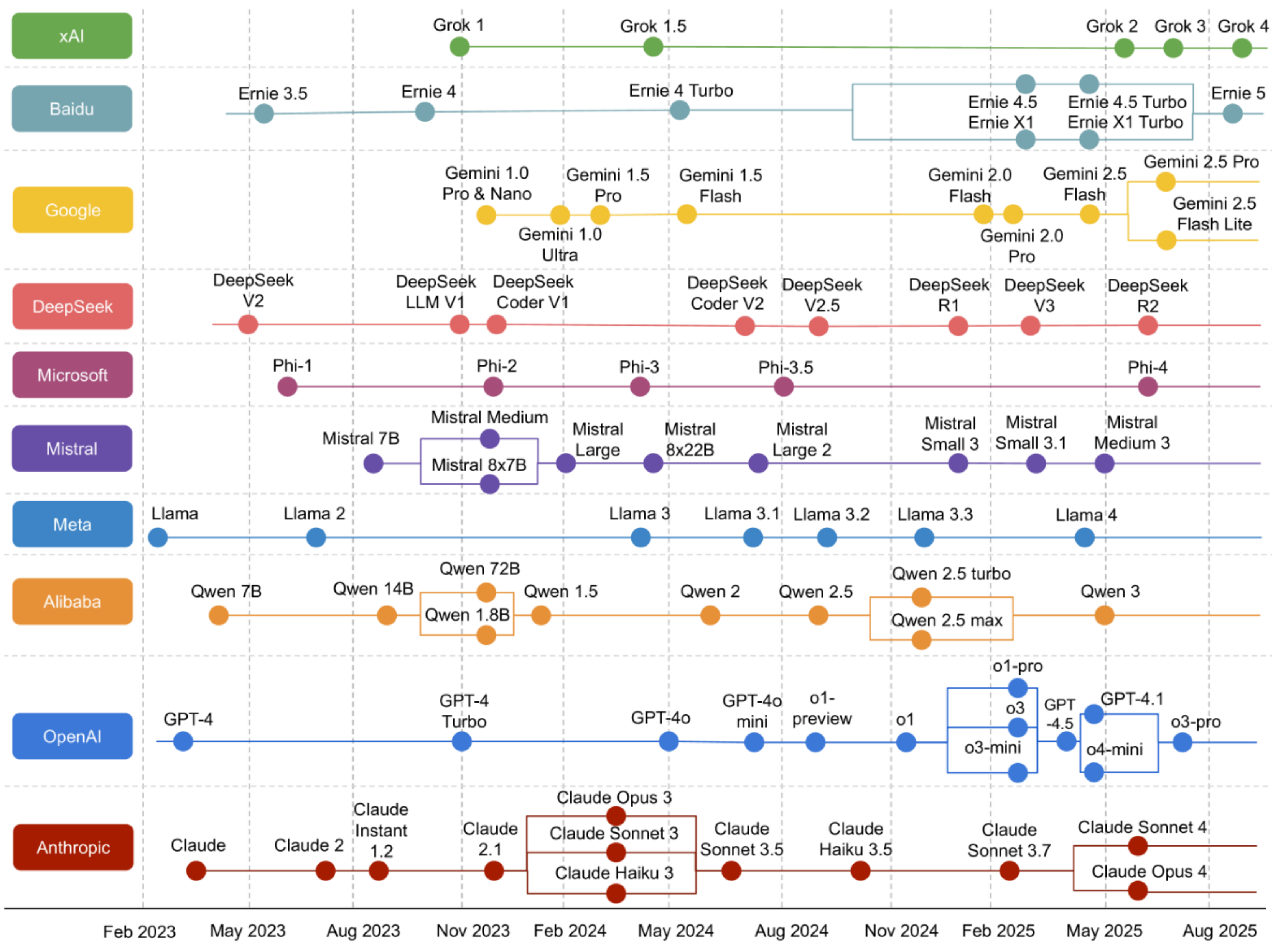
AI Agent Safety & ethics

Some of the main LLMs to emerge since 2023



Le and Abel, 2025
[arXiv:2507.18479](https://arxiv.org/abs/2507.18479)

Some of the main LLMs to emerge since 2023



Agents use LLMs for reasoning and planning – access is an issue

Le and Abel, 2025
arXiv:2507.18479

GPTs are next token predictors

During training, the GPT model is trained to predict the next token in a series of tokens.

GPTs are next token predictors

The cat played in the

This is our input context

GPTs are next token predictors

…
spaceship
concert
bedroom
The cat played in the garden
shed
bar
washer
…

Given an input context the model produces a probability distribution over the vocabulary for the next token.

GPTs are next token predictors

…
spaceship
concert
bedroom
The cat played in the garden
shed
bar
washer
…

Ideally the model will assign the highest probability to the actual next word in the training data.

GPTs are next token predictors

…
spaceship
concert
bedroom
The cat played in the garden
shed
bar
washer
…

Ideally the model will assign the highest probability to the actual next word in the training data.

GPTs are next token predictors

The cat played in the garden

Ideally the model will assign the highest probability to the actual next word in the training data.

GPTs are next token predictors

The cat played in the garden

GPT models are autoregressive, so with the token appended to the context the process continues

GPTs are next token predictors

...
except
only
if
The cat played in the garden every
when
each
unless
...

Given this new input context the model produces a new probability distribution for the next token.

GPTs are next token predictors

…
except
only
if
The cat played in the garden every
when
each
unless
…

Ideally the model will again assign the highest probability to the actual next word in the training data.

GPTs are next token predictors

The cat played in the garden every

Creating a new context

GPTs are next token predictors

The cat played in the garden every

Again, with this new input context the model produces a new probability distribution for the next token.

GPTs are next token predictors

…
Sunday
evening
night
The cat played in the garden every day
morning
noon
Christmas
…

Again, with this new input context the model produces a new probability distribution for the next token.

GPTs are next token predictors

...
Sunday
evening
night
The cat played in the garden every day
morning
noon
Christmas
...

Ideally the model will again assign the highest probability to the actual next word in the training data.

GPTs are next token predictors

The cat played in the garden every day

The sequence is complete

GPTs are next token predictors

GPT models can ‘**hallucinate**’, it’s a feature not a bug.

GPTs are next token predictors

The cat played in the

GPT models can ‘**hallucinate**’, it’s a feature not a bug.

GPTs are next token predictors

…
spaceship
concert
bedroom
The cat played in the garden
shed
bar
washer
…

The model may select words that differ from those seen in the actual training data.

GPTs are next token predictors

…
spaceship
concert
bedroom
The cat played in the garden
shed
bar
washer
…

The model may select words that differ from those seen in the actual training data.

GPTs are next token predictors

The cat played in the concert

The model may select words that differ from those seen in the actual training data.

GPTs are next token predictors

…
as
only
if
The cat played in the concert on
with
each
every
…

Again, with this new input context the model produces a new probability distribution for the next token.

GPTs are next token predictors

The cat played in the concert on

A new token is appended to the context

GPTs are next token predictors

...
Sunday
stage
bass
The cat played in the concert on drums
guitar
tv
acid
...

The process continues

GPTs are next token predictors

The cat played in the concert on drums

Ultimately creating a low probability outcome

GPTs are next token predictors

The cat played in the concert on drums

GPTs are next token predictors

… an example of “hallucination” but also the LLM doing what it was trained to do

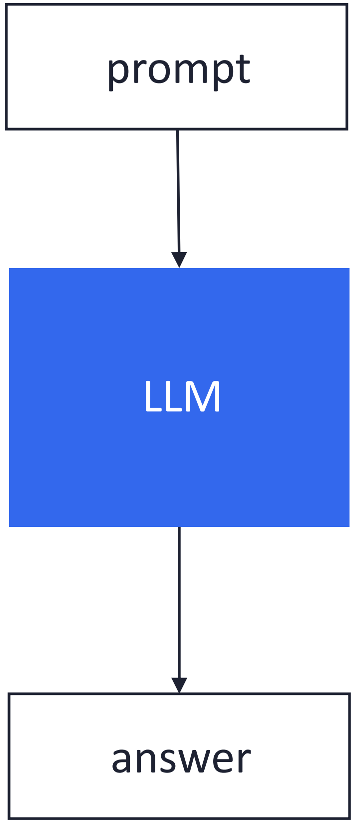
GPTs are next token predictors

AI Agents build on LLMs and inherit these characteristics

Delving into agents

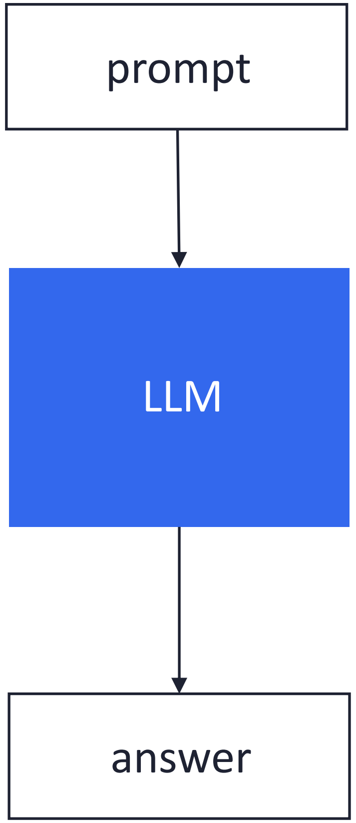
- LLM basics
- **AI Agent basics: tools and reasoning**
- AI Agents in biology
- AI Agent evaluation
- AI Agent Safety & ethics

LLM

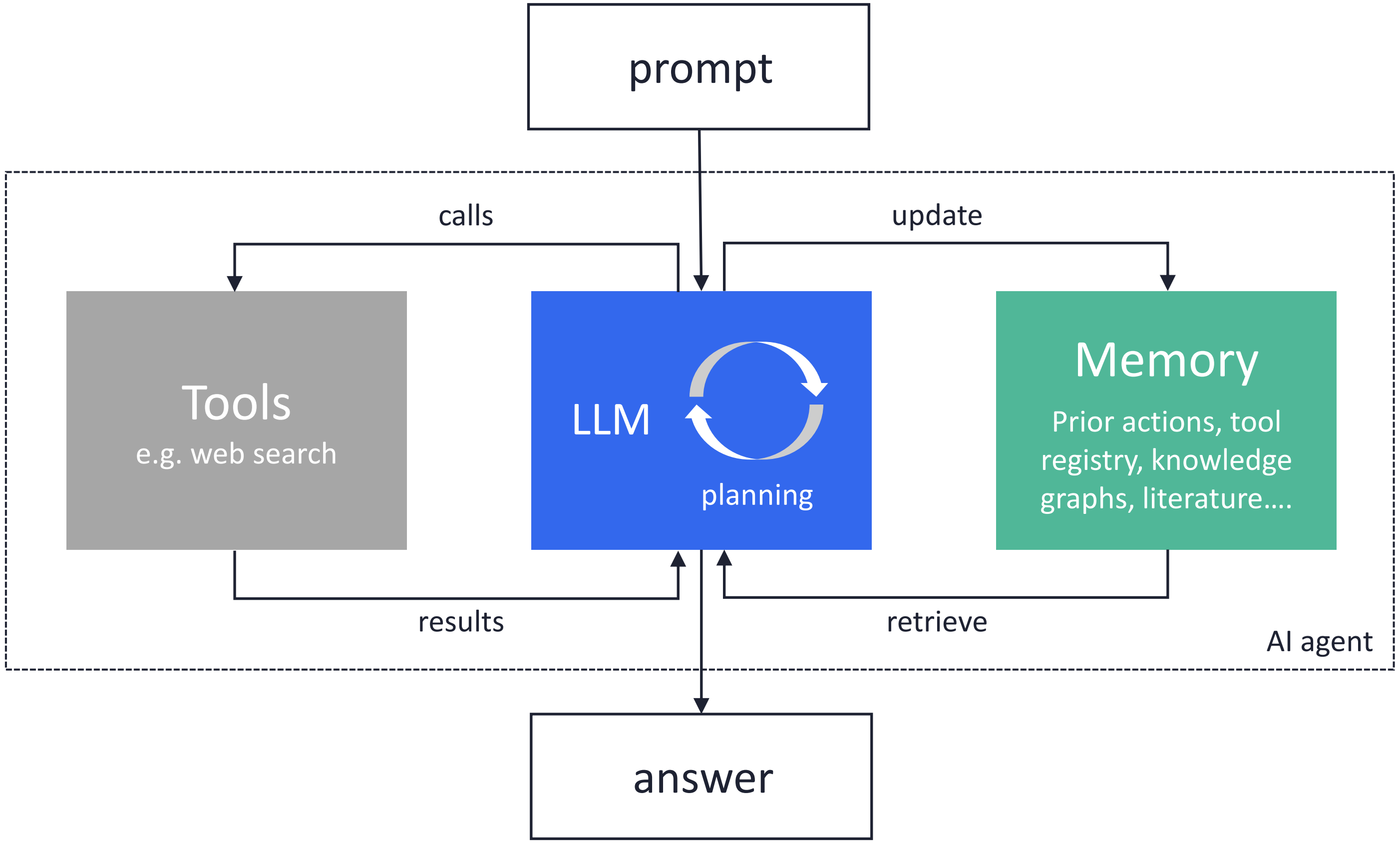


LLMs answer reflexively

LLM -> AI Agent



LLMs answer reflexively



Simple AI Agent adds tools, memory, and planning. Cycles of planning, tool use, and memory updates continue until answer found.

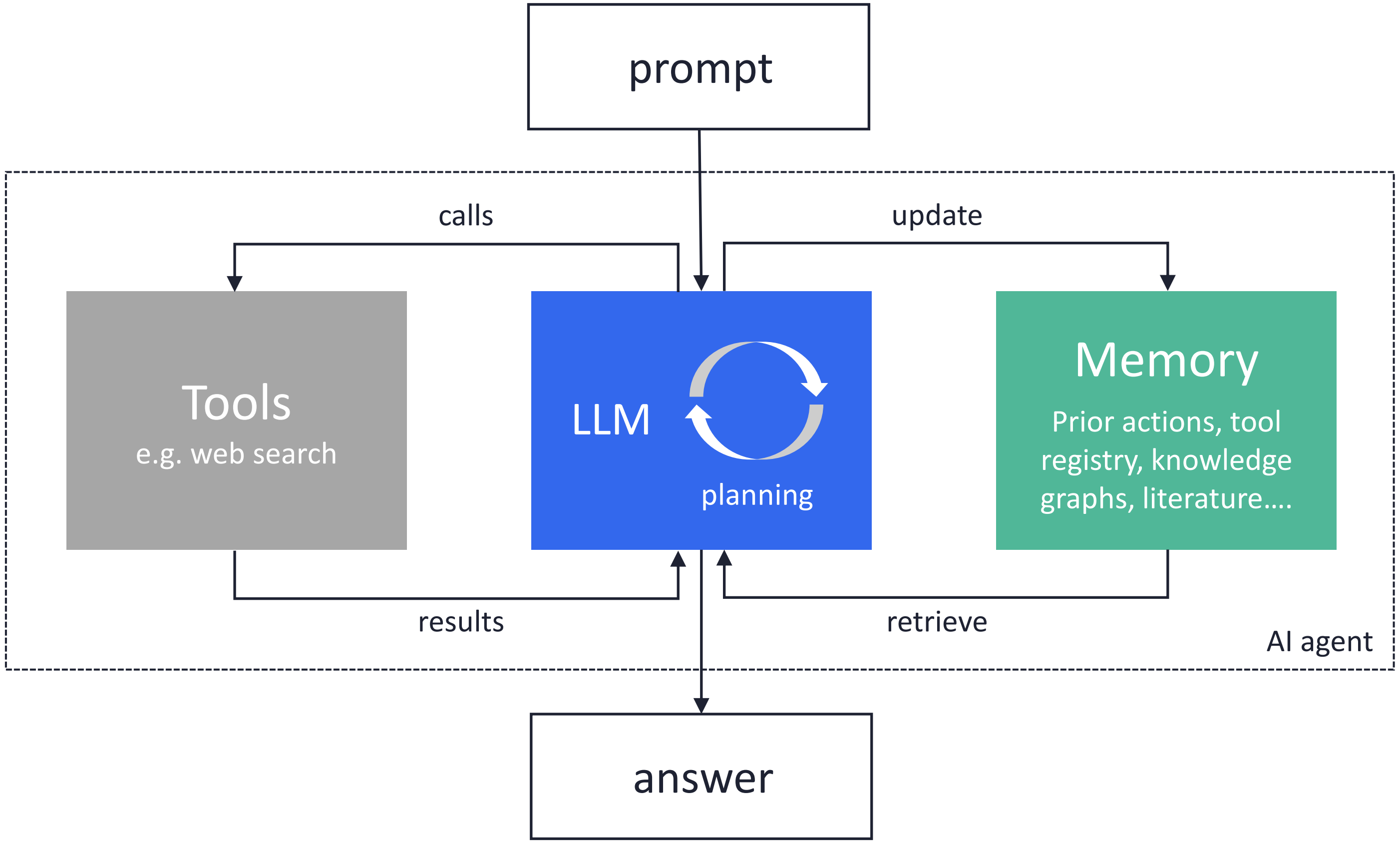
Tools

Tools allow agents to perform actions deterministically.

Examples:

- Code compiler
- Web search
- Calculator

So rather than do sums by next token prediction as an LLM, an agent would invoke a calculator.



Simple AI Agent adds tools, memory, and planning. Cycles of planning, tool use, and memory updates continue until answer found.

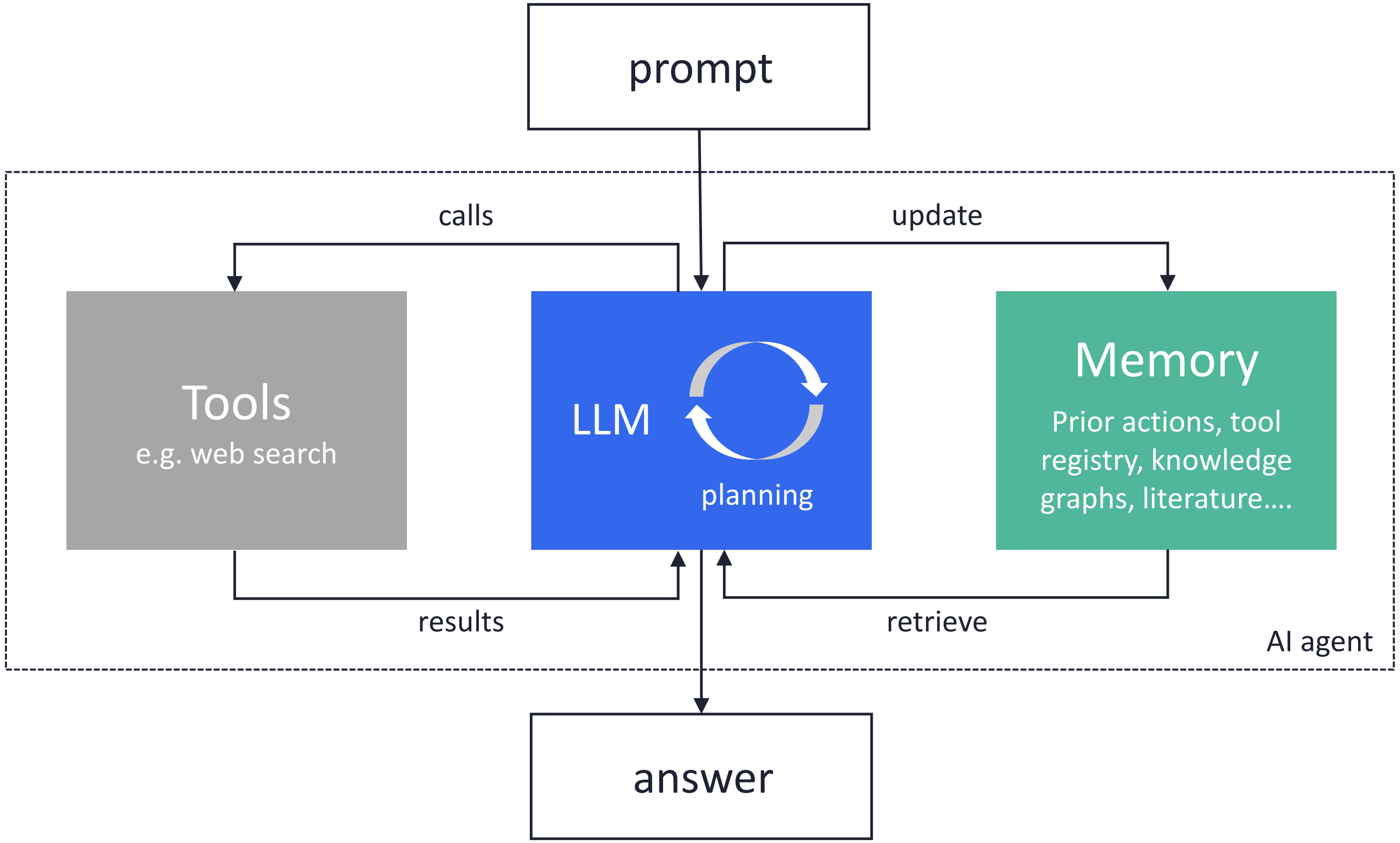
Tools, what tools?

