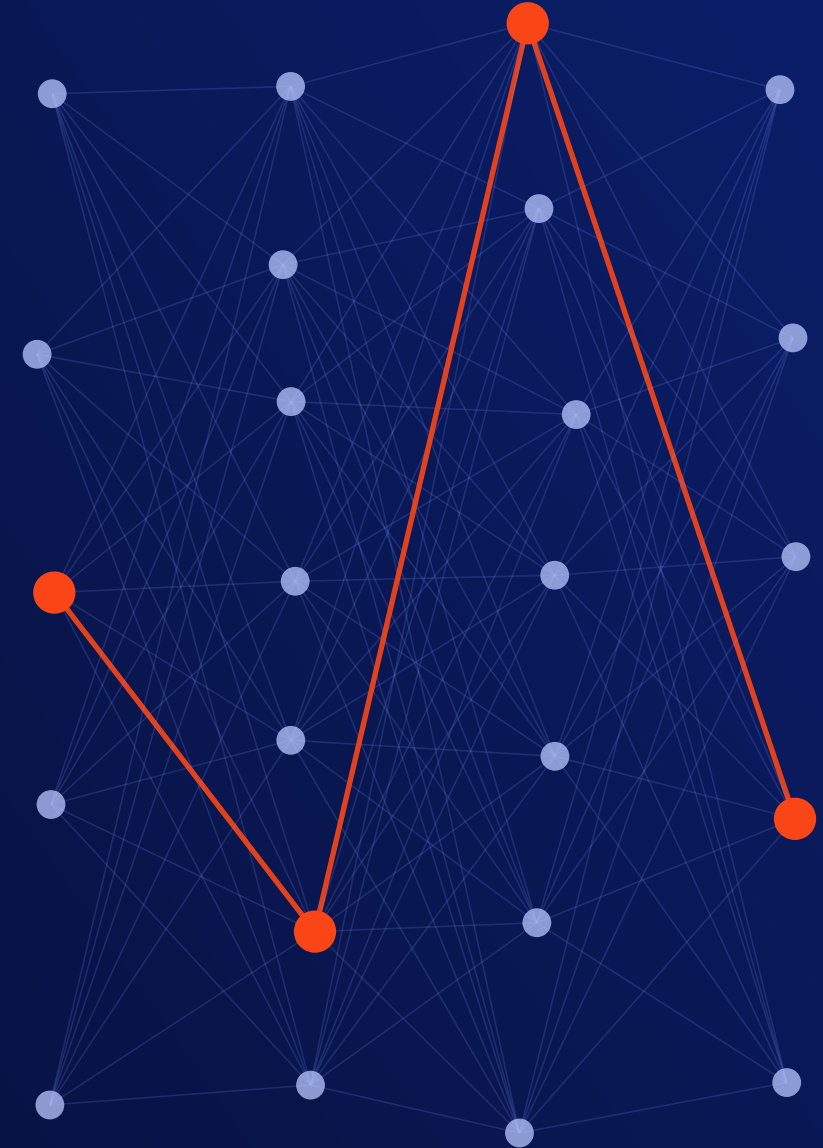

LECTURE 9-1

Retrieval, Tool Use, and Agentic Systems

Models that look things up and act



Parisa Rashidi · Herbert Wertheim College of Engineering, University of Florida



Where We Are

Weeks 6 to 8 built models of biomedical data. Today: models that look things up and act.

Week 8

EHR and time series

Patient timelines and signals as tokens

Week 9: today

Retrieval, tools, and agents

Models that look things up and act

Week 11

Generative models and AI for science

Diffusion for protein design

Three Things a Trained Model Cannot Do

Each limit has a fix outside the weights.

Frozen knowledge

trained up to a cutoff date; guidelines change after it

→ retrieval

No local data

never saw this hospital's protocols or this patient's record

→ retrieval and tools

Cannot act

writes text; cannot query a record or run a calculation

→ tools and agents



Retrieval-Augmented Generation

The Model Alone

A reasonable national answer, but not this hospital's rule.

"Within how many hours must antibiotics start under our sepsis protocol?"

Model alone

"Within 3 hours."

a reasonable national figure

no source to check

Our hospital's protocol

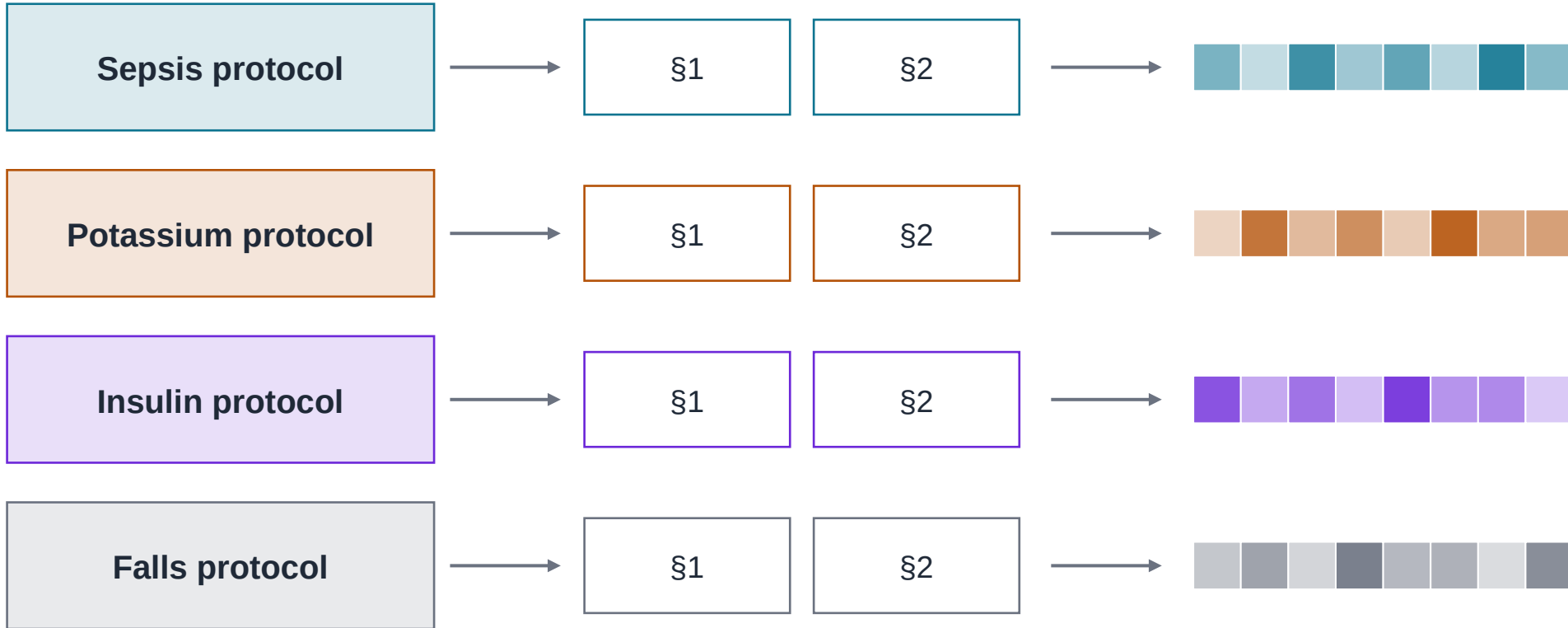
Sepsis protocol, section 2:

"start antibiotics within 1 hour"

synthetic document

Step 1: Chunk and Embed Once

Split the documents into chunks; turn each chunk into a vector, ahead of time.



Almanac: 1,000-token chunks, 1,536 numbers each

one vector per chunk, stored

Step 2: Score Every Chunk

Embed the question the same way; compare by cosine similarity.