To tell the LLM which tools are available to it the agent framework may

- inject the list of tools in context with the request (e.g. OpenAI API)

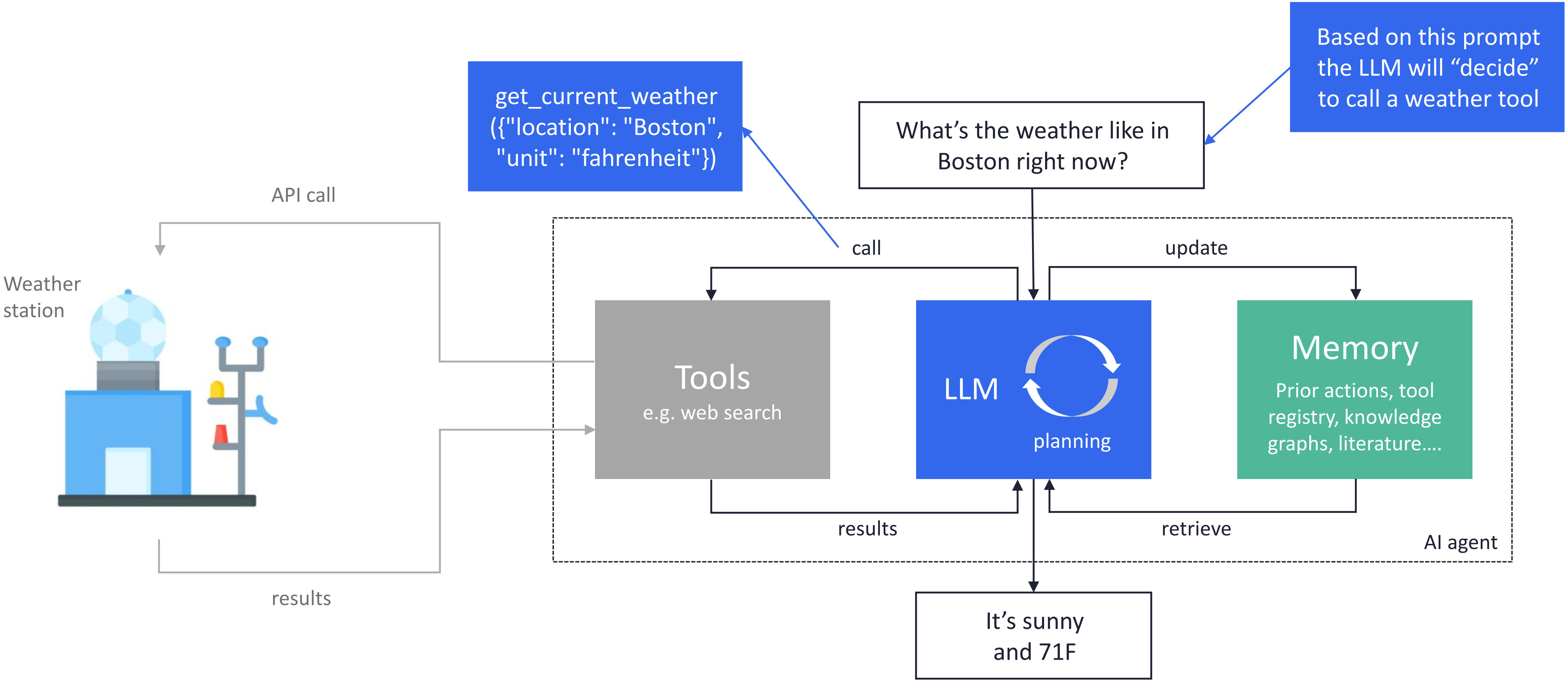
- include a tool registry the LLM can look up (e.g. LangChain)

Having chosen a tool, the LLM will output a function call (JSON)



Simple AI Agent adds tools, memory, and planning. Cycles of planning, tool use, and memory updates continue until answer found.

Tool use example

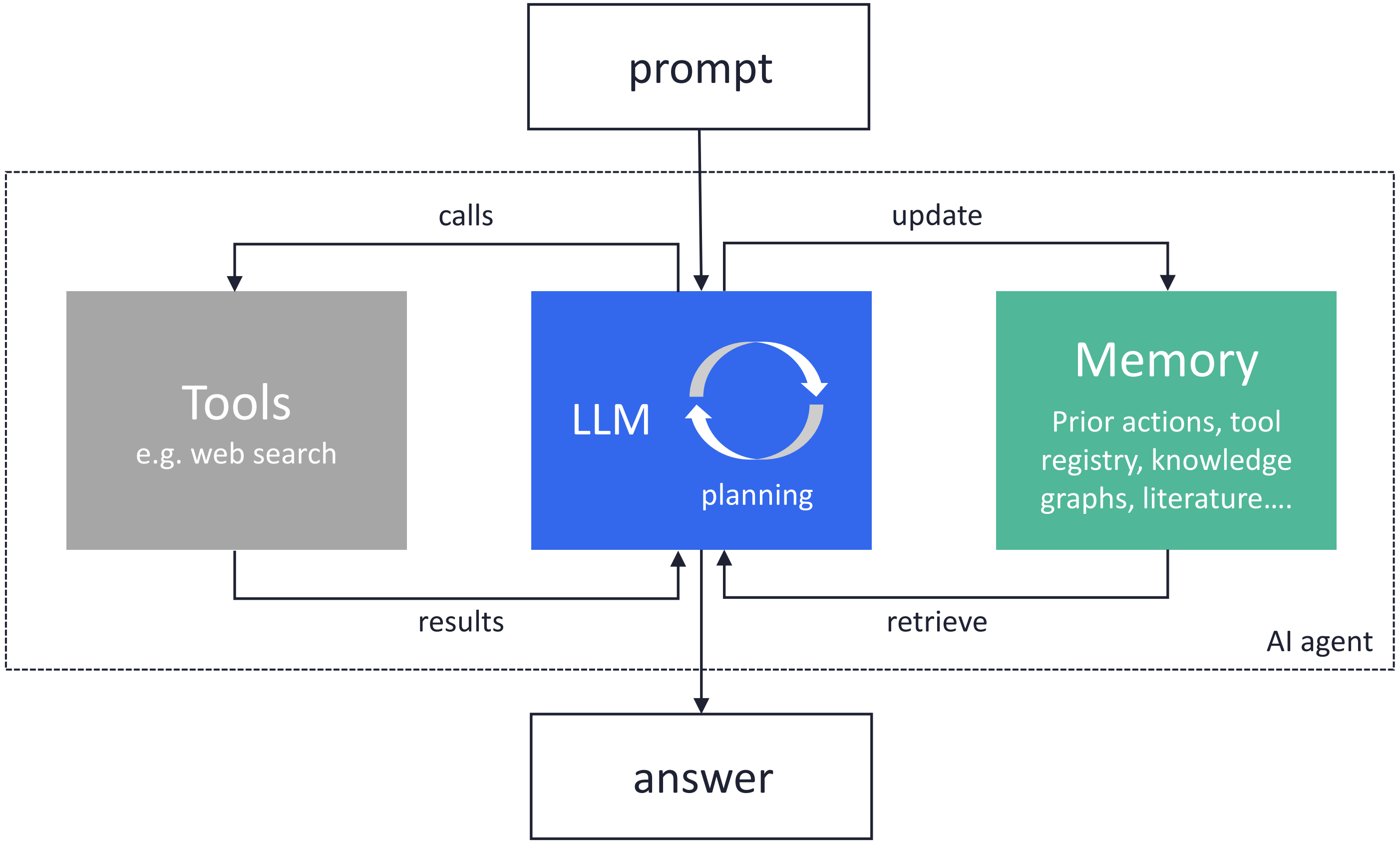


1. The LLM constructs the function call specifying the tool and parameters; 2. the tool code makes the API call and receives the results; 3. the LLM uses this to formulate the textual answer, namely "It's sunny and 71F"

Tools summary

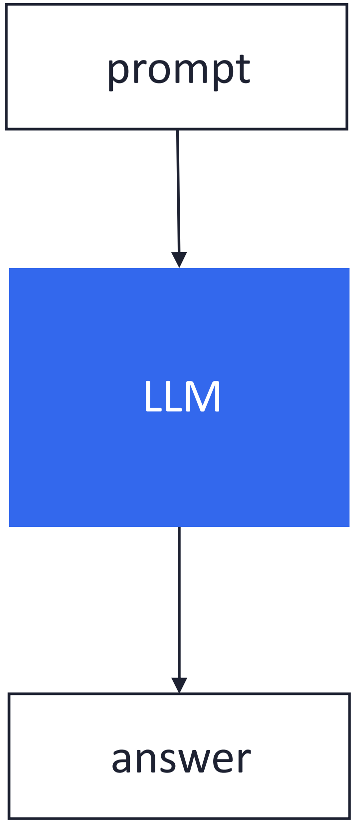
The LLM decides WHAT needs to be done

The agent application handles HOW to do it

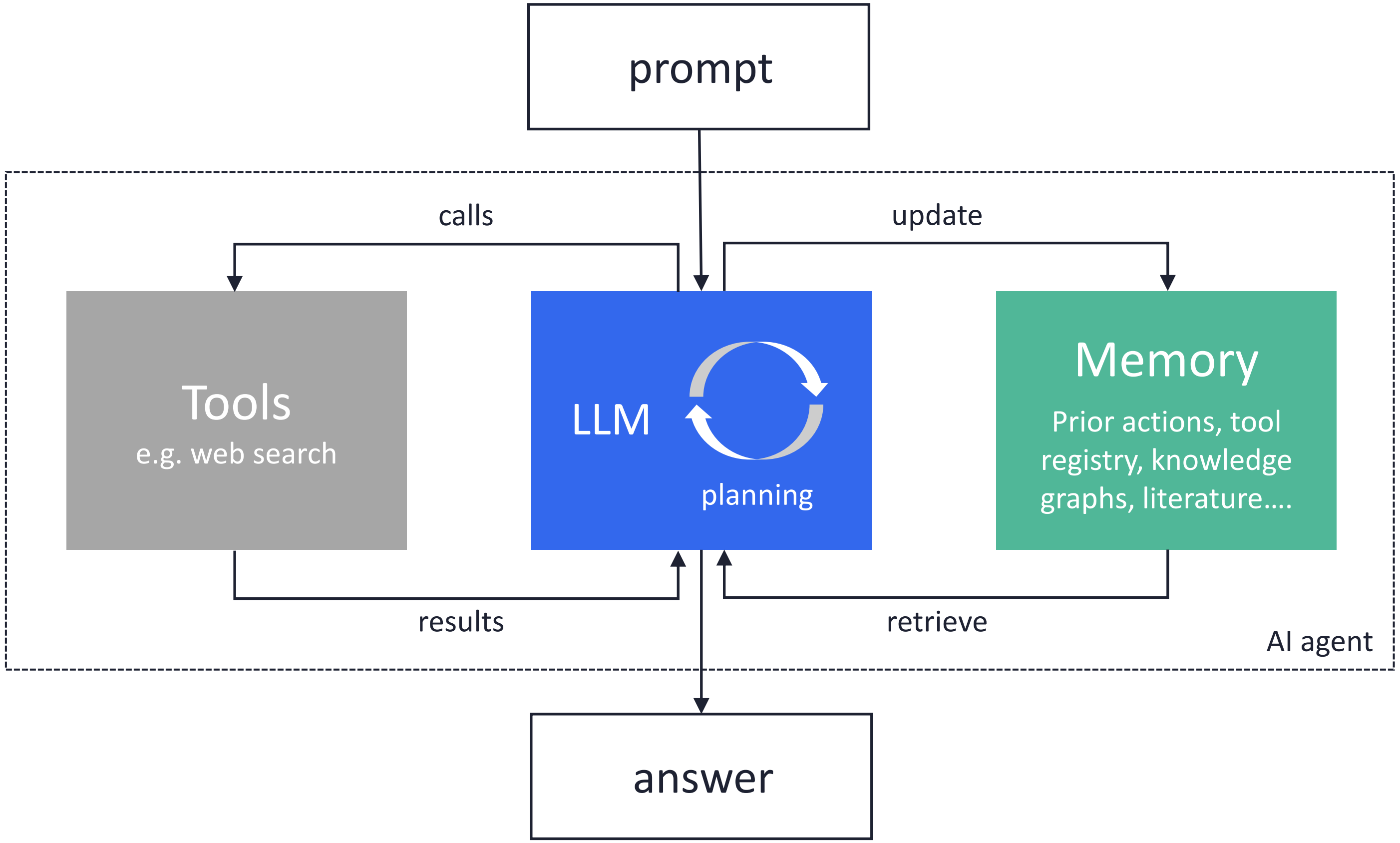


Simple AI Agent adds tools, memory, and planning. Cycles of planning, tool use, and memory updates continue until answer found.

Recap: LLM -> AI Agent

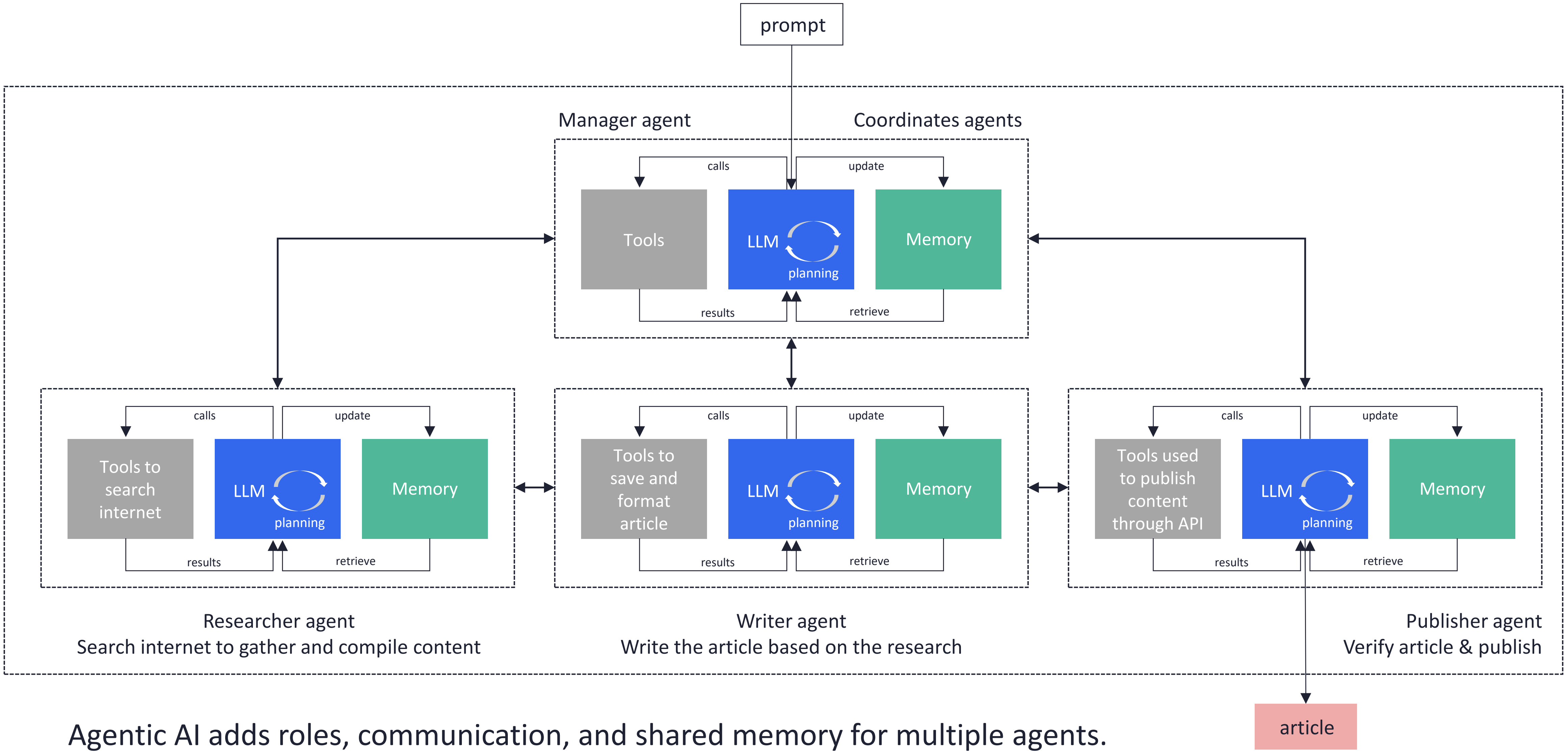


LLMs answer reflexively



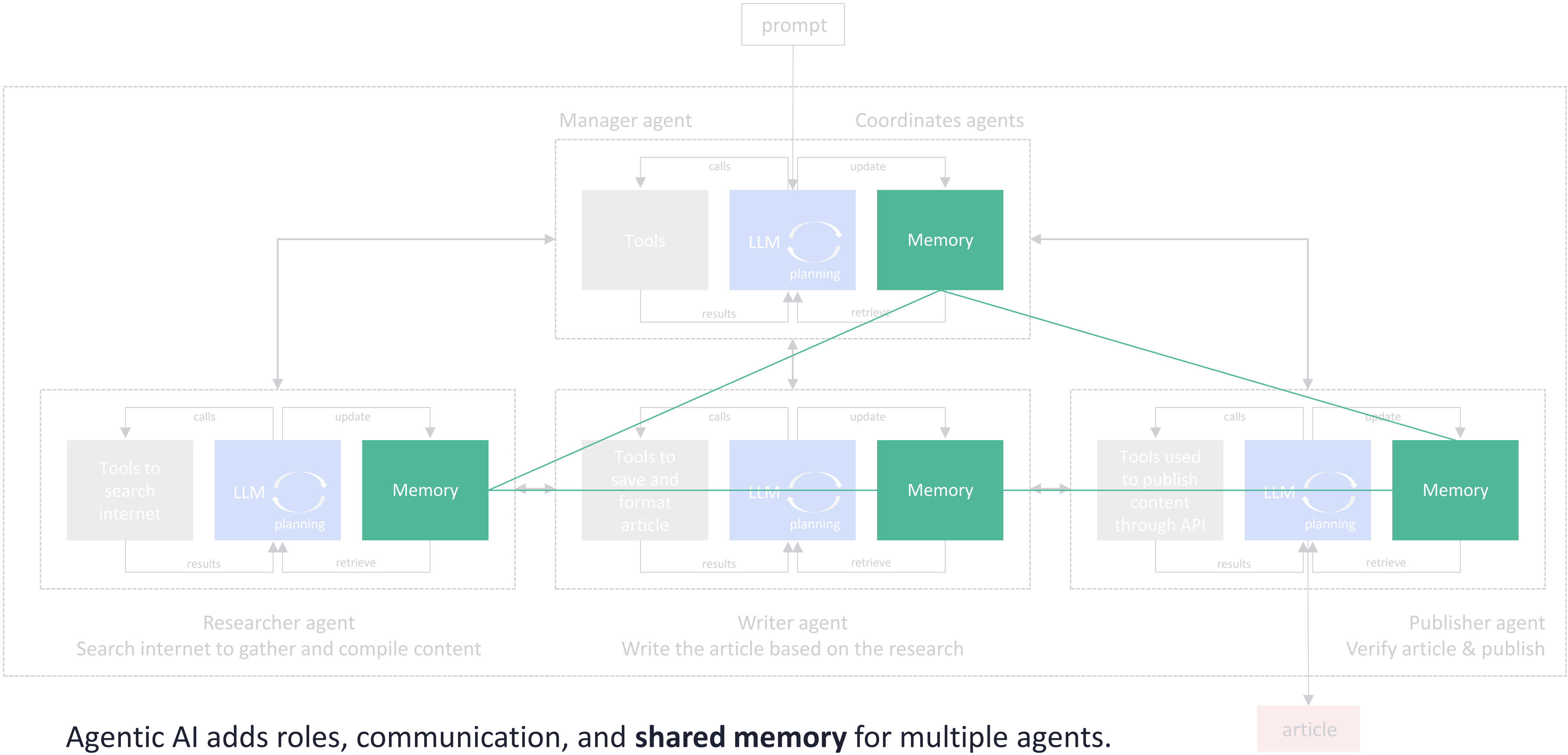
Simple AI Agent adds tools, memory, and planning. Cycles of planning, tool use, and memory updates continue until answer found.