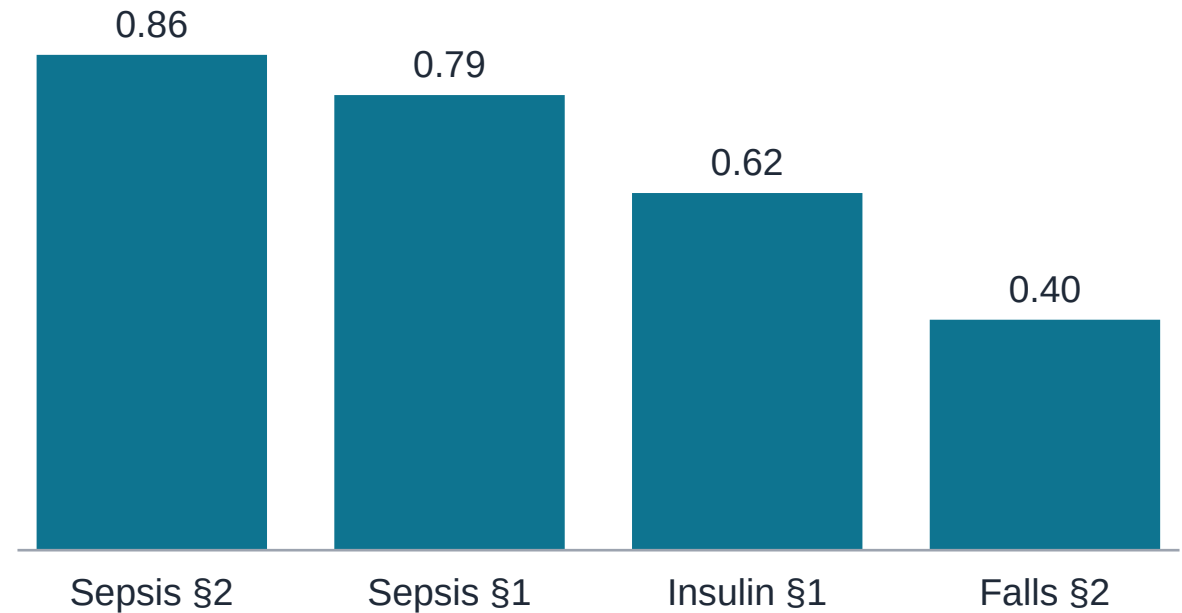
$$\cos(\mathbf{q}, \mathbf{c}) = \mathbf{q} \cdot \mathbf{c} / (\|\mathbf{q}\| \|\mathbf{c}\|)$$

$\mathbf{q}$ = the question vector

$\mathbf{c}$ = a chunk vector

vectors of length 1:

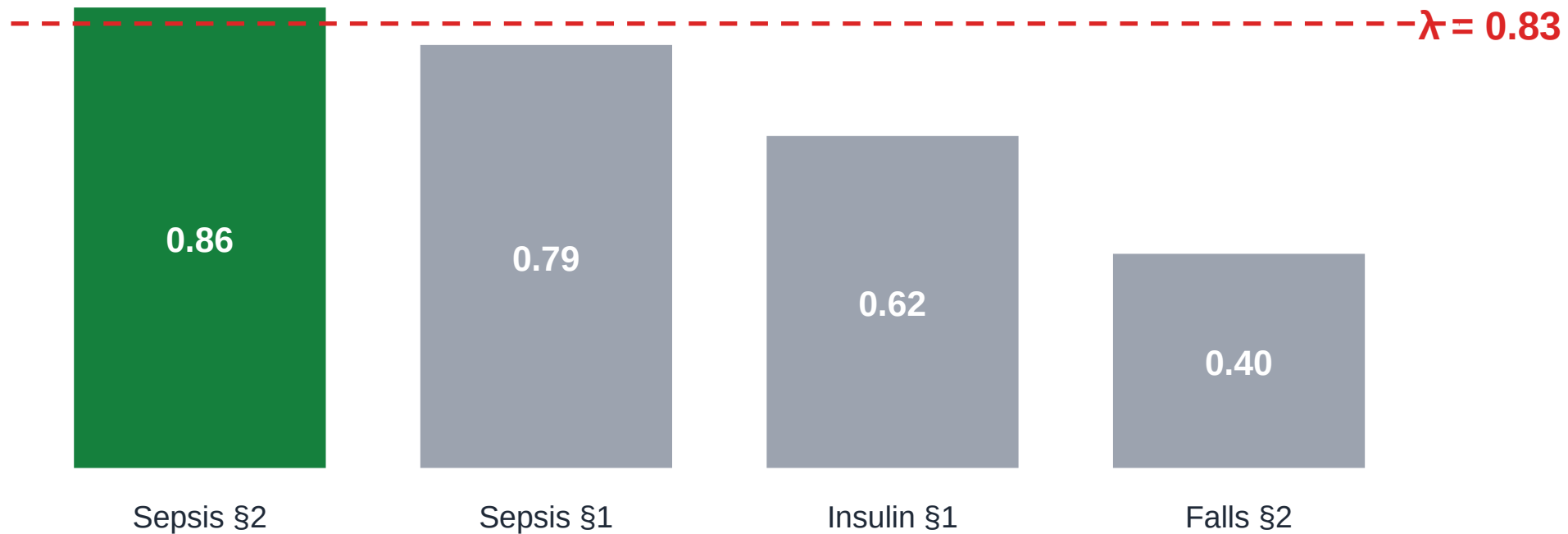
just the dot product



scores illustrative, as in clip 1

Step 3: Only Chunks Above a Threshold

Almanac passed only chunks scoring at least $\lambda = 0.83$.

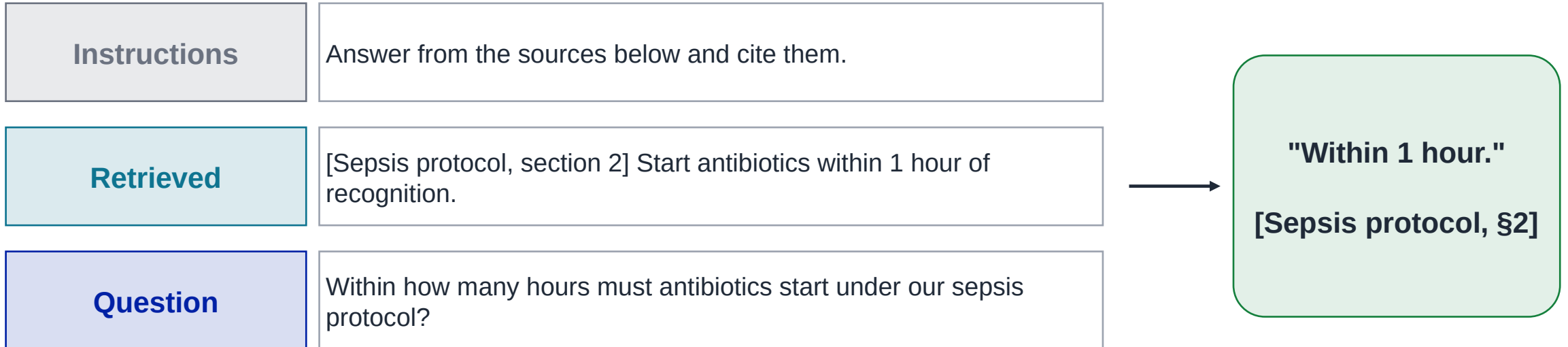


passes: Sepsis §2 only

Step 4: A Grounded Answer with a Citation

The passing chunk goes into the prompt with its source.

the prompt



When Nothing Passes: Two Designs

A question the sources cannot answer: its best match scores only 0.71.

Abstain (Almanac preprint)

"Not found in the sources."

- + never answers without a source
- less useful; the user must look elsewhere

Fall back (published Almanac)

answers from the model's own knowledge

- + answers more questions
- no source to check; should the user be told?

Animation: Retrieval-Augmented Generation

Animation clip: week09-clip1.mp4

Silent version embedded (click to play). The narrated version is in the course repository.

Almanac: What the Published Study Found

Ranked first of four systems on 314 clinical questions.