LLM -> AI Agent -> Agentic AI (sample architecture)



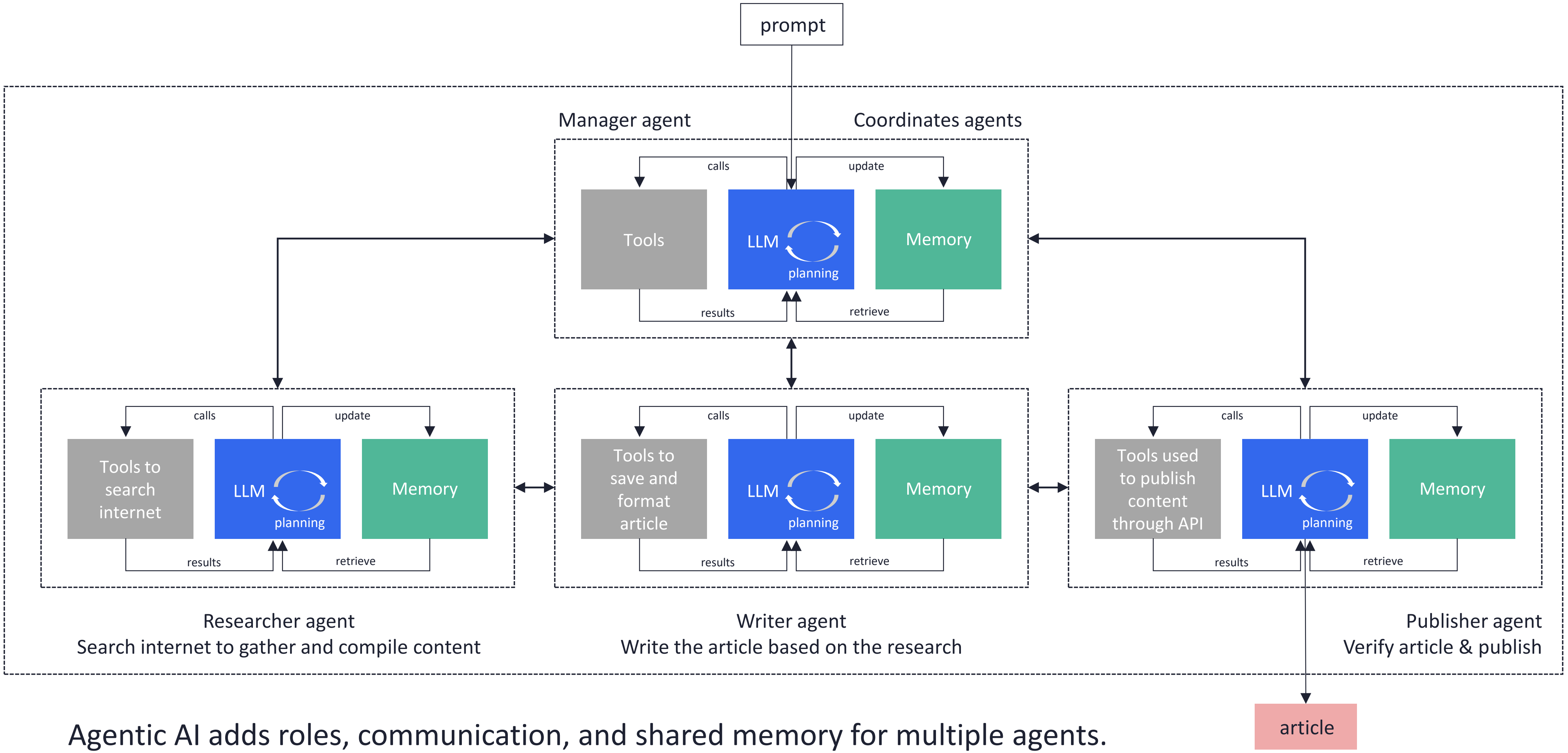
Agentic AI adds roles, communication, and shared memory for multiple agents.

LLM -> AI Agent -> Agentic AI (sample architecture)



Agentic AI adds roles, communication, and **shared memory** for multiple agents.

LLM -> AI Agent -> Agentic AI (sample architecture)



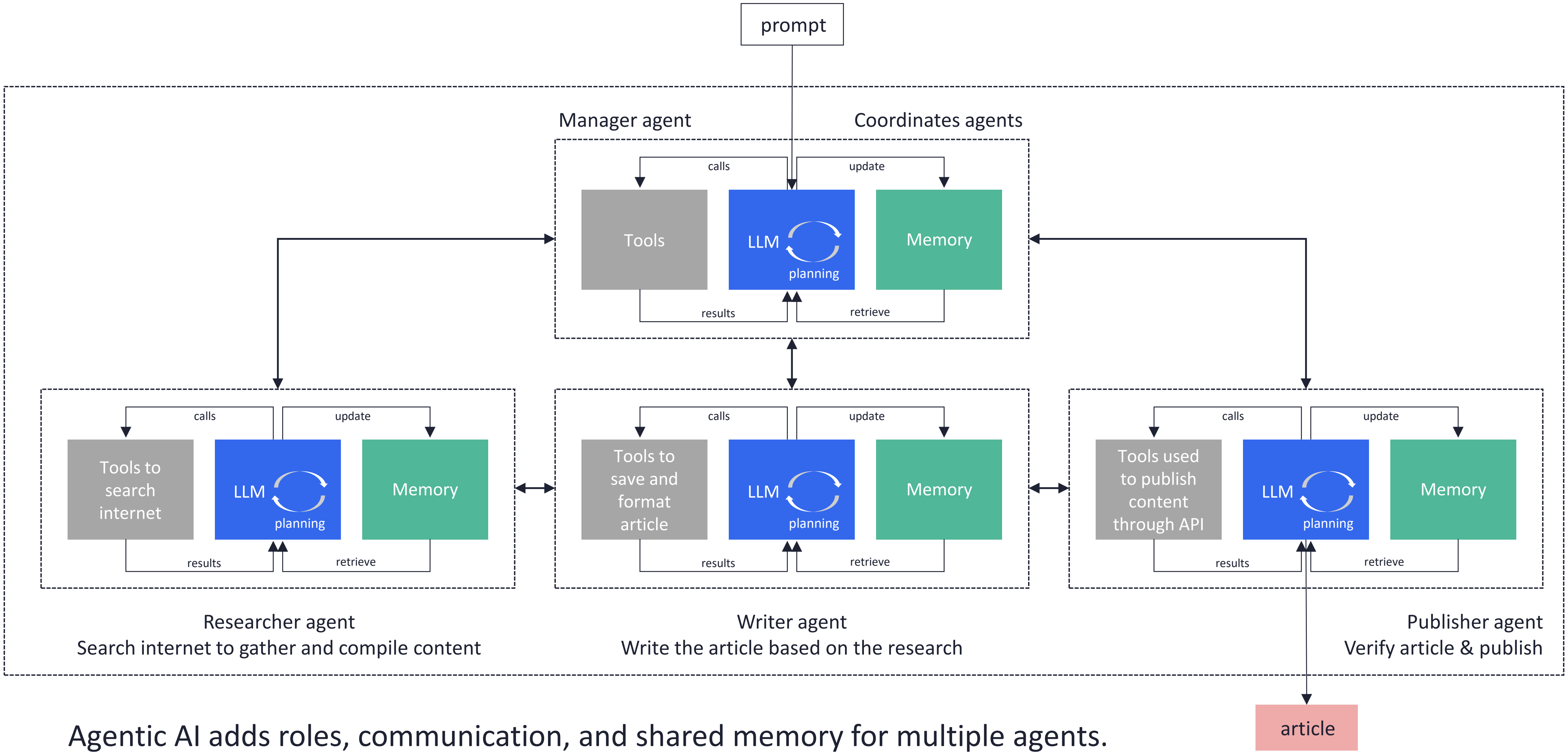
Agentic AI adds roles, communication, and shared memory for multiple agents.

Some frameworks for agent development

Provider & Framework	Notes	Usage example
LangChain	Open-source toolkit for LLMs, provides support for agent logic, can store and recall previous tool outputs and chain tools	ChemCrow
Hugging Face smolagents	Open-source toolkit supports models from OpenAI and Anthropic; integrates tools (registry) and ReAct-style reasoning	AstaBench agents (Allen AI Institute)
OpenAI Function calling	Function calling allows model to output a JSON object calling a tool, with the list of tools passed to the LLM	ChatGPT Agent mode
Anthropic Claude 2 and Agent SDK	Model Context Protocol (MCP) standard by Anthropic defines standard protocol for tool discovery and access, including security	CrewAI framework and now LangChain

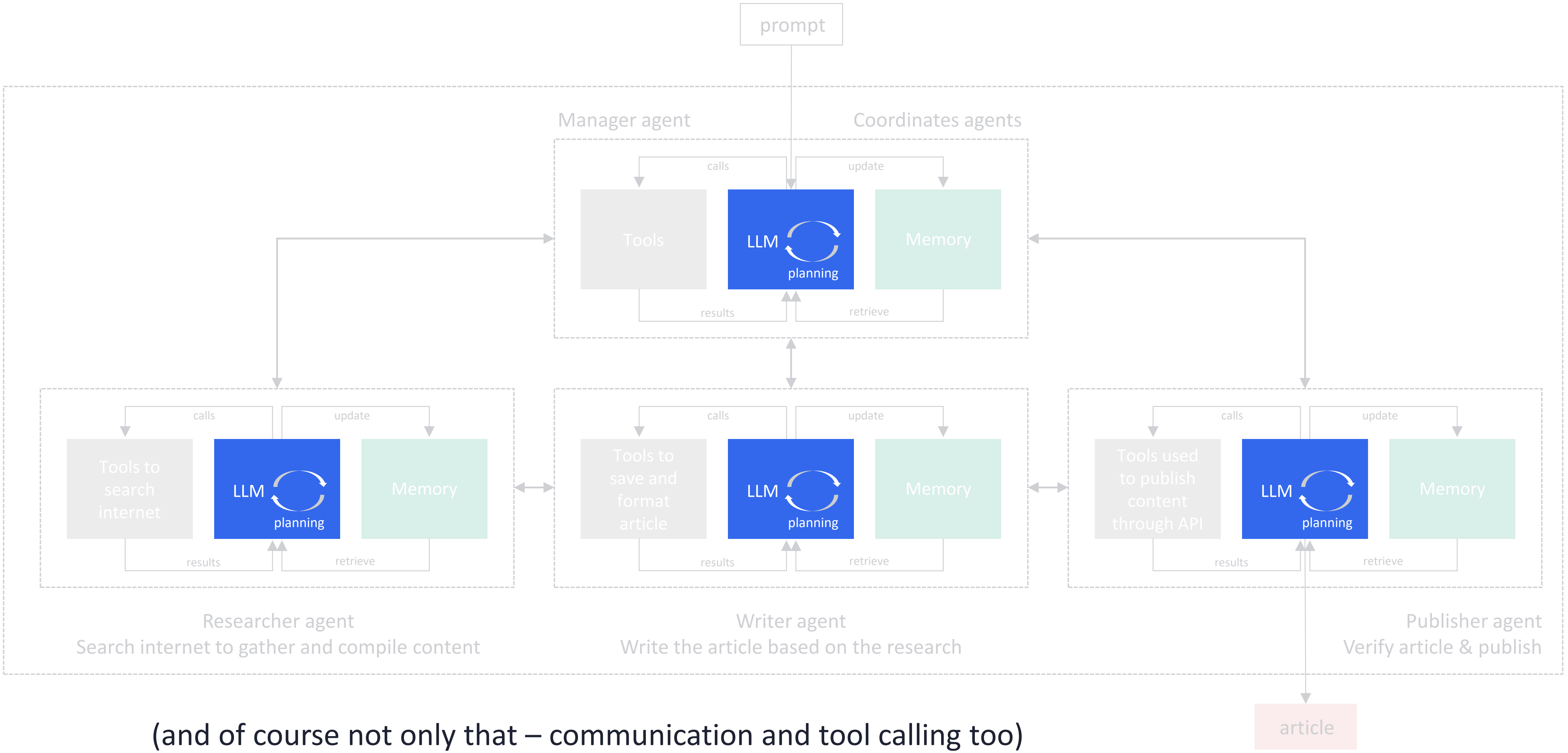
Some bioinformatics researchers code their own. Other solutions targeting enterprise include Amazon (Bedrock), IBM (WatsonX)

AI Agents and Agentic AI rely on LLMs for planning



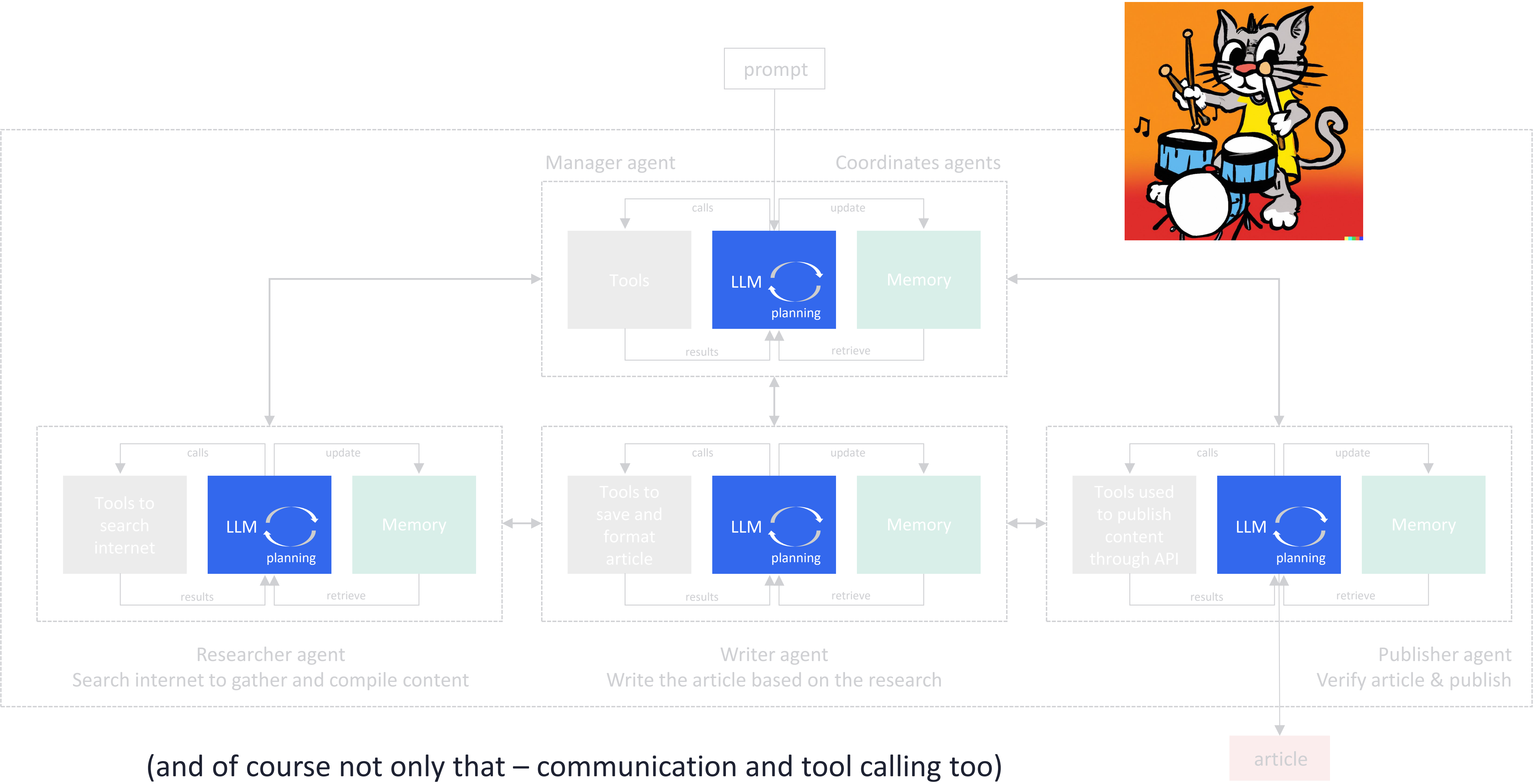
Agentic AI adds roles, communication, and shared memory for multiple agents.

AI Agents and Agentic AI rely on LLMs for planning



(and of course not only that – communication and tool calling too)

AI Agents and Agentic AI rely on LLMs for planning



(and of course not only that – communication and tool calling too)

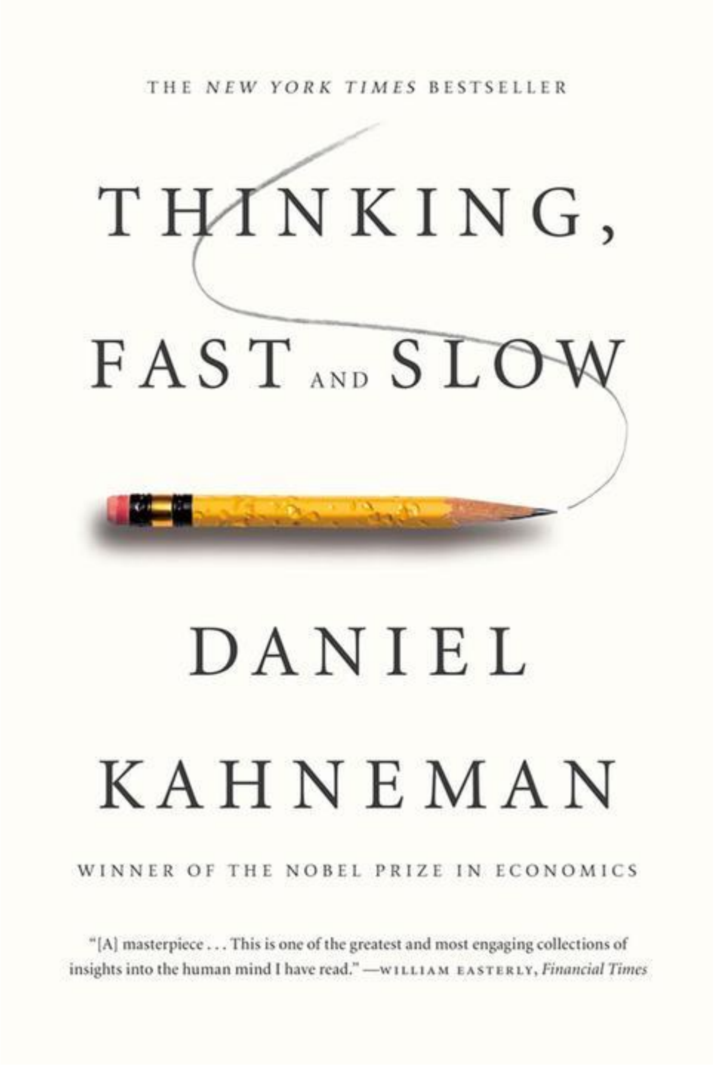
How to improve planning in AI agents?

Helping LLMs “reason”& “plan”

Research suggests two modes of human decision making exist

- **System 1** - Fast, automatic, unconscious

- **System 2** - Slow, deliberate, conscious



A great book but not obligatory for this review

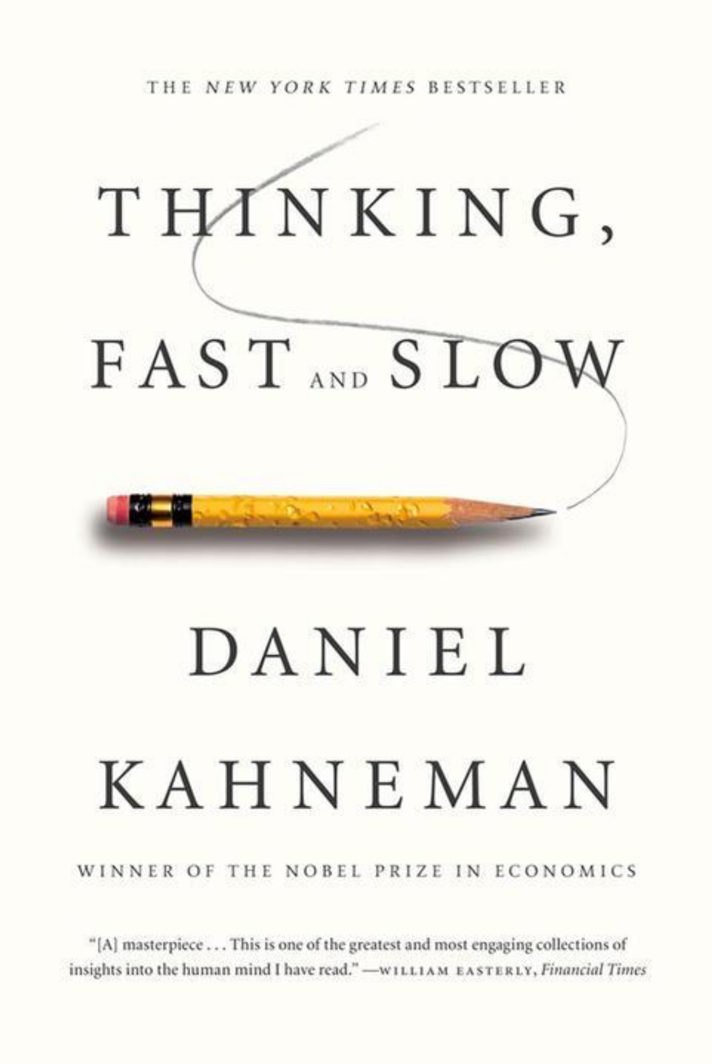
Helping LLMs “reason”& “plan”

Research suggests two modes of human decision making exist

- **System 1** - Fast, automatic, unconscious

  I saw a lion approaching and ran.