	Almanac	ChatGPT-4	Bing	Bard
Answer model	GPT-4 + retrieval	GPT-4		
Factuality (mean rank)	1.96 (best)	2.34		
Adversarial prompts resisted	100%	7%	70%	77%

314 questions, 9 specialties, 10 clinician graders; 25 questions public.

Almanac also ranked first for completeness and for preference.

Preprint versus Publication

The same system name, two different studies.

	Preprint (2023)	Published (NEJM AI 2024)
Answer model	text-davinci-003	GPT-4
Questions	130, from 5 physicians	314, 9 specialties, 10 graders
Compared with	ChatGPT	ChatGPT-4, Bing, Bard
Adversarial	95% vs 0%	100% vs 7% (ChatGPT-4)
Preference	ChatGPT preferred 57%	Almanac ranked first
Nothing passes	abstains	falls back on the model

Reading Almanac Critically

Three questions to bring to Thursday.

Refusal

The 100% adversarial score comes mostly from declining to answer.

Ranks

A mean rank says who was ahead, not by how much.

Benchmark

Written and graded by the same clinicians; 25 of 314 questions public.

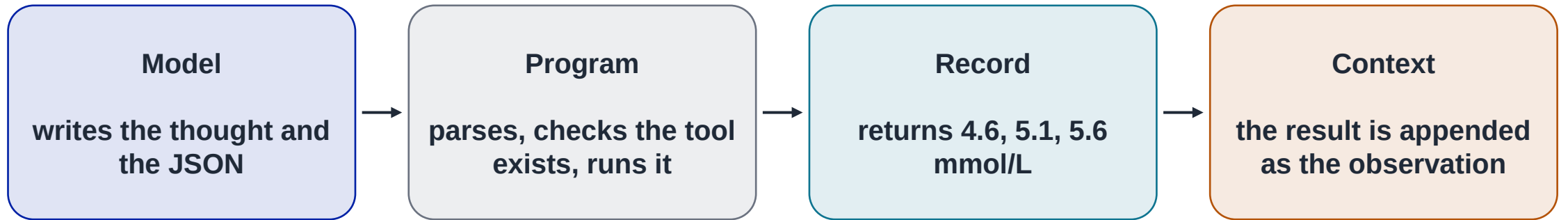


Reason, Act, Observe

A Tool Call Is Just Text

The model writes JSON in an agreed format; the program does the rest.

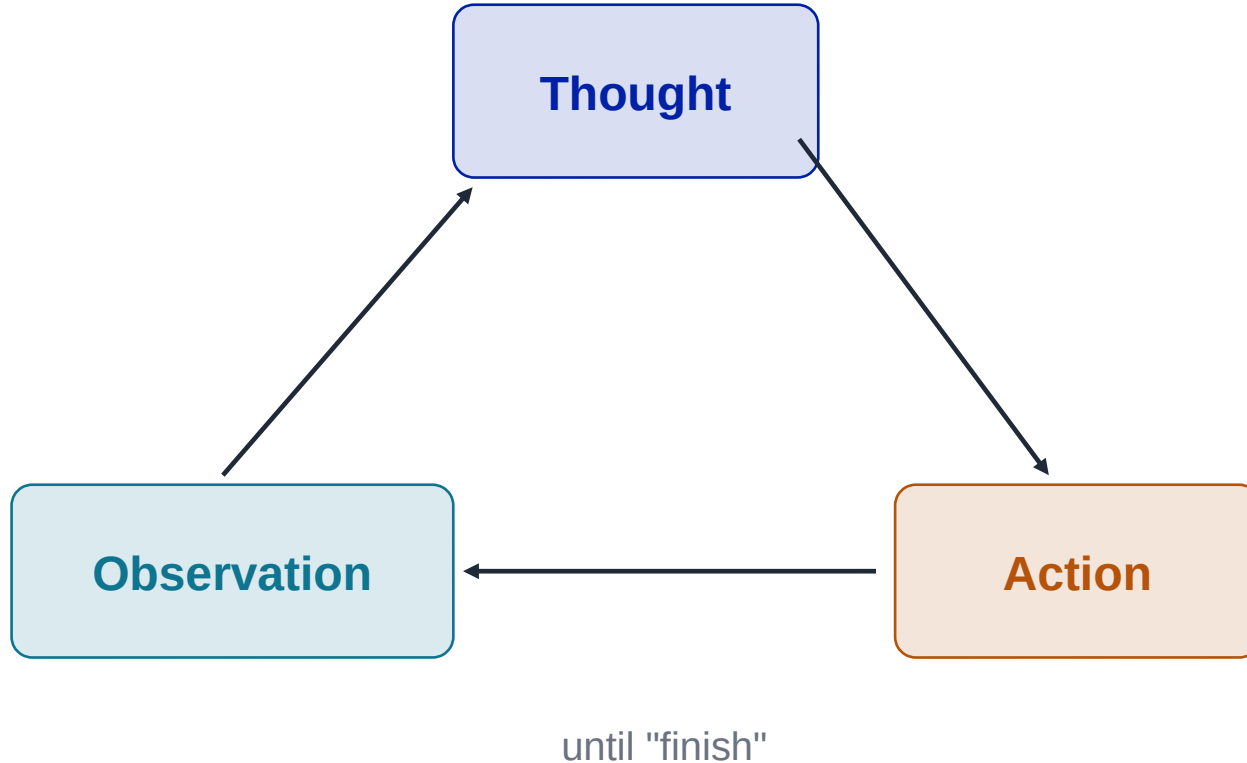
```
Action: {"tool": "get_labs", "args": {"patient_id": "P-001", "test": "potassium", "last": 3}}
```



The program, not the model, runs the tool. So the program decides which tools exist.

The ReAct Loop

Thought, action, observation, repeated until the model writes finish.



Our task (synthetic)

Is P-001's potassium rising,
and is it abnormal?

tools: get_labs, get_meds
(both read-only)

One Run, Step by Step

Two thoughts, two actions, one observation.

Thought

I need the last three potassium results.

Action

```
{"tool": "get_labs", "args": {"patient_id": "P-001", "test": "potassium", "last": 3}}
```

Observation

4.6, 5.1, 5.6 mmol/L

Thought

Rising; 5.6 is above the usual upper limit of about 5.0.

Action

finish: "Yes, rising (4.6, 5.1, 5.6 mmol/L) and above the upper limit."

Prompted Then, Post-Trained Now

ReAct used a few worked examples; today's models are trained to call tools.

ReAct (2022 to 2023)

PaLM 540B, few-shot prompting
worked examples in the prompt
no training
free-text actions parsed by the program

Today

post-trained to call tools (Week 5)
tools passed as JSON schemas
structured output the API checks
same loop: model writes, program runs

Animation: Reason, Act, Observe

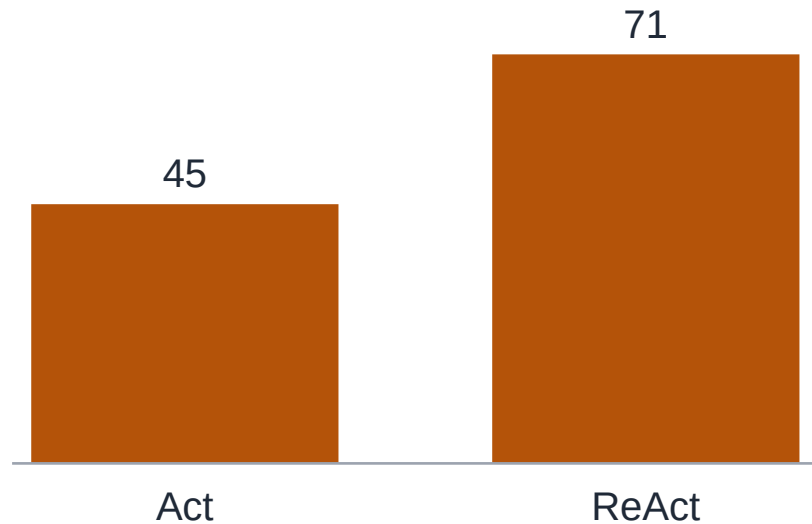
Animation clip: week09-clip2.mp4

Silent version embedded (click to play). The narrated version is in the course repository.