- **System 2** - Slow, deliberate, conscious



A great book but not obligatory for this review

Helping LLMs “reason”& “plan”

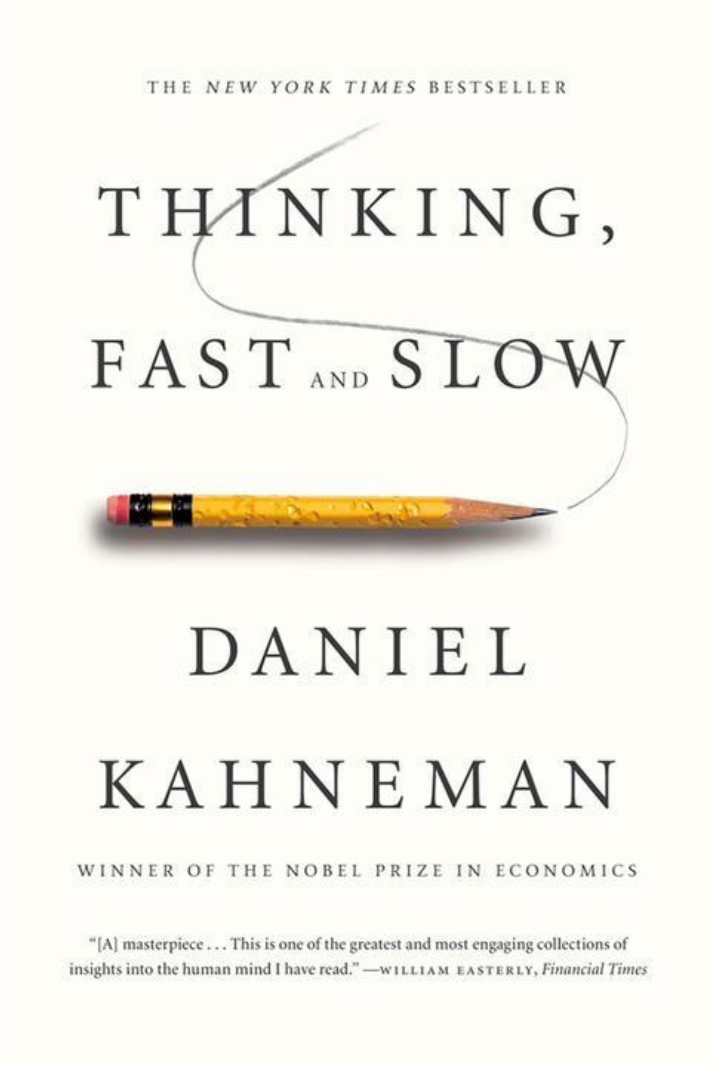
Research suggests two modes of human decision making exist

- **System 1** - Fast, automatic, unconscious

  I saw a lion approaching and ran.

- **System 2** - Slow, deliberate, conscious

  I went to the market and bought ten apples. I gave 2 apples to the neighbour and 2 to the repairman. I then went and bought 5 more apples and ate 1. ***How many apples are left?***

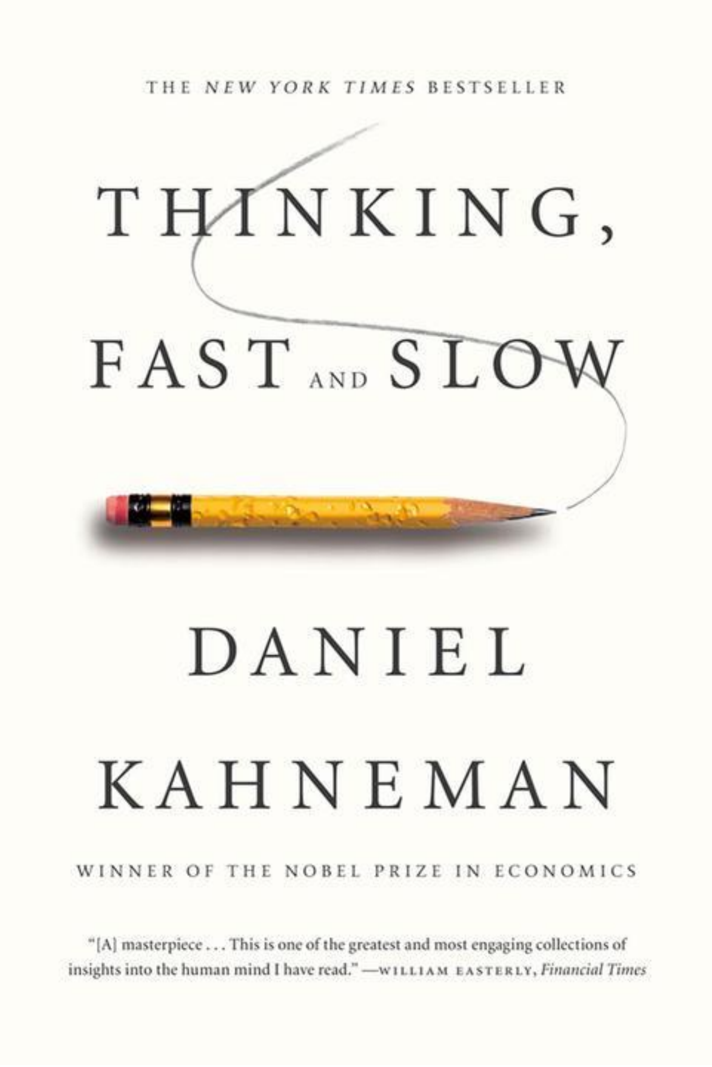


A great book but not obligatory for this review

Helping LLMs "reason"& "plan"

Research suggests two modes of human decision making exist

- **System 1** - Fast, automatic, unconscious
- **System 2** - Slow, deliberate, conscious



A great book but not obligatory for this review

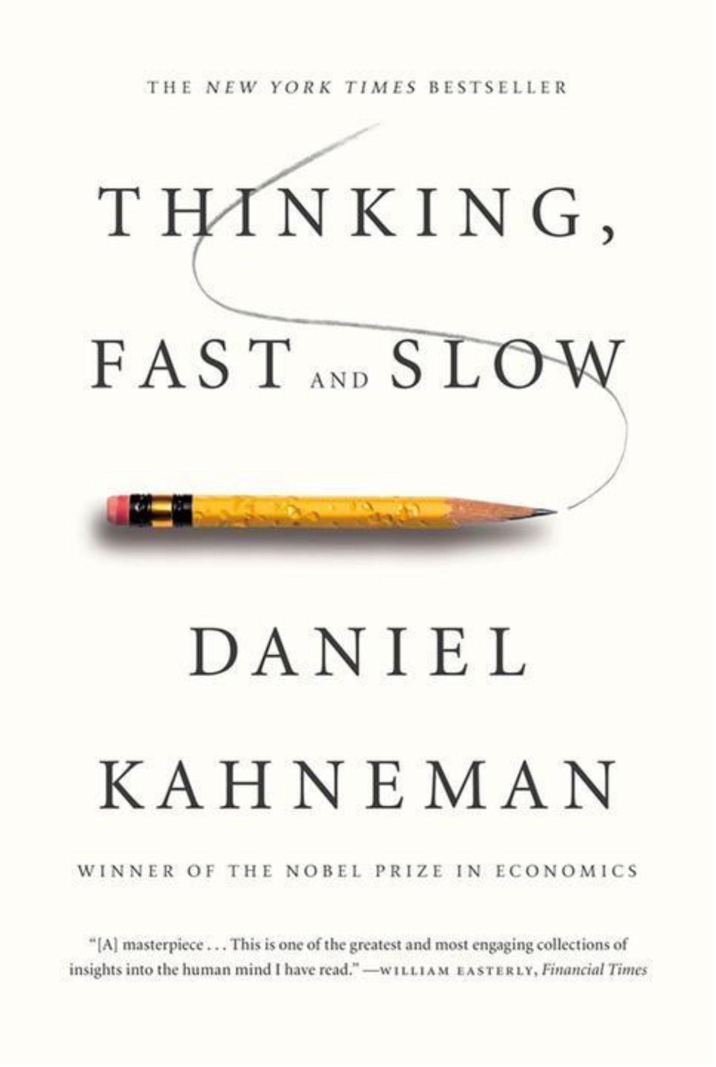
Helping LLMs "reason"& "plan"

Research suggests two modes of human decision making exist

- **System 1** - Fast, automatic, unconscious

  LLM token choices resemble System 1

- **System 2** - Slow, deliberate, conscious



A great book but not obligatory for this review

Helping LLMs “reason”& “plan”

Research suggests two modes of human decision making exist

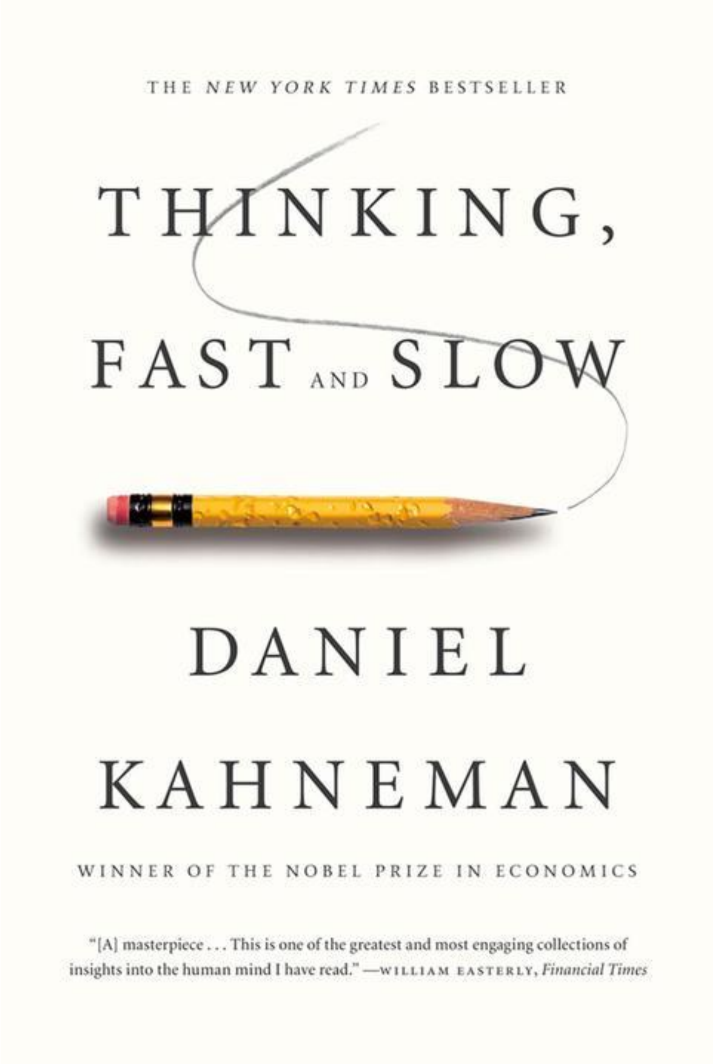
- **System 1** - Fast, automatic, unconscious

  LLM token choices resemble System 1

- **System 2** - Slow, deliberate, conscious

  Augmenting LLMs with System 2-like processes might improve their planning

  (Yao *et al*., arXiv:2305.10601v2)



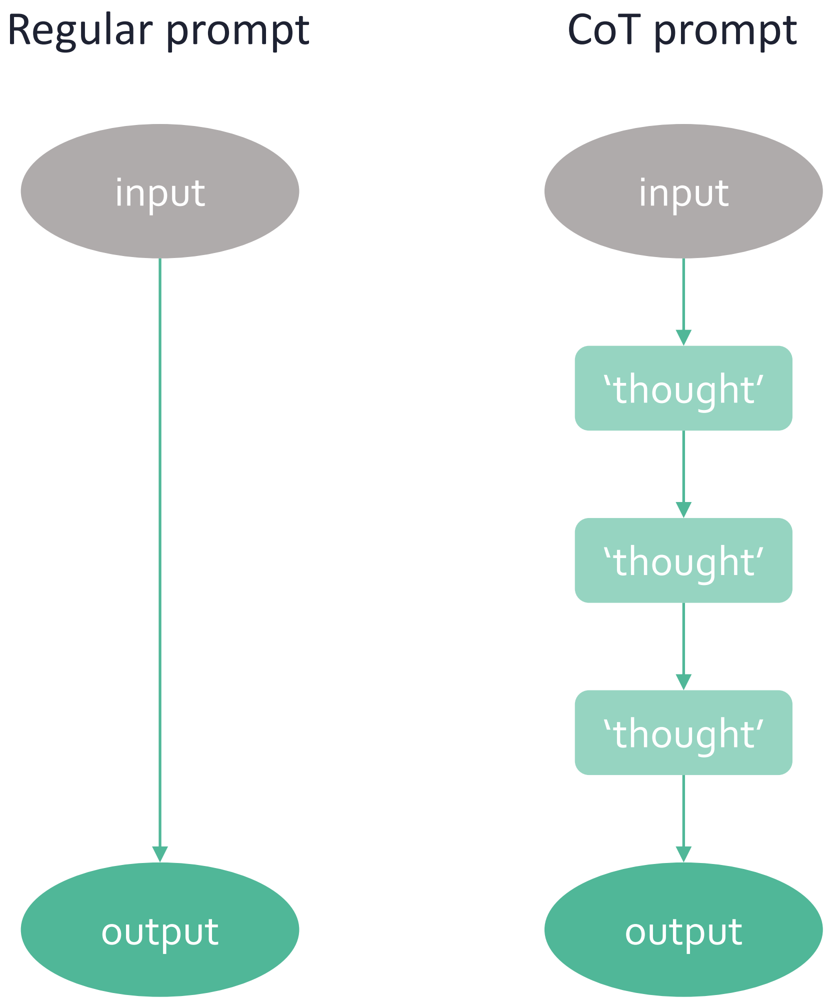
A great book but not obligatory for this review

Approaches to improve LLM “reasoning”

- Prompt-based approaches to “elicit reasoning” (test time)
  - Chain of Thoughts – single path exploration
  - Tree of Thoughts – multi path exploration
  - Graph of Thoughts – multi path exploration
  - ReAct – interleaving path exploration (CoT) and tools
- Creating Large Reasoning Models (LRMs) (post training)

This is background for you. I don’t expect an in-depth review on progress in LLM based reasoning.

Chain of Thought prompting for LLMs



"generating a chain of thought -- a series of intermediate reasoning steps -- significantly improves the ability of large language models to perform complex reasoning."

-> Break down the problem in steps

"Chain-of-Thought Prompting Elicits Reasoning in Large Language Models"

Wei *et al.*, 2022, Google Research

arXiv:2201.11903v6

Chain of Thought prompting for LLMs – example

Regular prompt

I went to the market and bought ten apples. I gave 2 apples to the neighbour and 2 to the repairman.
I then went and bought 5 more apples and ate 1.
How many apples do I have left?

Chain of Thought prompting for LLMs – example

Regular prompt

I went to the market and bought ten apples. I gave 2 apples to the neighbour and 2 to the repairman.
I then went and bought 5 more apples and ate 1.
How many apples do I have left?

Answer: 11 apples

Chain of Thought prompting for LLMs – example

Regular prompt

I went to the market and bought ten apples. I gave 2 apples to the neighbour and 2 to the repairman.
I then went and bought 5 more apples and ate 1.
How many apples do I have left?

Answer: 11 apples

WRONG

Chain of Thought prompting for LLMs – example

Regular prompt

I went to the market and bought ten apples. I gave 2 apples to the neighbour and 2 to the repairman.
I then went and bought 5 more apples and ate 1.
How many apples do I have left?

Answer: 11 apples

WRONG

Chain of Thought prompting for LLMs – example

Regular prompt

I went to the market and bought ten apples. I gave 2 apples to the neighbour and 2 to the repairman.
I then went and bought 5 more apples and ate 1.
How many apples do I have left?

Answer: 11 apples

WRONG

Chain of Thought prompt

I went to the market and bought ten apples. I gave 2 apples to the neighbour and 2 to the repairman.
I then went and bought 5 more apples and ate 1.
How many apples do I have left? <u>Let's think step by step.</u>

Chain of Thought prompting for LLMs – example

Regular prompt

I went to the market and bought ten apples. I gave 2 apples to the neighbour and 2 to the repairman.
I then went and bought 5 more apples and ate 1.
How many apples do I have left?

Answer: 11 apples

WRONG

Chain of Thought prompt

I went to the market and bought ten apples. I gave 2 apples to the neighbour and 2 to the repairman.
I then went and bought 5 more apples and ate 1.
How many apples do I have left? <u>Let's think step by step.</u>

First, you started with 10 apples.

Chain of Thought prompting for LLMs – example

Regular prompt

I went to the market and bought ten apples. I gave 2 apples to the neighbour and 2 to the repairman.
I then went and bought 5 more apples and ate 1.
How many apples do I have left?

Answer: 11 apples

WRONG

Chain of Thought prompt

I went to the market and bought ten apples. I gave 2 apples to the neighbour and 2 to the repairman.
I then went and bought 5 more apples and ate 1.
How many apples do I have left? <u>Let's think step by step.</u>

First, you started with 10 apples.
You gave away 2 apples to the neighbor and 2 to the repairman, so you had 6 apples left.

Chain of Thought prompting for LLMs – example

Regular prompt

I went to the market and bought ten apples. I gave 2 apples to the neighbour and 2 to the repairman.
I then went and bought 5 more apples and ate 1.
How many apples do I have left?

Answer: 11 apples

WRONG

Chain of Thought prompt

I went to the market and bought ten apples. I gave 2 apples to the neighbour and 2 to the repairman.
I then went and bought 5 more apples and ate 1.
How many apples do I have left? Let's think step by step.

First, you started with 10 apples.
You gave away 2 apples to the neighbor and 2 to the repairman, so you had 6 apples left.
Then you bought 5 more apples, so now you had 11 apples.

SIB

Chain of Thought prompting for LLMs – example

Regular prompt

I went to the market and bought ten apples. I gave 2 apples to the neighbour and 2 to the repairman.
I then went and bought 5 more apples and ate 1.
How many apples do I have left?

Answer: 11 apples

WRONG

Chain of Thought prompt

I went to the market and bought ten apples. I gave 2 apples to the neighbour and 2 to the repairman.
I then went and bought 5 more apples and ate 1.
How many apples do I have left? Let's think step by step.

First, you started with 10 apples.
You gave away 2 apples to the neighbor and 2 to the repairman, so you had 6 apples left.
Then you bought 5 more apples, so now you had 11 apples.
Finally, you ate 1 apple, so you would have 10 apples.

SIB

Chain of Thought prompting for LLMs – example

Regular prompt

I went to the market and bought ten apples. I gave 2 apples to the neighbour and 2 to the repairman.
I then went and bought 5 more apples and ate 1.
How many apples do I have left?

Answer: 11 apples

WRONG

Chain of Thought prompt

I went to the market and bought ten apples. I gave 2 apples to the neighbour and 2 to the repairman.
I then went and bought 5 more apples and ate 1.
How many apples do I have left? **<u>Let's think step by step.</u>**

First, you started with 10 apples.
You gave away 2 apples to the neighbor and 2 to the repairman, so you had 6 apples left.
Then you bought 5 more apples, so now you had 11 apples.
Finally, you ate 1 apple, so you would have 10 apples.

Answer: 10 apples

Chain of Thought prompting for LLMs – example

Regular prompt

I went to the market and bought ten apples. I gave 2 apples to the neighbour and 2 to the repairman.
I then went and bought 5 more apples and ate 1.
How many apples do I have left?

Answer: 11 apples

WRONG

Chain of Thought prompt

I went to the market and bought ten apples. I gave 2 apples to the neighbour and 2 to the repairman.
I then went and bought 5 more apples and ate 1.
How many apples do I have left? <u>Let's think step by step.</u>

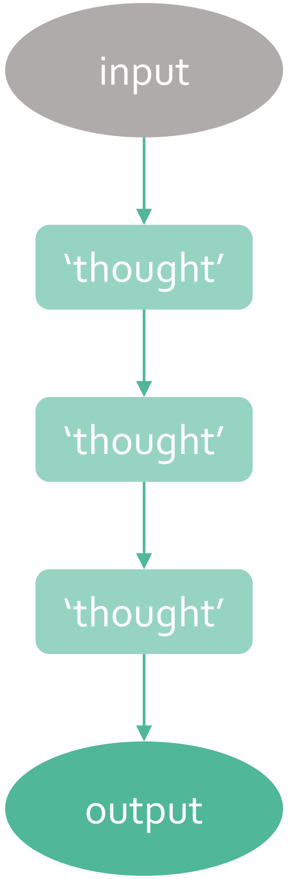
First, you started with 10 apples.
You gave away 2 apples to the neighbor and 2 to the repairman, so you had 6 apples left.
Then you bought 5 more apples, so now you had 11 apples.
Finally, you ate 1 apple, so you would have 10 apples.

Answer: 10 apples

RIGHT

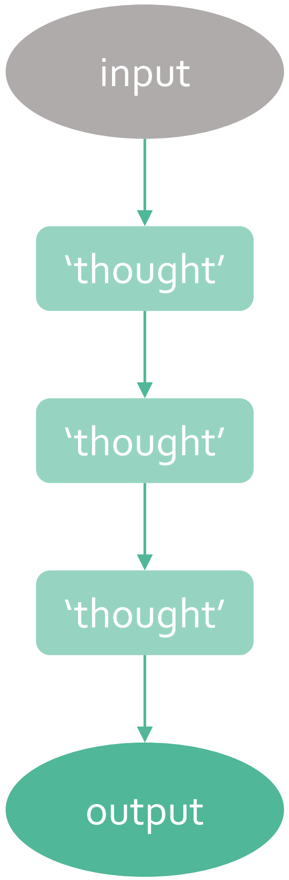
From Chains to Trees of Thoughts

Chain of Thoughts (CoT)

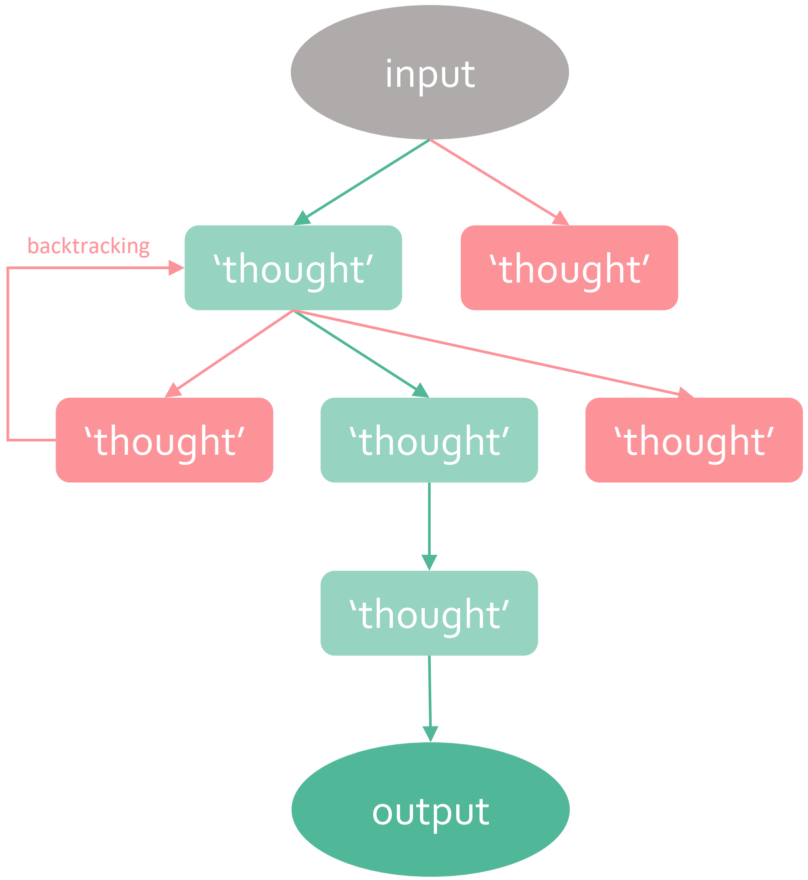


From Chains to Trees of Thoughts

Chain of Thoughts (CoT)



Tree of Thoughts (ToT)



The LLM generates & evaluates multiple "thoughts" – heuristic approach seemingly characteristic of human thinking

Requires external memory and controller to store/probe paths (e.g. LangChain ToT module)

"Tree of Thoughts: Deliberate Problem Solving with Large Language Models"

Yao *et al.*, 2023, at Google DeepMind & Princeton arXiv:2305.10601v2

Graph of Thoughts for LLMs

Generalizes Tree of Thoughts. Instead of a strict tree, reasoning is modeled as a graph, where thoughts can merge, reference, or reuse one another.

Again, this requires external memory and controller to store/probe paths (e.g. …)

“Beyond Chain-of-Thought, Effective Graph-of-Thought Reasoning in Language Models”

Yao *et al.*, 2023, at Shanghai and Wuhan

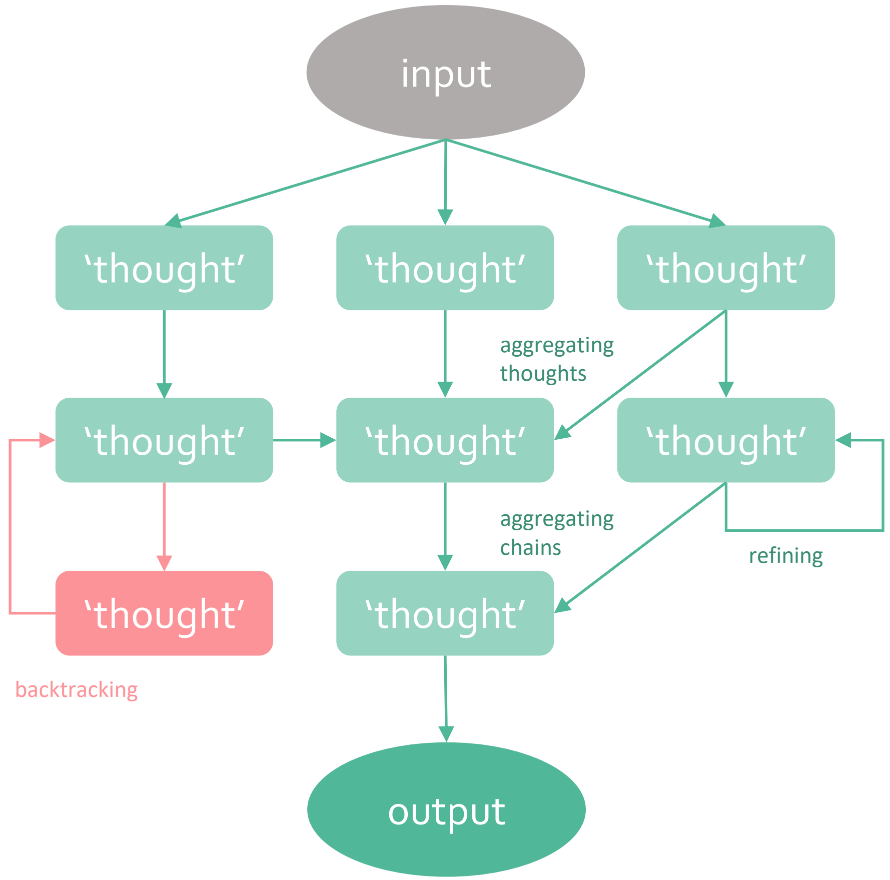
arXiv:2305.16582v2

“Graph of Thoughts: Solving Elaborate Problems with Large Language Models”

Besta *et al.*, 2023, at ETH Zurich

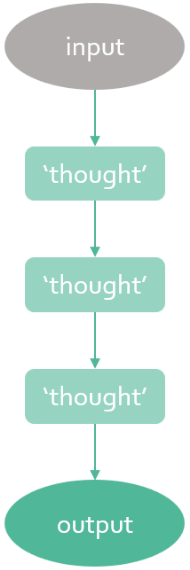
arXiv:2308.09687

Graph of Thoughts (GoT)

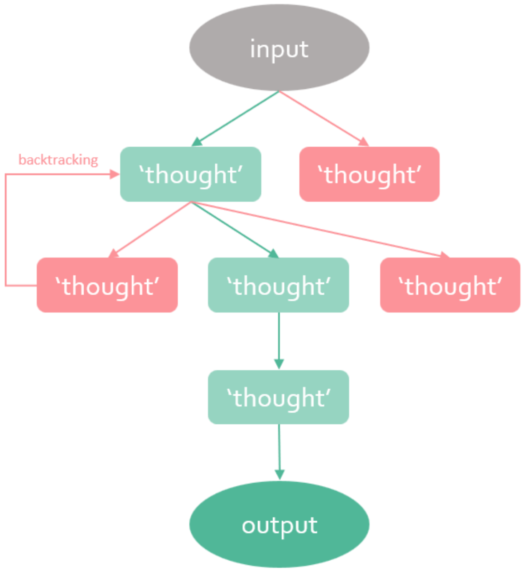


Recap: path exploration to explicit “reasoning” in LLMs

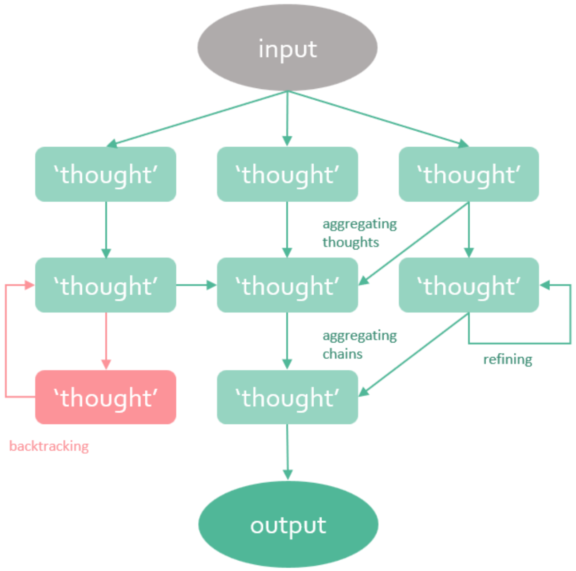
Chain of Thoughts (CoT)



Tree of Thoughts (ToT)



Graph of Thoughts (GoT)



	Chain of Thoughts (CoT)	Tree of Thoughts (ToT)	Graph of Thoughts (GoT)
Structure	Single linear sequence	Hierarchical branches	Network
LLM calls	One	Many (branch expansions)	Many (plus merging)
Memory	In-context only	External tree memory	External graph memory
Control	Model-driven	External controller (search)	External controller + memory manager
Analogy	One person thinking aloud	Trying multiple strategies	Multiple people brainstorming & combining ideas

Tree of Thoughts for LLMs – performance

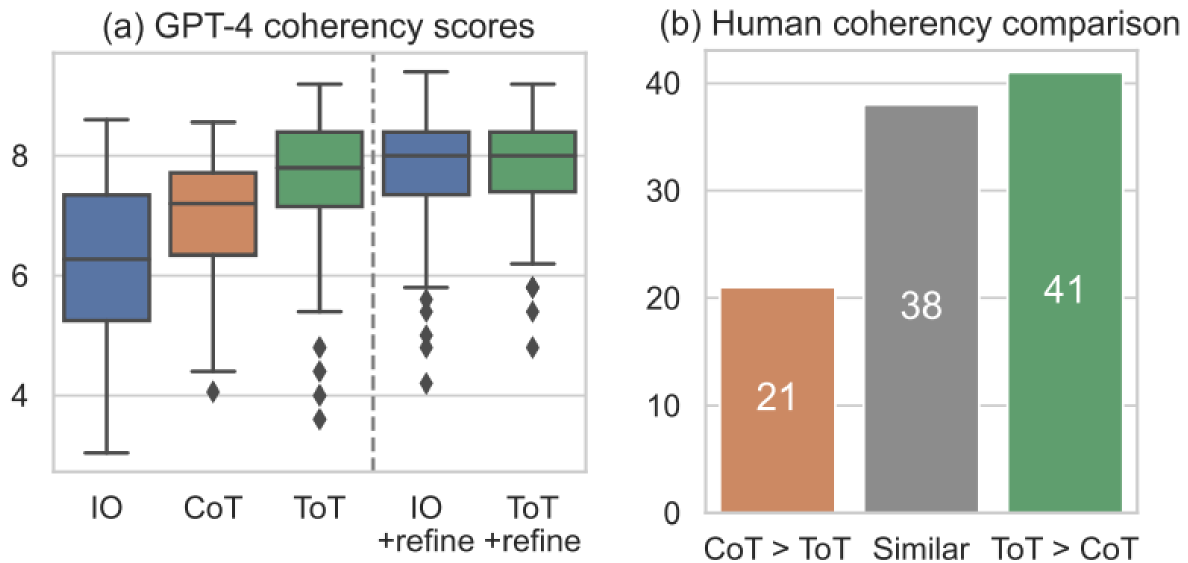


Figure 5: Creative Writing results.

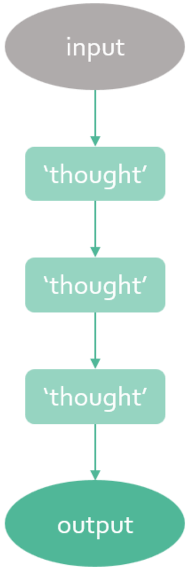
Method	Success Rate (%) Letter	Word	Game
IO	38.7	14	0
CoT	40.6	15.6	1
ToT (ours)	**78**	**60**	**20**
+best state	82.4	67.5	35
-prune	65.4	41.5	5
-backtrack	54.6	20	5

Table 3: Mini Crosswords results.

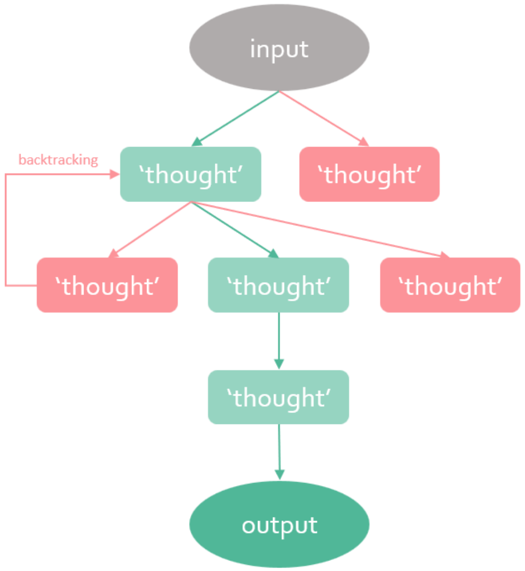
"Tree of Thoughts: Deliberate Problem Solving with Large Language Models", Yao *et al*., arXiv:2305.10601v2

Recap: path exploration to elicit “reasoning” in LLMs

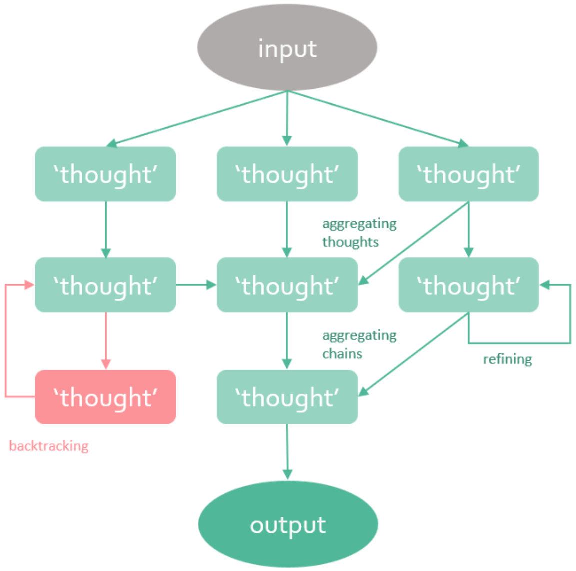
Chain of Thoughts (CoT)



Tree of Thoughts (ToT)



Graph of Thoughts (GoT)



Structure	Single linear sequence	Hierarchical branches	Network
LLM calls	One	Many (branch expansions)	Many (plus merging)
Memory	In-context only	External tree memory	External graph memory
Control	Model-driven	External controller (search)	External controller + memory manager
Analogy	One person thinking aloud	Trying multiple strategies	Multiple people brainstorming & combining ideas

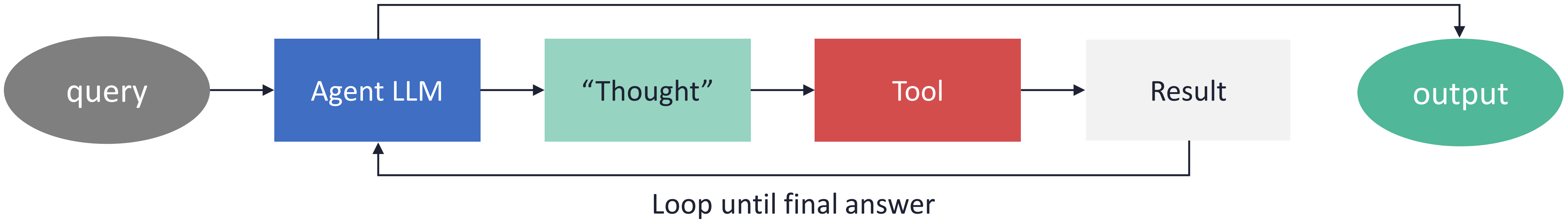
ReAct agent loops

“ReAct: Synergizing Reasoning and Acting in Language Models”

Yao *et al*., 2022, Google Research and Princeton, arXiv:2210.03629v3

ReAct prompting guides model to

- use chain of thought reasoning
- take actions & use tools
- make observations
- loop
- output final answer



ReAct’s combination of CoT with a connection to information sources reduces hallucinations

Adapted from https://www.ibm.com/think/topics/react-agent

ReAct performance

ReAct improves performance and reduces hallucination as a source of failure.

Source: *ReAct: Synergizing Reasoning and Acting in Language Models*, Yao *et al.*, 2022, arXiv:2210.03629

Prompt Method[a]	**HotpotQA** (EM)	**Fever** (Acc)
`Standard`	28.7	57.1
`CoT` (Wei et al., 2022)	29.4	56.3
`CoT-SC` (Wang et al., 2022a)	33.4	60.4
`Act`	25.7	58.9
`ReAct`	27.4	60.9
`CoT-SC` → `ReAct`	34.2	**64.6**
`ReAct` → `CoT-SC`	**35.1**	62.0
Supervised SoTA[b]	67.5	89.5

HotPotQA is a multi-hop QA benchmark that requires reasoning over two or more Wikipedia passages. FEVER is a fact verification benchmark Wikipedia passage to verify the claim.

	Type	Definition	`ReAct`	`CoT`
Success	True positive	Correct reasoning trace and facts	94%	86%
	False positive	Hallucinated reasoning trace or facts	6%	14%
Failure	Reasoning error	Wrong reasoning trace (including failing to recover from repetitive steps)	47%	16%
	Search result error	Search return empty or does not contain useful information	23%	-
	Hallucination	Hallucinated reasoning trace or facts	0%	56%
	Label ambiguity	Right prediction but did not match the label precisely	29%	28%

Table 2: Types of success and failure modes of `ReAct` and `CoT` on HotpotQA, as well as their percentages in randomly selected examples studied by human.

Approaches to improve LLM “reasoning”

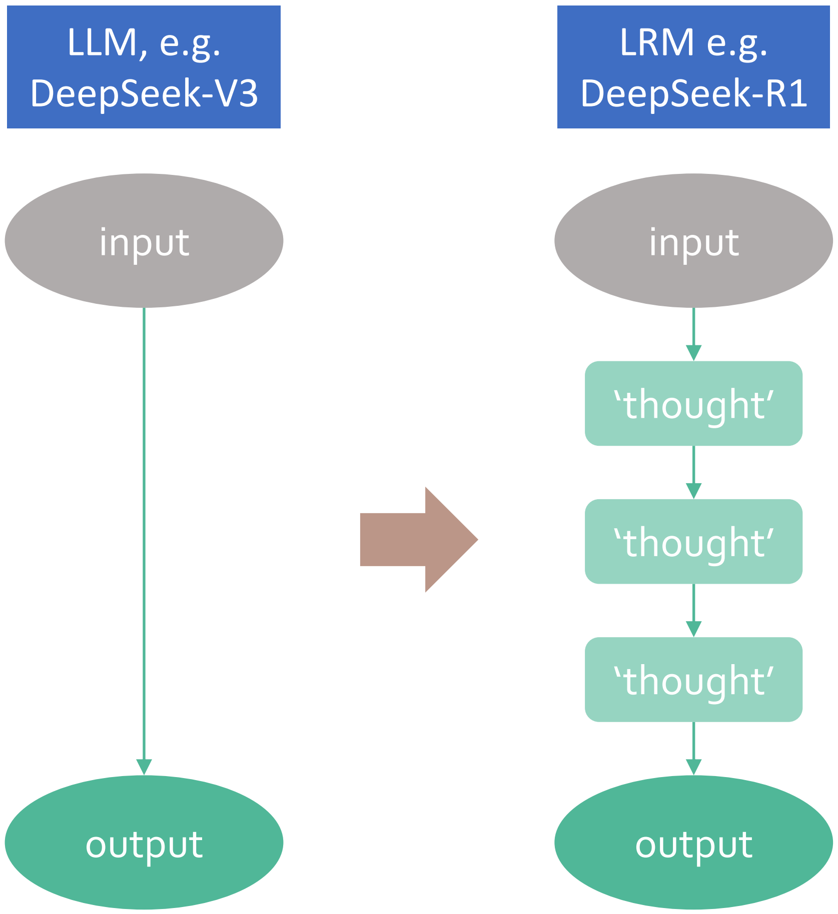
- Prompt-based approaches to “elicit reasoning” (<u>test time</u>)

    - Chain of Thoughts – single path exploration
    - Tree of Thoughts – multi path exploration
    - Graph of Thoughts – multi path exploration
    - ReAct – interleaving path exploration (CoT) and tools

- Creating Large Reasoning Models (LRMs) (<u>post training</u>)