ReAct Results

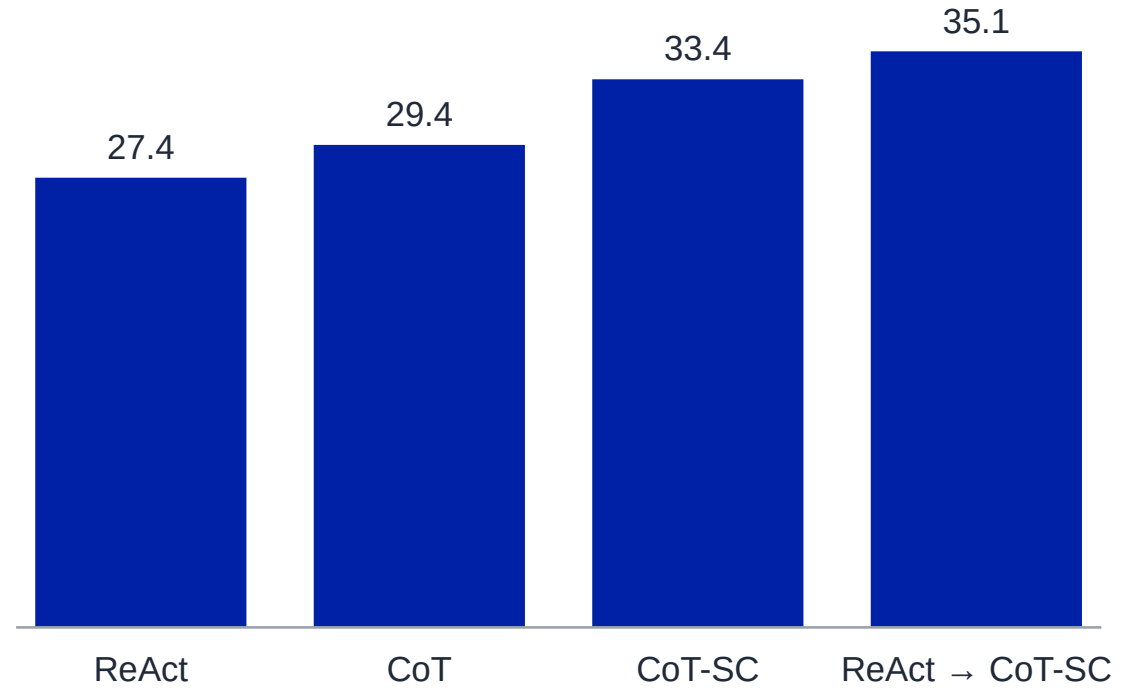
Actions help most when the task is acting; on question answering, the hybrid wins.

ALFWorld success, best of 6 prompts (%)



worst ReAct prompt: 48%

HotpotQA exact match



How ReAct Fails

Grounding removed hallucination; reasoning errors took its place.

Chain of thought failures

56% hallucination

facts come from the weights

ReAct failures

0% hallucination

47% reasoning errors (incl. loops)

23% unhelpful search results

facts come from the tool

In the Notebook: A Small Model as the Agent

Qwen3-0.6B on eight tasks: the format was easy, the judgment was not.

	Calls parsed	Tasks finished	Answers right
Model alone	22 of 22	5 of 8	3 of 8
With repeat guard	19 of 19	8 of 8	6 of 8

A well-formed tool call says nothing about whether the reasoning after it is right.

Remaining errors: 5.6 mmol/L called normal; a "No" with the right reason.