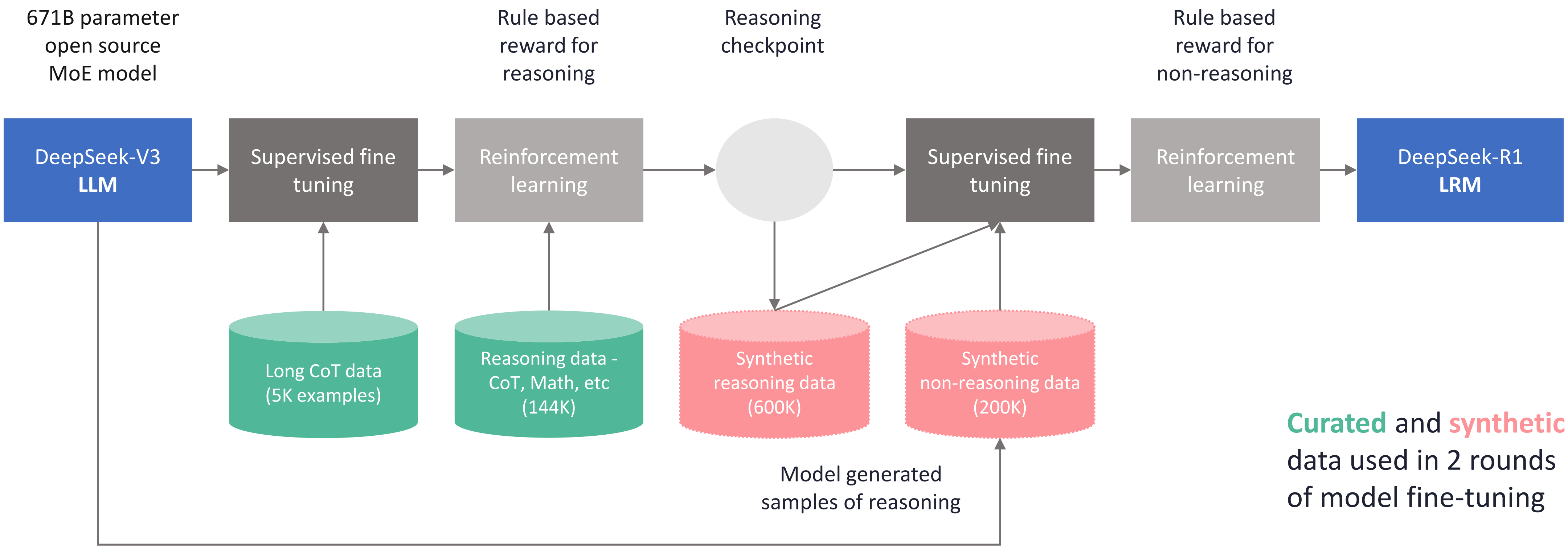
Large Reasoning Models (LRMs)

- LRMs are post-trained to break down problems into steps and “reason”
- More tokens – and more compute time – is spent at test time inferencing
- Examples include:
  - **DeepSeek R1**
  - OpenAI-o1/o3
  - Google Gemini Flash 2.0 Thinking
  - Alibaba Open QwQ



Illustrate with DeepSeek-R1 – open source, for open agents, **well described training process**

How DeepSeek-R1 was trained



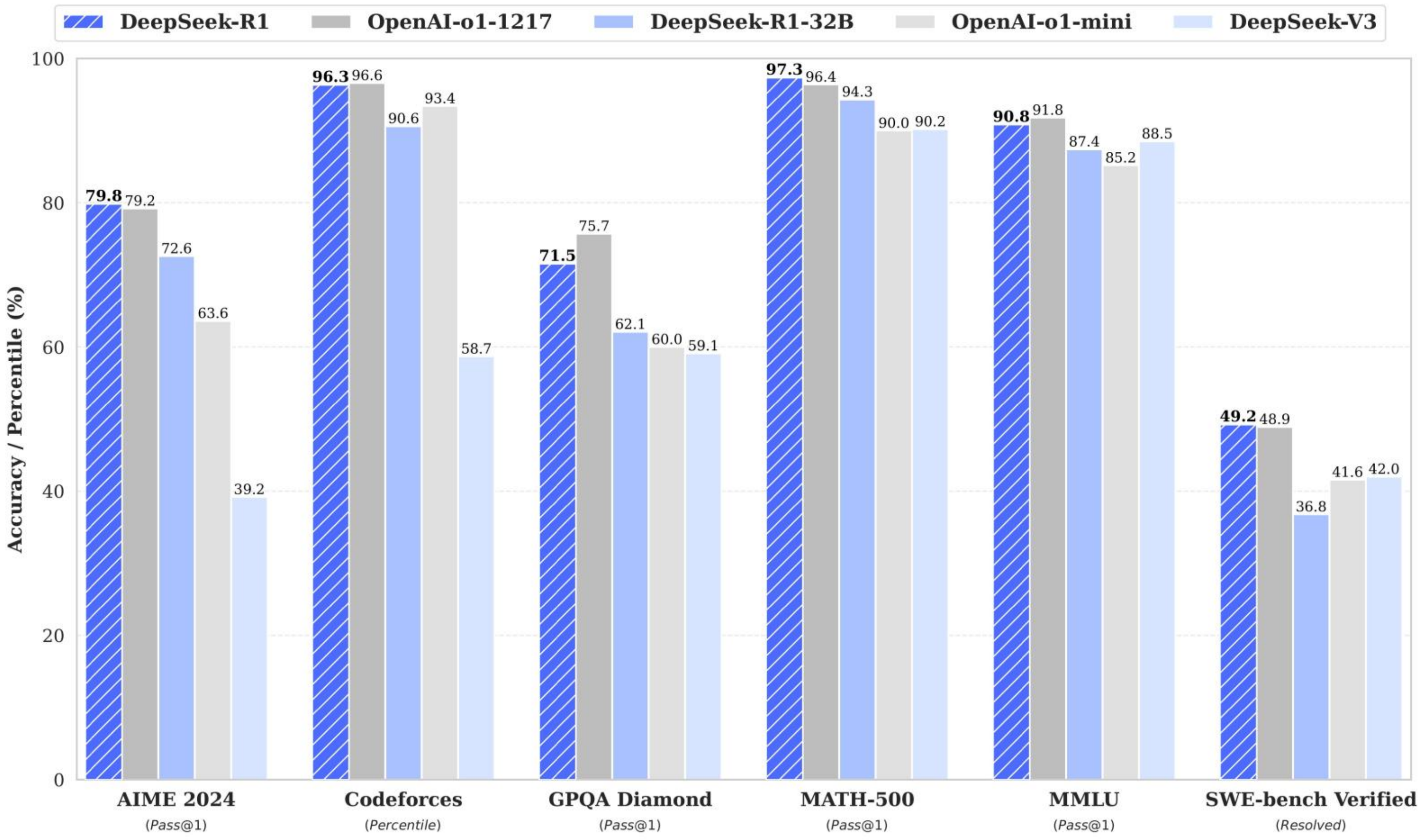
Curated and synthetic data used in 2 rounds of model fine-tuning

Adapted from https://medium.com/%40lmpo/deepseek-r1-affordable-efficient-and-state-of-the-art-ai-reasoning-f293b0bd8d65
See also https://newsletter.maartengrootendorst.com/p/a-visual-guide-to-reasoning-llms

DeepSeek-R1 performance

6 benchmarks with a variety of subjects and levels of reasoning:

- AIME2024 (math)
- Codeforces (prog)
- GPQA (science)
- MATH-500 (math)
- MMLU (57 subjects)
- SWE (software)



From https://huggingface.co/deepseek-ai/DeepSeek-R1

Recap: approaches to improve LLM “reasoning”

- Prompt-based approaches to “elicit reasoning” (test time)
  - Chain of Thoughts – single path exploration
  - Tree of Thoughts – multi path exploration
  - Graph of Thoughts – multi path exploration
  - ReAct – interleaving path exploration (CoT) and tools
- Creating Large Reasoning Models (LRMs) (post training)

Q: do LLMs actually reason?

Observations, FYI:

- LLM reasoning breaks down on new puzzles (out of domain)
- LLMs can arrive at correct solutions despite creating inconsistent chains of thoughts
- LLMs are trained on vast corpora of text that includes examples of people doing reasoning, so they learn to emit tokens that simulate the process of reasoning

Refs:

Survey on Enhancing Causal Reasoning Ability of Large Language Models, Li *et al.*, arXiv:2503.09326

Is Chain-of-Thought Reasoning of LLMs a Mirage? A Data Distribution Lens, Zhao *et al.* arXiv:2508.01191v2

The Illusion of Thinking: Understanding the Strengths and Limitations of Reasoning Models via the Lens of Problem Complexity, Shoajee *et al.*, arXiv:2506.06941v2

Unveiling Causal Reasoning in Large Language Models: Reality or Mirage?, Chi *et al.* arXiv:2506.21215v1

Stop Anthropomorphizing Intermediate Tokens as Reasoning/Thinking Traces!, Kampbampati *et al.*, arXiv:2504:09762v2

Anthropomorphizing models by claiming they reason is common in the literature.

LLMs and LRMs can effectively mimic reasoning but remain next token generators.

CoT and other approaches elicit reasoning mimicry in LLMs.
Are there ways we could make agents reason deterministically?

One solution may be to remove this responsibility from the LLM.
Can you find examples of this approach in your review?

Delving into agents

1. LLM basics
2. AI Agent basics: tools and reasoning
3. AI Agents in biology
4. AI Agent evaluation
5. AI Agent Safety & ethics

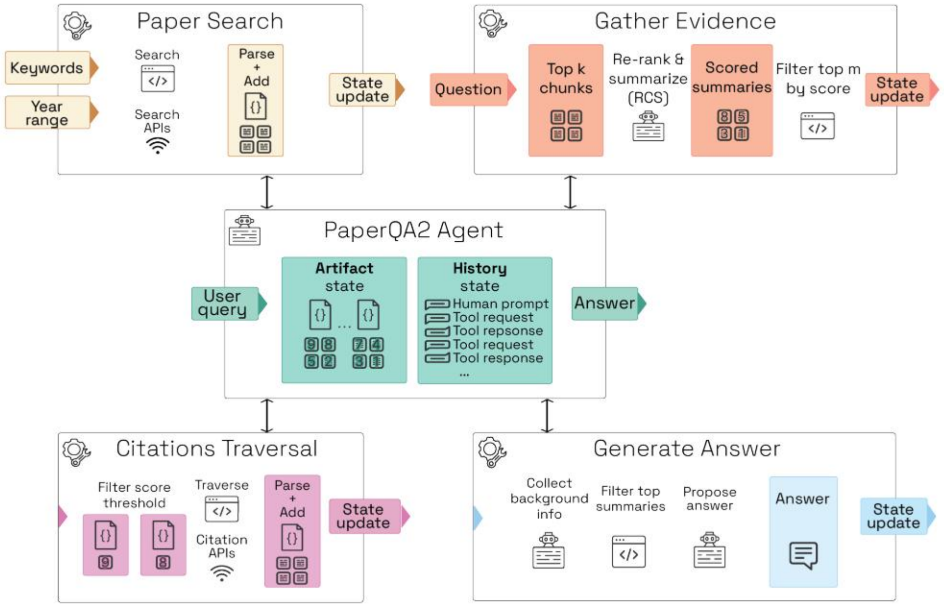
How to reconcile claims around agentic AI with demonstrated reality?

Things to look out for when reading agent papers

Autonomy	What part of the workflow is executed by humans
Iterative optimization	Whether the agent iteratively optimizes or simply executes a single workflow
Controls	Comparing multi agent *vs* single agent *vs* LLM *vs* scripted workflow
Robustness testing	Perturbations like changing prompts, data, LLMs, agent removal and changing roles
Experimental validation	When applicable
Failure analysis	Planning, communication, memory, tools
Code/tool availability	Is the agent reproducible/testable? Does it use proprietary models?
Benchmark suitability	If LLMs perform well it may really be testing intrinsic knowledge; data leakage may be an issue

Example 1 – PaperQA2

PaperQA2 from FutureHouse answers questions on papers and write Wikipedia style summaries



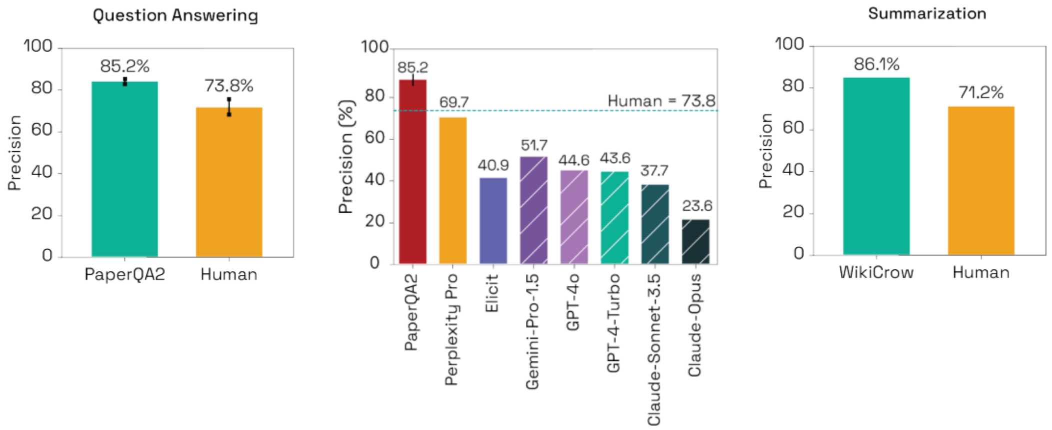
Single agent: LangChain & GPT-4 Turbo for orchestration

FutureHouse

[arXiv:2409.13740](https://arxiv.org/abs/2409.13740)

LANGUAGE AGENTS ACHIEVE SUPERHUMAN SYNTHESIS OF SCIENTIFIC KNOWLEDGE

Michael D. Skarlinski[1] Sam Cox[1,2] Jon M. Laurent[1]
James D. Braza[1] Michaela Hinks[1] Michael J. Hammerling[1]
Manvitha Ponnapati[1] Samuel G. Rodriques[1,3*] Andrew D. White[1,2*]



Curated test set: questions whose answers appear only once in the literature and never in abstracts

Reading of PaperQA2

Autonomy	Low	Orchestrates tool calls within a narrow, pre-defined retrieval pipeline; humans perform environment setup and dataset and eval preparation

Reading of PaperQA2

Autonomy	Low	Orchestrates tool calls within a narrow, pre-defined retrieval pipeline; humans perform environment setup and dataset and eval preparation
Iterative optimization	No	Agent iteratively searches the literature, and can vary order of tool calls, but there is no iterative hypothesis or workflow refinement

Reading of PaperQA2

Autonomy	Low	Orchestrates tool calls within a narrow, pre-defined retrieval pipeline; humans perform environment setup and dataset and eval preparation
Iterative optimization	No	Agent iteratively searches the literature, and can vary order of tool calls, but there is no iterative hypothesis or workflow refinement
Controls	Yes	Agent vs LLM

Reading of PaperQA2

Autonomy	Low	Orchestrates tool calls within a narrow, pre-defined retrieval pipeline; humans perform environment setup and dataset and eval preparation
Iterative optimization	No	Agent iteratively searches the literature, and can vary order of tool calls, but there is no iterative hypothesis or workflow refinement
Controls	Yes	Agent vs LLM
Robustness testing	No	No stress-testing, such as through prompt alterations, adversarial bibliography with a single correct answer, etc

Reading of PaperQA2

Autonomy	Low	Orchestrates tool calls within a narrow, pre-defined retrieval pipeline; humans perform environment setup and dataset and eval preparation
Iterative optimization	No	Agent iteratively searches the literature, and can vary order of tool calls, but there is no iterative hypothesis or workflow refinement
Controls	Yes	Agent vs LLM
Robustness testing	No	No stress-testing, such as through prompt alterations, adversarial bibliography with a single correct answer, etc
Experimental validation	Yes…	In this context validation is testing information retrieval from literature

Reading of PaperQA2

Autonomy	Low	Orchestrates tool calls within a narrow, pre-defined retrieval pipeline; humans perform environment setup and dataset and eval preparation
Iterative optimization	No	Agent iteratively searches the literature, and can vary order of tool calls, but there is no iterative hypothesis or workflow refinement
Controls	Yes	Agent vs LLM
Robustness testing	No	No stress-testing, such as through prompt alterations, adversarial bibliography with a single correct answer, etc
Experimental validation	Yes…	In this context validation is testing information retrieval from literature
Failure analysis	Limited	

Reading of PaperQA2

Autonomy	Low	Orchestrates tool calls within a narrow, pre-defined retrieval pipeline; humans perform environment setup and dataset and eval preparation
Iterative optimization	No	Agent iteratively searches the literature, and can vary order of tool calls, but there is no iterative hypothesis or workflow refinement
Controls	Yes	Agent vs LLM
Robustness testing	No	No stress-testing, such as through prompt alterations, adversarial bibliography with a single correct answer, etc
Experimental validation	Yes…	In this context validation is testing information retrieval from literature
Failure analysis	Limited	
Code/tool availability	Partial	Uses proprietary models like GPT-4 Turbo

Reading of PaperQA2

Autonomy	Low	Orchestrates tool calls within a narrow, pre-defined retrieval pipeline; humans perform environment setup and dataset and eval preparation
Iterative optimization	No	Agent iteratively searches the literature, and can vary order of tool calls, but there is no iterative hypothesis or workflow refinement
Controls	Yes	Agent vs LLM
Robustness testing	No	No stress-testing, such as through prompt alterations, adversarial bibliography with a single correct answer, etc
Experimental validation	Yes…	In this context validation is testing information retrieval from literature
Failure analysis	Limited	
Code/tool availability	Partial	Uses proprietary models like GPT-4 Turbo
Benchmark suitability	Partial	Curated QA pairs, with answers appearing only once and not in abstracts In QA testing, LLMs answer more than half the questions correctly

Summary: PaperQA2 shows single agents are capable of literature review and this is improved by modular prompting. How they deal with conflicting and false claims needs further testing.

Example 2 - The Virtual Lab, a multi-agent AI


nature > articles > article

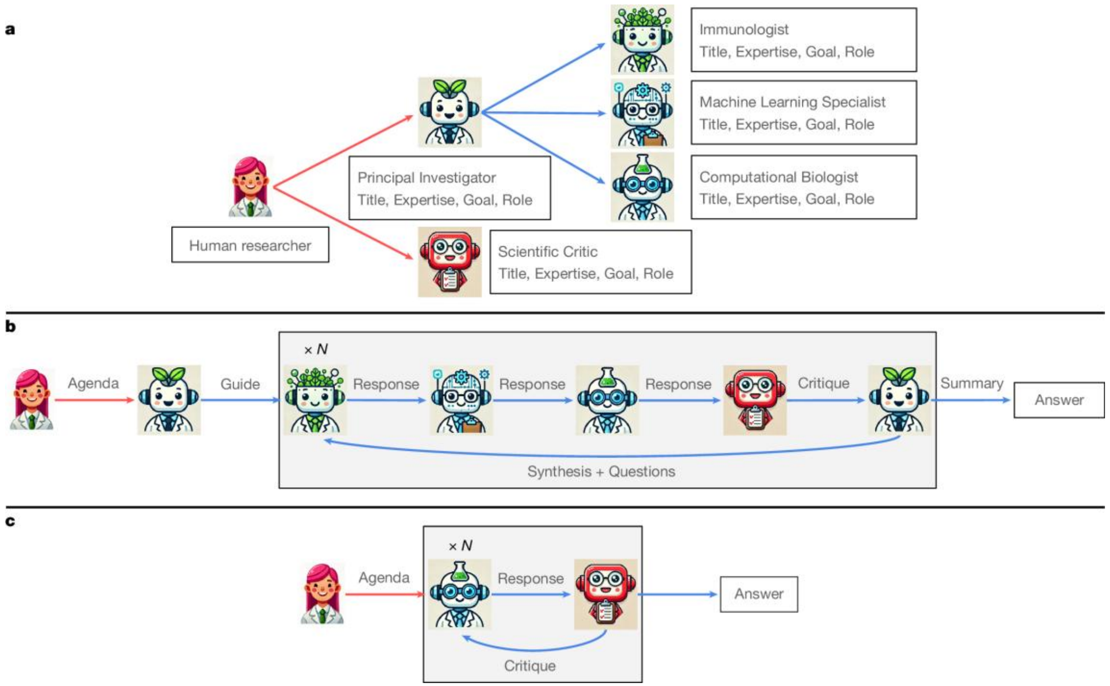
Article | Published: 29 July 2025

The Virtual Lab of AI agents designs new SARS-CoV-2 nanobodies

Kyle Swanson, Wesley Wu, Nash L. Bulaong, John E. Pak & James Zou

Nature **646**, 716–723 (2025) | Cite this article

47k Accesses | **22** Citations | **541** Altmetric | Metrics



A multi-agent system to design nanobodies for SARS-CoV-2 KP3

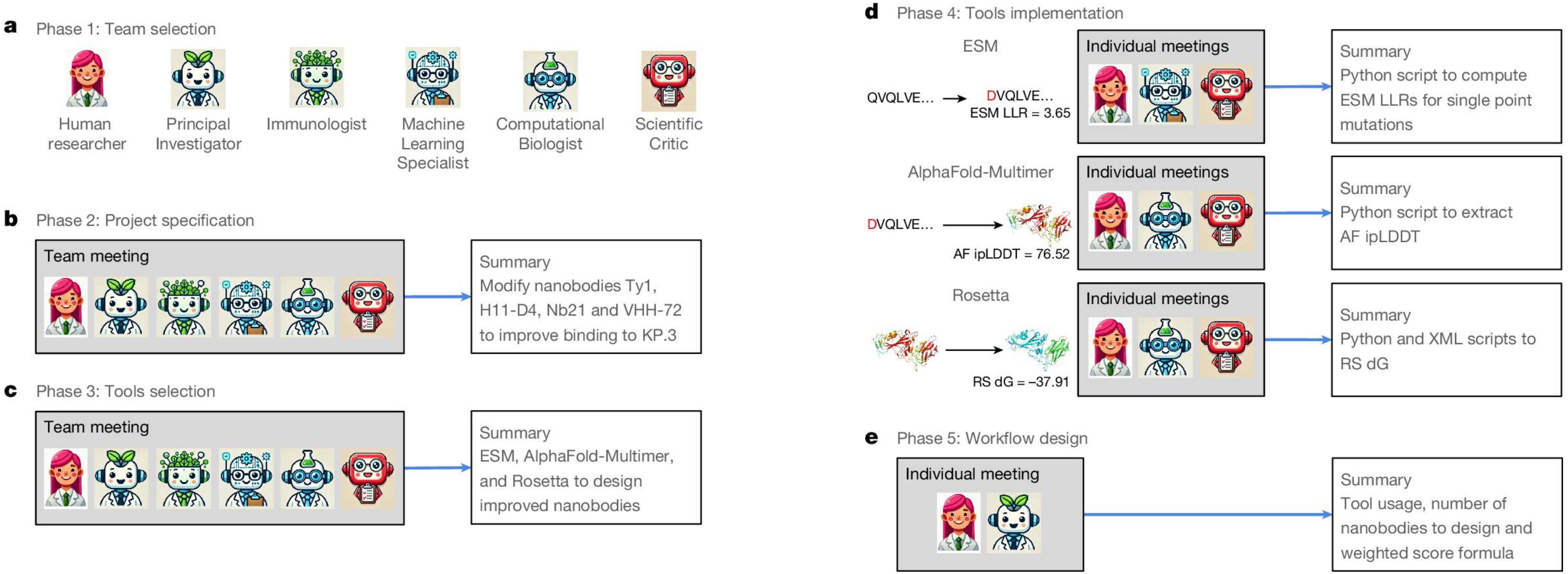
Did it succeed?