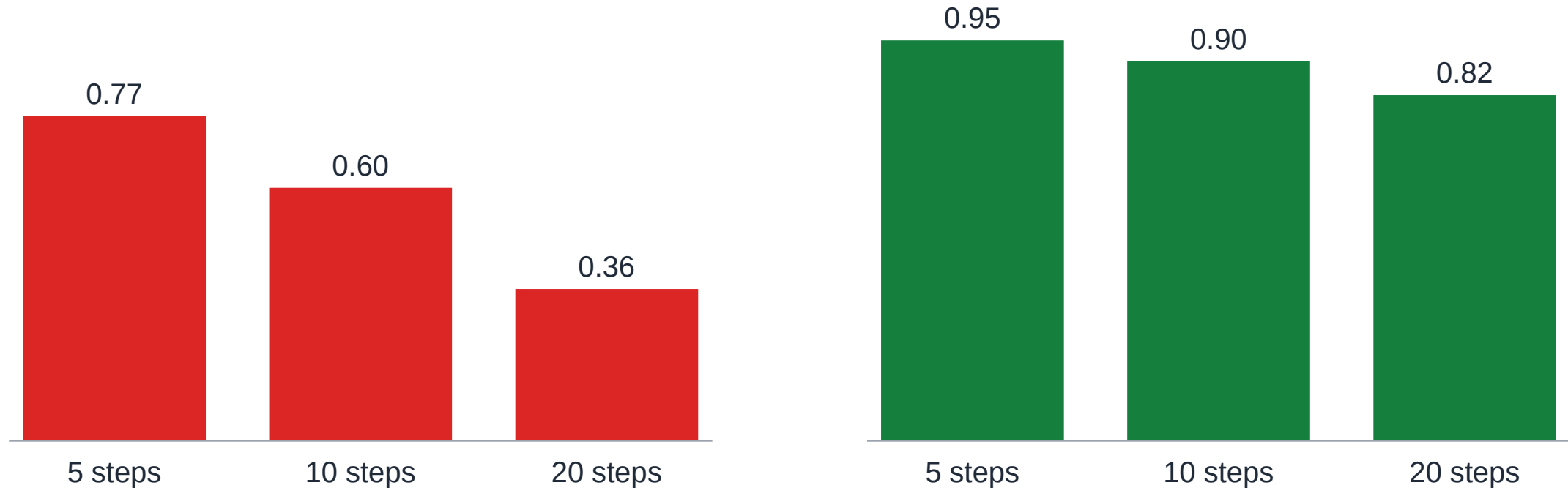
Errors Compound Over Steps

Small Per-Step Errors Add Up

If each step succeeds with probability p , n steps succeed with p^n .

$p = 0.95$ per step

$p = 0.99$ per step



Animation: Errors Compound Over Steps

Animation clip: week09-clip3.mp4

Silent version embedded (click to play). The narrated version is in the course repository.

Limiting the Damage

Most protections live in the program around the model, not in the model.

Check results

compare answers with the data and reference ranges

Read-only tools

get_labs and get_meds; nothing that orders or sends

Stop loops

refuse repeated identical calls (notebook: 3 loops fixed)

Clinician checkpoint

any action that changes care waits for approval

Retrieval, Long Context, or Fine-Tuning?

Three ways to give a model knowledge it was not trained on.

	Retrieval	Long context	Fine-tuning
Knowledge updates	edit the documents	paste new text	retrain
Cites a source	yes	possible	no
Main risk	misses the passage	cost per question	stale or invented facts
Best for	changing local facts	one long record	format and behavior

Recap: Look Things Up, Then Act

Retrieve

Chunk, embed, score, threshold, cite. Almanac: first of four on 314 questions.

Read critically

Adversarial wins by refusing; ranks hide effect sizes; preprint $\neq$ publication.

Act

A tool call is JSON the program runs. ReAct: 71% vs 45% on ALFWorld.

Compound

0.95 per step $\rightarrow$ 0.36 over 20 steps. Check, limit, approve.

Before Thursday

Read

Yao et al., ReAct (ICLR 2023) and Zakka et al., Almanac (NEJM AI 2024; the published version, not the preprint).

Write

Reading responses on Canvas before 1:00 PM Thursday.

Explore

Week 9 reference notebook (ungraded): retrieval with a threshold, a ReAct agent with Qwen3-0.6B, error compounding.

After the break

Week 11: generative models and AI for science. Progress report due Fri Mar 26, 11:59 PM.