Did it do so autonomously?

GPT-4o agents

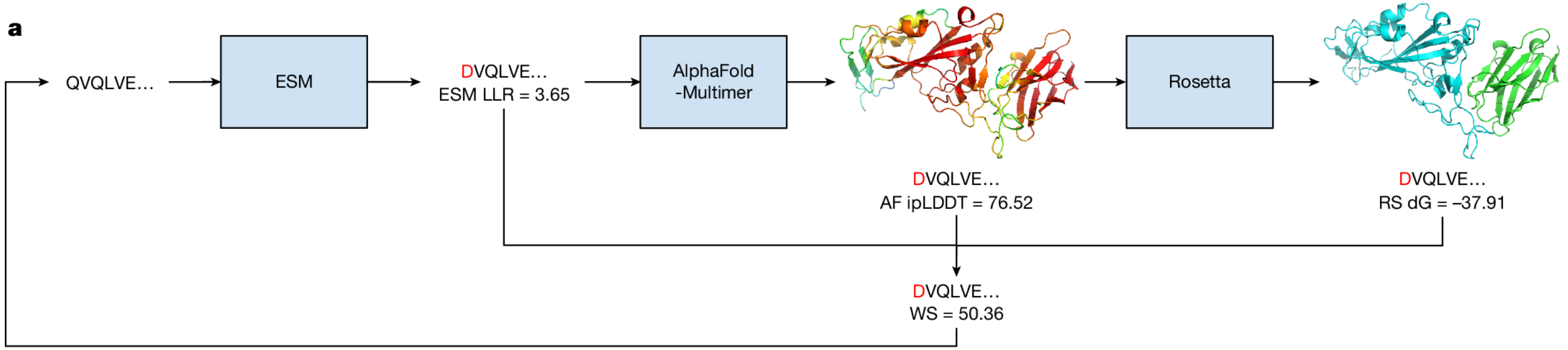
https://www.nature.com/articles/s41586-025-09442-9/

The Virtual Lab for nanobody design

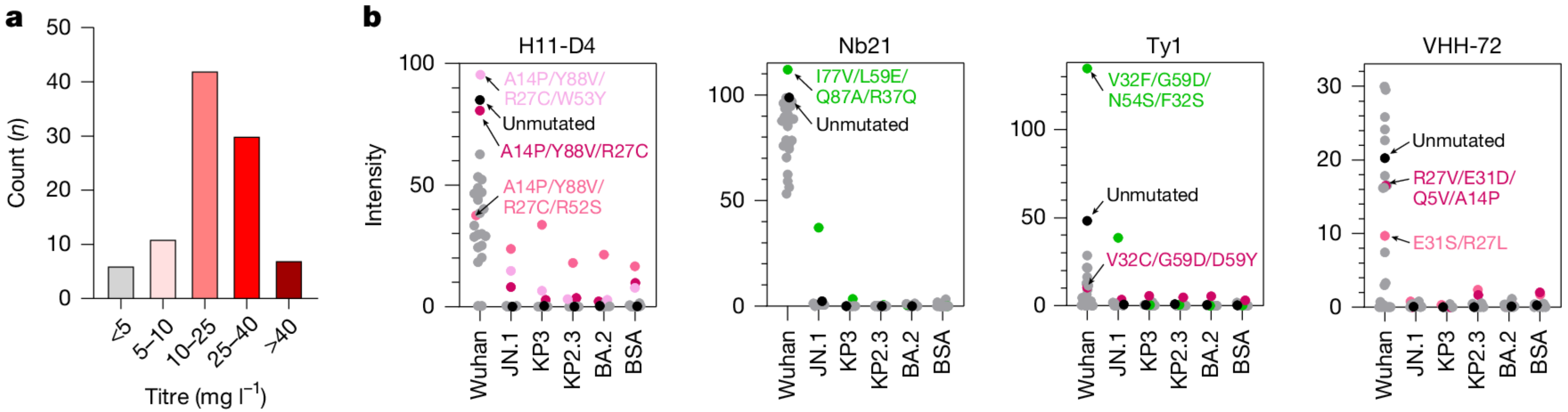


Production & validation of Virtual Lab nanobodies

Workflow suggested by Agents (fig 3)

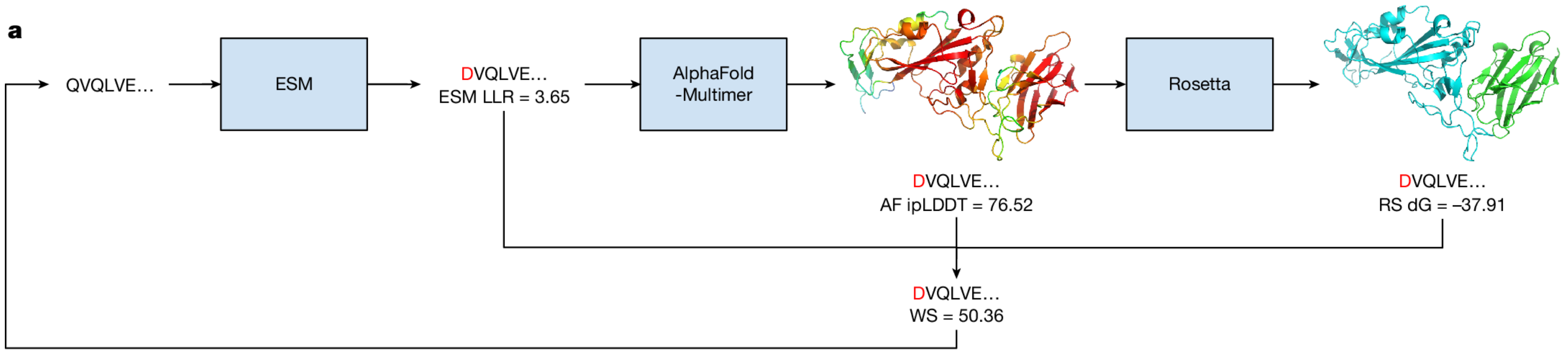


Experimental validation

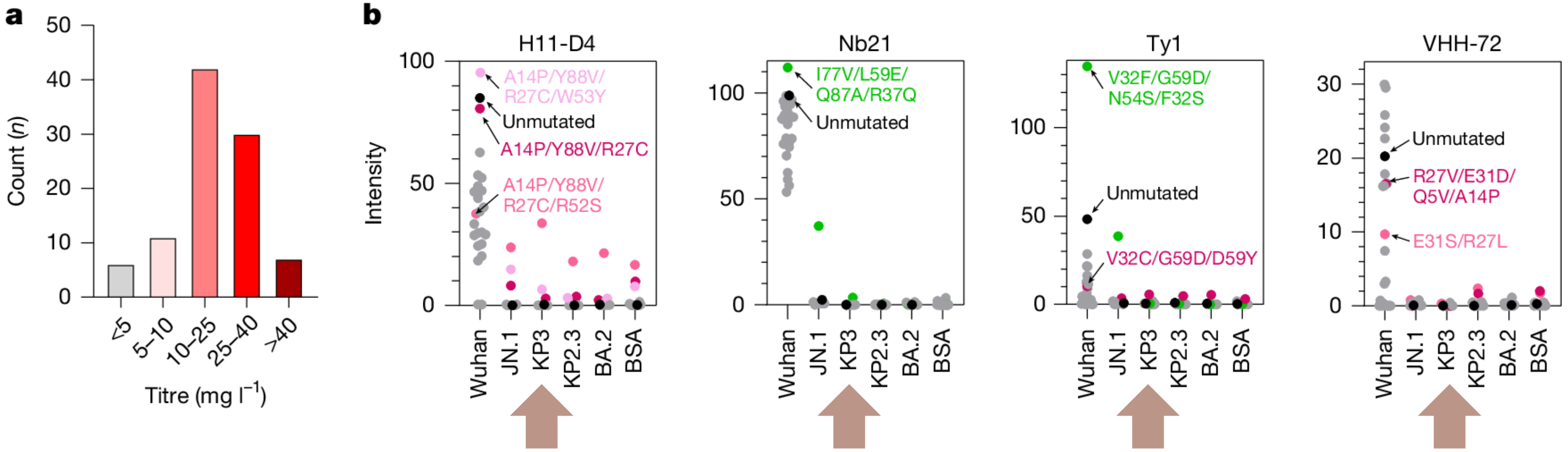


Production & validation of Virtual Lab nanobodies

Workflow suggested by Agents (fig 3)



Experimental validation



Target was SARS-CoV2-KP3, but nanobodies show increased binding only to JN.1 and Wuhan variants

Reading of the Virtual Lab

Autonomy	Low	Agents write scripts for tool calling, researcher debugs and runs them, writes standalone scripts for job scheduling, and runs experiments

Reading of the Virtual Lab

Autonomy	Low	Agents write scripts for tool calling, researcher debugs and runs them, writes standalone scripts for job scheduling, and runs experiments
Iterative optimization	No	Workflow is defined once and repeatedly executed

Reading of the Virtual Lab

Autonomy	Low	Agents write scripts for tool calling, researcher debugs and runs them, writes standalone scripts for job scheduling, and runs experiments
Iterative optimization	No	Workflow is defined once and repeatedly executed
Controls	No	No multi-agent vs single agent vs LLM vs scripted workflow with experimental validation; multi agent system produces better discussions…

Reading of the Virtual Lab

Autonomy	Low	Agents write scripts for tool calling, researcher debugs and runs them, writes standalone scripts for job scheduling, and runs experiments
Iterative optimization	No	Workflow is defined once and repeatedly executed
Controls	No	No multi-agent vs single agent vs LLM vs scripted workflow with experimental validation; multi agent system produces better discussions…
Robustness testing	No	No stress-testing, such as through prompt alterations, sequence alterations

Reading of the Virtual Lab

Autonomy	Low	Agents write scripts for tool calling, researcher debugs and runs them, writes standalone scripts for job scheduling, and runs experiments
Iterative optimization	No	Workflow is defined once and repeatedly executed
Controls	No	No multi-agent vs single agent vs LLM vs scripted workflow with experimental validation; multi agent system produces better discussions…
Robustness testing	No	No stress-testing, such as through prompt alterations, sequence alterations
Experimental validation	Yes	Result did not match goal; nanobodies designed for KP3 targeted JN.1 and Wuhan

Reading of the Virtual Lab

Autonomy	Low	Agents write scripts for tool calling, researcher debugs and runs them, writes standalone scripts for job scheduling, and runs experiments
Iterative optimization	No	Workflow is defined once and repeatedly executed
Controls	No	No multi-agent vs single agent vs LLM vs scripted workflow with experimental validation; multi agent system produces better discussions…
Robustness testing	No	No stress-testing, such as through prompt alterations, sequence alterations
Experimental validation	Yes	Result did not match goal; nanobodies designed for KP3 targeted JN.1 and Wuhan
Failure analysis	Limited	Limited, examples of coding failure fixed by critic

Reading of the Virtual Lab

Autonomy	Low	Agents write scripts for tool calling, researcher debugs and runs them, writes standalone scripts for job scheduling, and runs experiments
Iterative optimization	No	Workflow is defined once and repeatedly executed
Controls	No	No multi-agent vs single agent vs LLM vs scripted workflow with experimental validation; multi agent system produces better discussions…
Robustness testing	No	No stress-testing, such as through prompt alterations, sequence alterations
Experimental validation	Yes	Result did not match goal; nanobodies designed for KP3 targeted JN.1 and Wuhan
Failure analysis	Limited	Limited, examples of coding failure fixed by critic
Code/tool availability	Partial	Proprietary models like GPT-4 Turbo, Rosetta needs a license, prompts and meeting agendas are missing, unlikely to reproduce

Reading of the Virtual Lab

Autonomy	Low	Agents write scripts for tool calling, researcher debugs and runs them, writes standalone scripts for job scheduling, and runs experiments
Iterative optimization	No	Workflow is defined once and repeatedly executed
Controls	No	No multi-agent vs single agent vs LLM vs scripted workflow with experimental validation; multi agent system produces better discussions…
Robustness testing	No	No stress-testing, such as through prompt alterations, sequence alterations
Experimental validation	Yes	Result did not match goal; nanobodies designed for KP3 targeted JN.1 and Wuhan
Failure analysis	Limited	Limited, examples of coding failure fixed by critic
Code/tool availability	Partial	Proprietary models like GPT-4 Turbo, Rosetta needs a license, prompts and meeting agendas are missing, unlikely to reproduce
Benchmark suitability	n/a	No benchmarking in this paper, it's a proof of principle

Summary: The Virtual Lab is a planning and code-generation system, not an autonomous laboratory.
Execution, troubleshooting, and all physical work remain human-driven.

Reading of the Virtual Lab

Virtual Lab is an impressive agentic planning demonstration. It shows that LLM agents can design complex computational workflows and write runnable code.

However, the system is not autonomous, executes no code on its own, lacks robustness testing, and depends heavily on human oversight. Iteration is not adaptive; controls do not test agent utility against simpler baselines; failure analysis is shallow; code is incompletely available; the biological result (KP.3 binding) did not meet the stated objective.

It is a good proof-of-concept for agent-assisted workflow generation.

Things to look out for when reading agent papers

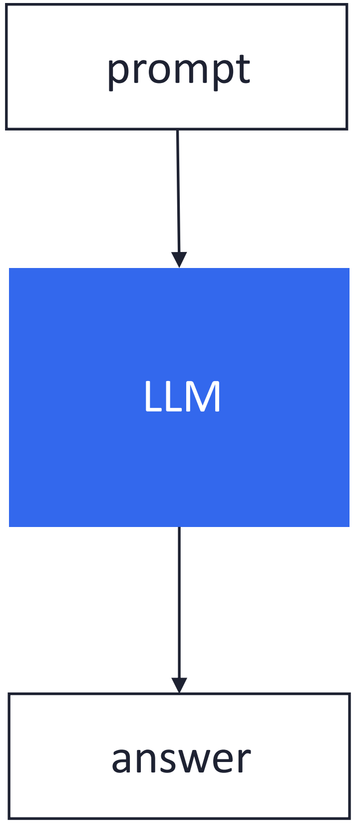
Autonomy	What part of the workflow is executed by humans
Iterative optimization	Whether the agent iteratively optimizes or simply executes a single workflow
Controls	Comparing multi agent *vs* single agent *vs* LLM *vs* scripted workflow
Robustness testing	Perturbations like changing prompts, data, LLMs, agent removal and changing roles
Experimental validation	When applicable
Failure analysis	Planning, communication, memory, tools
Code/tool availability	Is the agent reproducible/testable? Does it use proprietary models?
Benchmark suitability	If LLMs perform well it may really be testing intrinsic knowledge; data leakage may be an issue

LLM Benchmarks

$\neq$

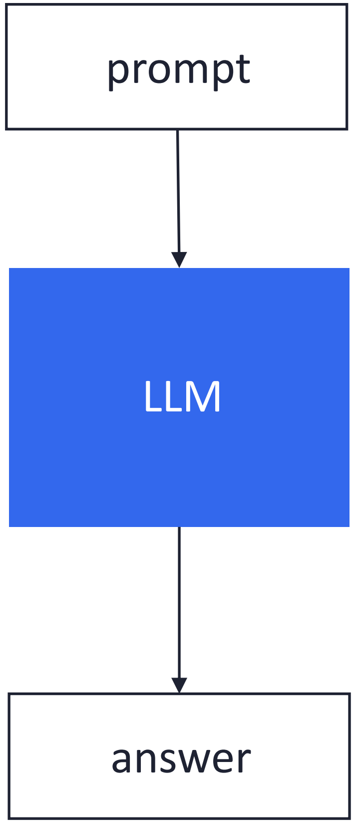
Agent Benchmarks

LLM benchmarks test intrinsic knowledge



Simple QA benchmarks
test intrinsic knowledge

LLM benchmarks test intrinsic knowledge



Simple QA benchmarks
test intrinsic knowledge

e.g. Massive Multitask Language Understanding (MMLU) QA

Subject: Biology

Q: What is the origin of the hyoid bone?

A: First pharyngeal arch

B: First and second pharyngeal arches

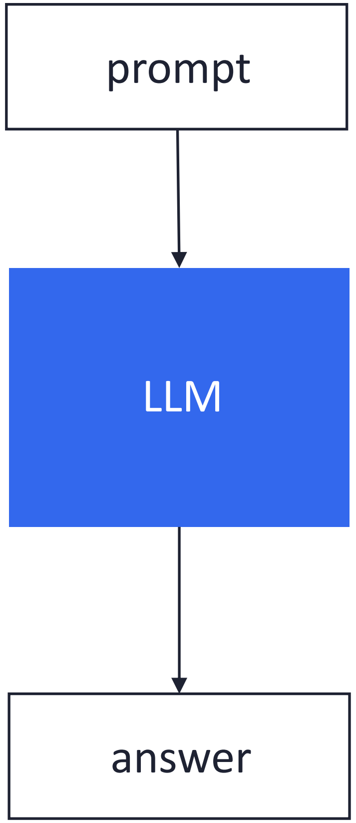
C: Second pharyngeal arch

D: Second and third pharyngeal arches

Model Output: D

Evaluation: Correct

LLM benchmarks test intrinsic knowledge



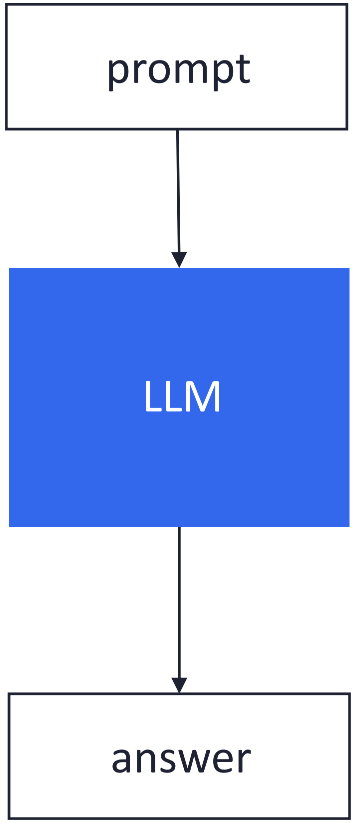
Simple QA benchmarks
test intrinsic knowledge

e.g. Massive Multitask Language Understanding (MMLU) QA

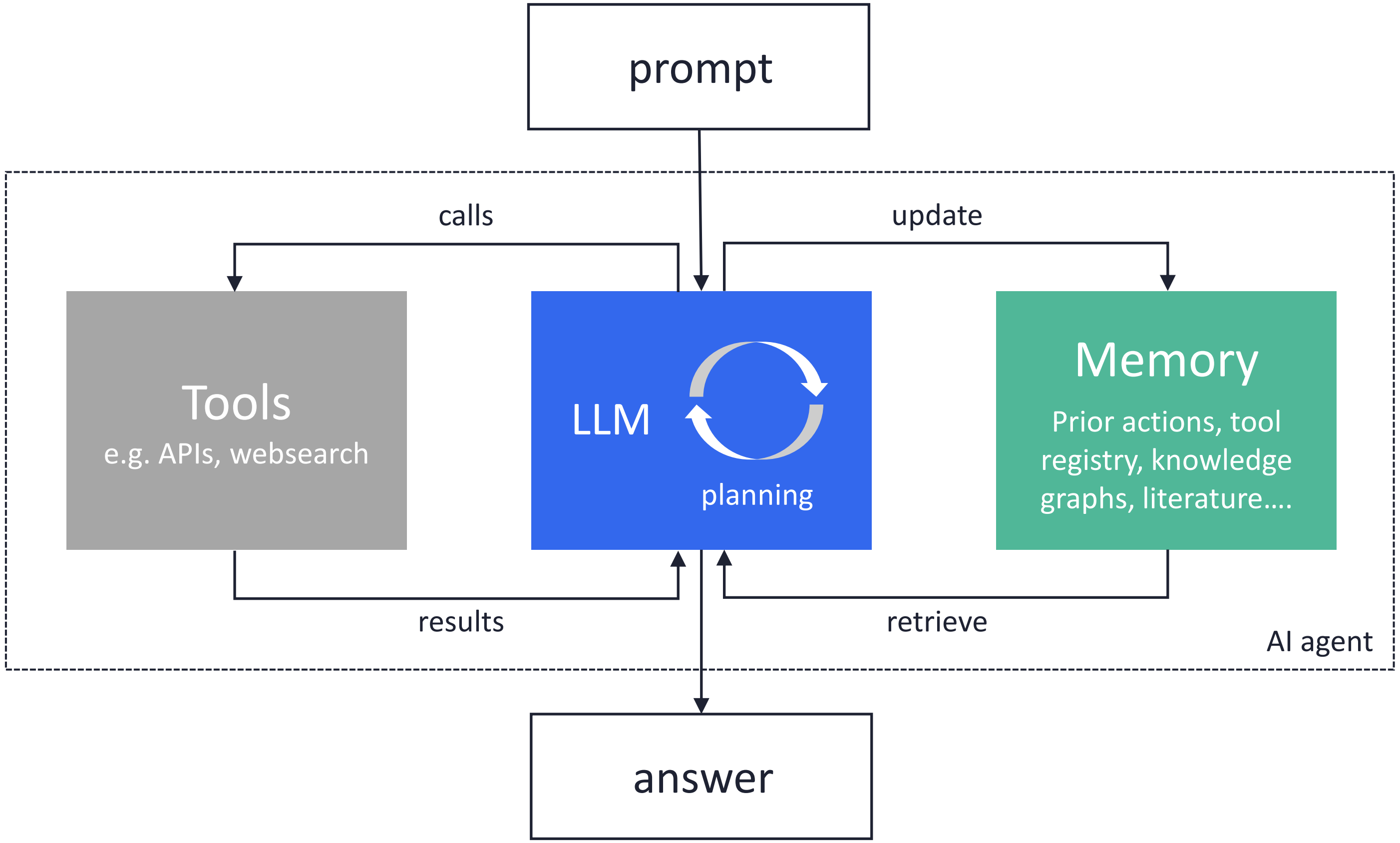
Model	MMLU All Subjects - EM
Claude 3.5 Sonnet (20241022)	**0.873**
DeepSeek v3	0.872
Gemini 1.5 Pro (002)	0.869
Claude 3.5 Sonnet (20240620)	0.865
Claude 3 Opus (20240229)	0.846
Llama 3.1 Instruct Turbo (405B)	0.845
GPT-4o (2024-08-06)	0.843
GPT-4o (2024-05-13)	0.842
Qwen2.5 Instruct Turbo (72B)	0.834
Gemini 1.5 Pro (001)	0.827

https://crfm.stanford.edu/helm/mmlu/latest/

Agents need different types of benchmarks



Simple QA benchmarks
test intrinsic knowledge



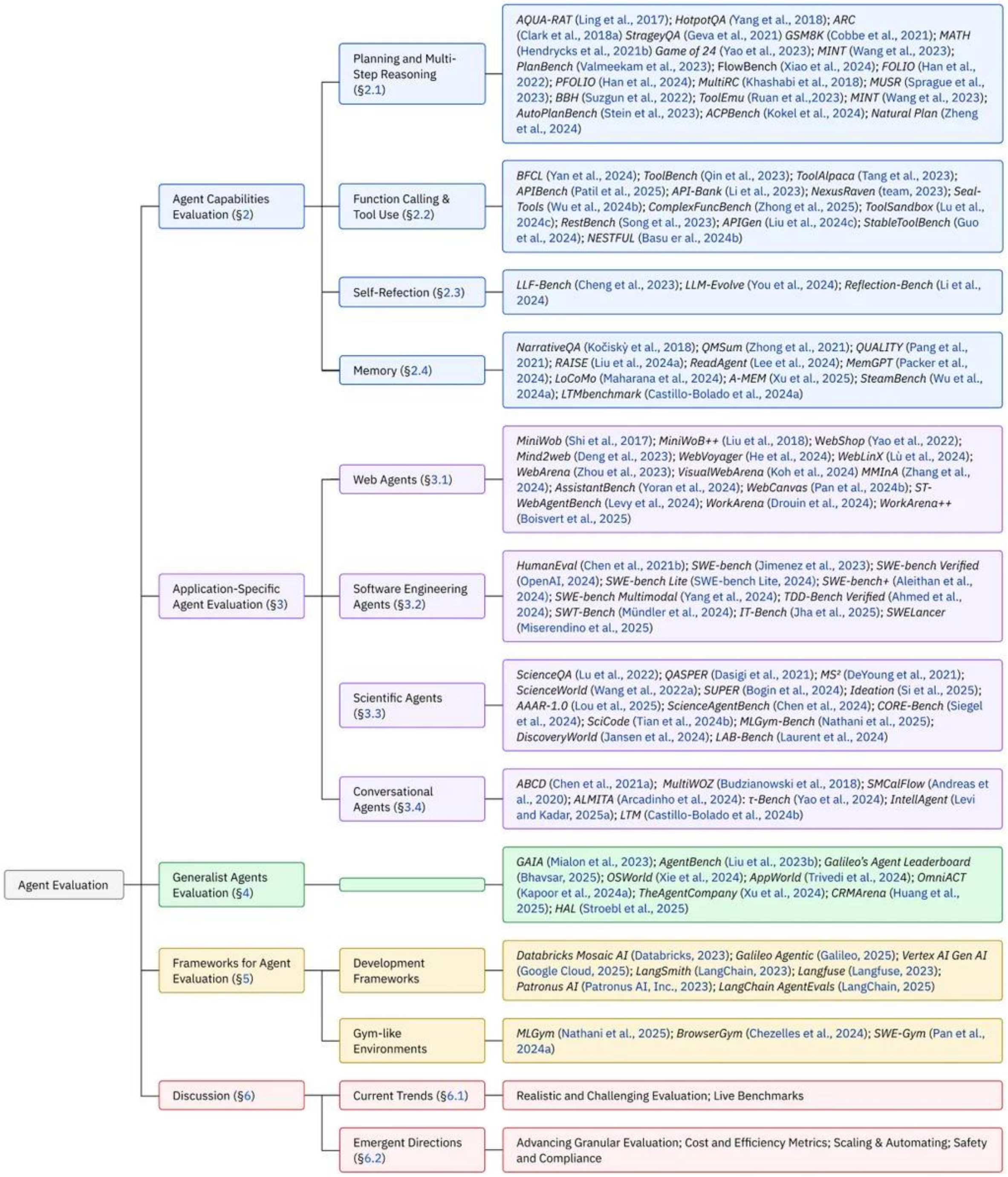
Process centric benchmarks test planning (“reasoning traces”), communication, tool use, and computational efficiency

Benchmarks for AI agents and Agentic AI

A recent review by IBM on benchmarking agents provides an overview of some of the main efforts

https://research.ibm.com/blog/AI-agent-benchmarks

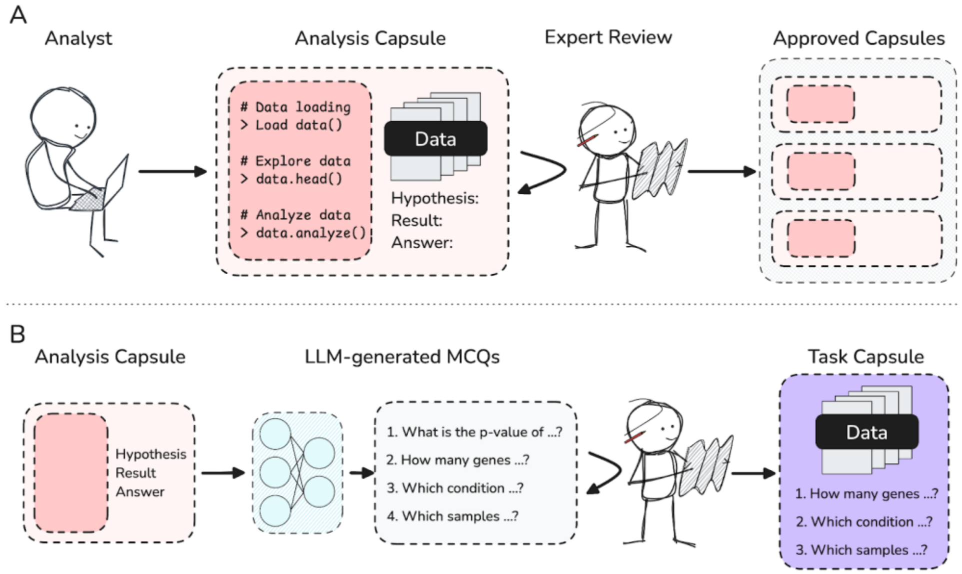
Let's look at 2 not listed…



Selected Benchmarks for AI agents and Agentic AI

Benchmark	Comments	Reference	Developers
BixBench	61 analytical scenarios in bioinformatics with 205 open ended questions and multiple choice questions	Mitchener *et al.*, (2025) arXiv:2503.00096 and www.futurehouse.org/research-announcements/bixbench	**FutureHouse**, a non-profit research lab, base San Francisco. Mission: "build semi-autonomous AIs for scientific research"
AstaBench	2,400 problems where AI can help human scientists	Bragg *et al.*, (2025) arXiv:2510.21652 and allenai.org/blog/astabench	**Allen Institute for AI (AI2)**, a U.S. nonprofit research institute founded by Microsoft co-founder Paul Allen. Mission: "Building breakthrough AI to solve the world's biggest problems"

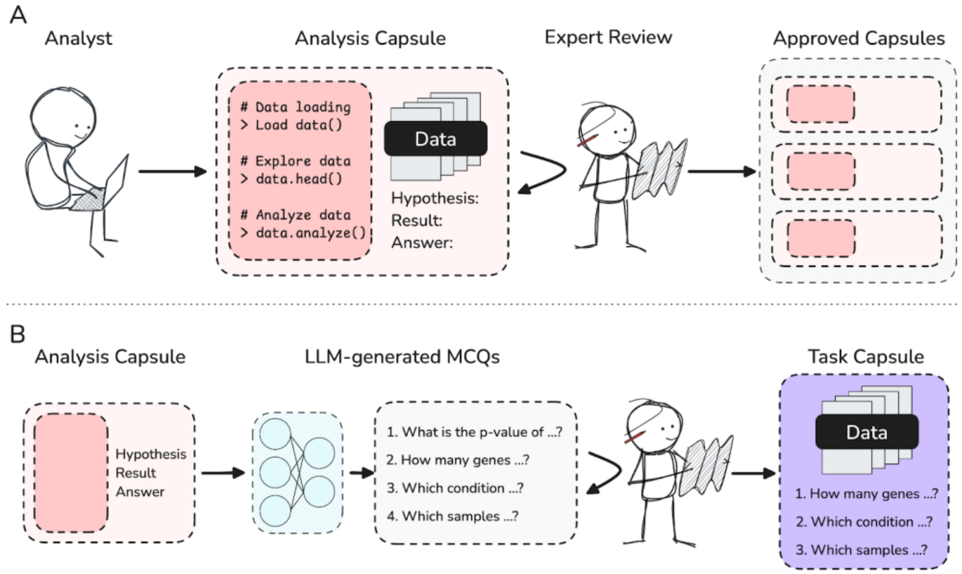
Benchmarking AI Agents on BixBench

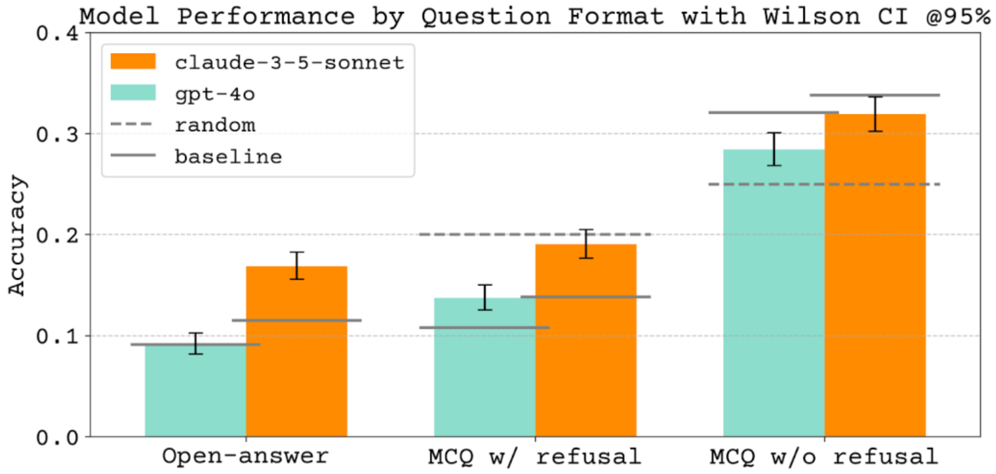


Experts create “analysis capsules” (in Jupyter).
LLMs generate questions for them (open and MCQs).
Agents are given the data and hypothesis and must perform the analyses to answer the linked questions.

Mitchener *et al.*, (2025) arXiv:2503.00096 and https://www.futurehouse.org/research-announcements/bixbench

Benchmarking AI Agents on BixBench





Experts create “analysis capsules” (in Jupyter).
LLMs generate questions for them (open and MCQs).
Agents are given the data and hypothesis and must perform the analyses to answer the linked questions.

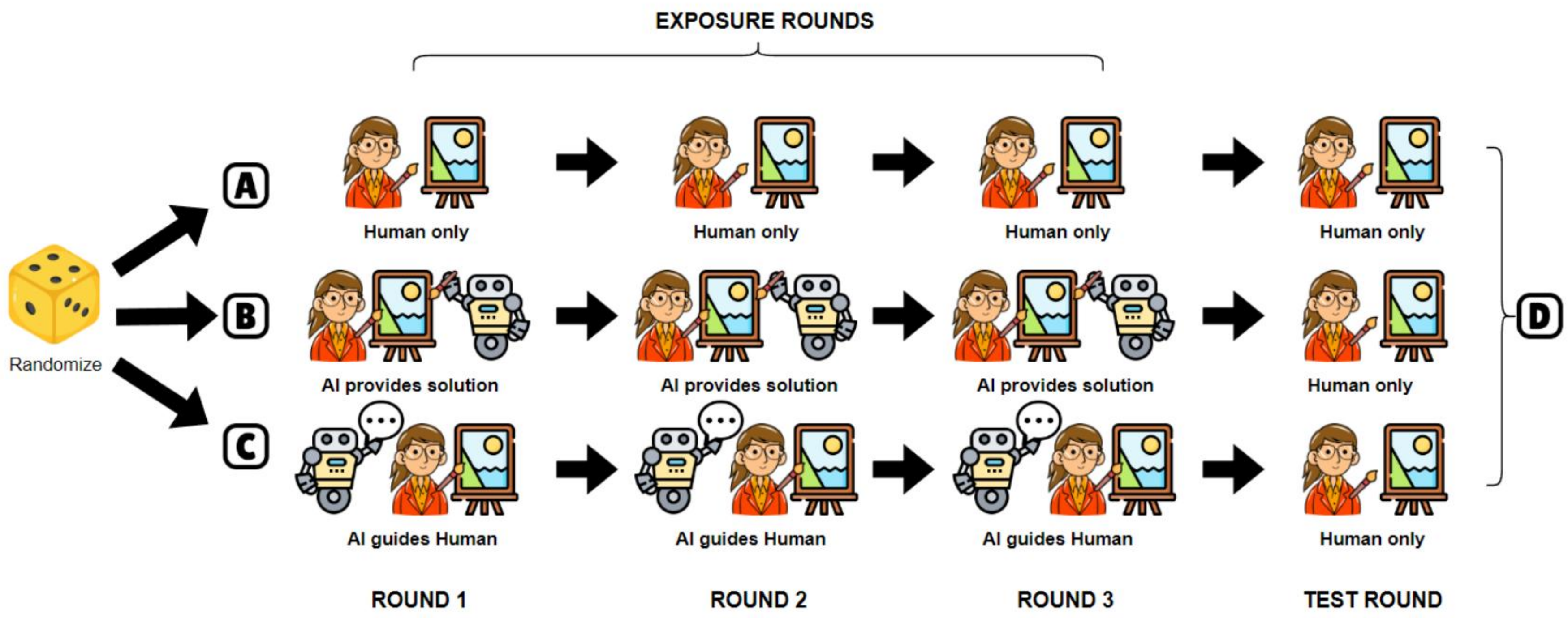
Agent performance is low for open answers and close to random for MCQs. Baseline is “pure recall” (no analysis). When allowed to refuse (center) the models often do so, particularly for complex analyses.

Mitchener *et al.*, (2025) arXiv:2503.00096 and https://www.futurehouse.org/research-announcements/bixbench

BixBench reveals the limitations of current LLMs in agentic bioinformatics.
Do other benchmarking studies say the same?

Delving into agents

LLM usage may impact human creativity



Teams using LLMs get a short-term boost followed by a longer-term reduction in creativity

From: Kumar *et al*., 2025, *Human Creativity in the Age of LLMs*, arXiv:2410.03703

Think about what the effects of widespread use of Agentic AI might be on scientists

LLMs may be increase the rate of false discoveries

- One unexpected consequence of the drive to create “AI-ready datasets” is the emergence of AI papermills
- PLOS journals now systematically reject papers based on the NHANES dataset without further validation
- Agentic AI and AI labs could amplify this tendency

META-RESEARCH ARTICLE

Explosion of formulaic research articles, including inappropriate study designs and false discoveries, based on the NHANES US national health database

Tulsi Suchak[1], Anietie E. Aliu[1], Charlie Harrison[2], Reyer Zwiggelaar[2], Nophar Geifman[1], Matt Spick[1]*

1 School of Health Sciences, Faculty of Health and Medical Sciences, University of Surrey, Guildford, United Kingdom, **2** Department of Computer Science, Aberystwyth University, Ceredigion, United Kingdom

PLoS Biol 23(5): e3003152

HOME > NEWS > SCIENCEINSIDER > LOW-QUALITY PAPERS ARE SURGING BY EXPLOITING PUBLIC DATA SETS AN...

SCIENCEINSIDER | SCIENTIFIC COMMUNITY

Low-quality papers are surging by exploiting public data sets and AI

Paper mills are also likely contributing to “false discoveries”

14 MAY 2025 · 4:15 PM ET · BY CATHLEEN O'GRADY

Science, Vol 388, Issue 6749

“Dual use” of GenAI may create new threats

- Researchers repurposed a generative model using public toxicity data to generate 40,000 toxic molecules
- It recreated VX (banned since 1993) & many other known chemical warfare agents.
- Low alignment of Agentic AI labs could lead to creation of new pathogens or toxins, and other problems such as the release of medical data…

Dual use of artificial-intelligence-powered drug discovery

An international security conference explored how artificial intelligence (AI) technologies for drug discovery could be misused for de novo design of biochemical weapons. A thought experiment evolved into a computational proof.

Fabio Urbina, Filippa Lentzos, Cédric Invernizzi and Sean Ekins

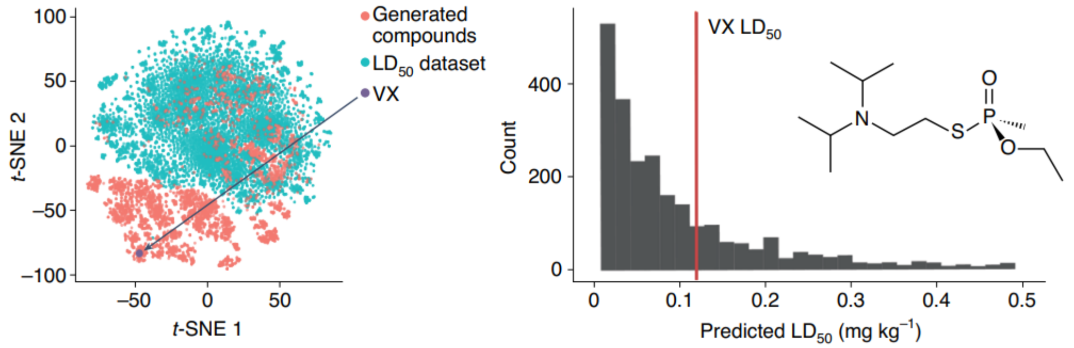


Fig. 1 | A *t*-SNE plot visualization of the LD_{50} dataset and top 2,000 MegaSyn AI-generated and predicted toxic molecules illustrating VX. Many of the molecules generated are predicted to be more toxic in vivo in the animal model than VX (histogram at right shows cut-off for VX LD_{50}). The 2D chemical structure of VX is shown on the right.

Urbina, F., Lentzos, F., Invernizzi, C. et al. Dual use of artificial-intelligence-powered drug discovery. *Nat Mach Intell* **4**, 189–191 (2022).

We are done!

LLM basics

AI agent basics: tools and reasoning

AI Agents in biology

AI Agent evaluation

AI agent Safety & ethics

Themes to explore in this literature review

- Definitions of AI agents and Agentic AI, and relation to LLMs
- Agentic AI in biology and/or bioinformatics
- Evaluation and benchmarking
- Limitations of current agents (e.g. in reasoning)
- Ethical considerations and risks

These slides and a starting bibliography are attached to
https://lab.dessimoz.org/teaching/rqb/schedule_2025