AI Agents for Biomedical Knowledge Work

What changes when the AI can act on your files

SEAN DAVIS, MD, PHD · 2026-07-28

You already know the hard part

You use chatbots. You write prompts. You judge whether the answer is any good.

That intuition transfers directly. This session adds three things:

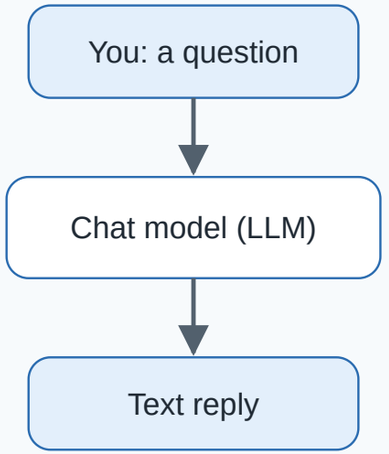
- what changes when the AI can **act in your files**, not just chat
- what is actually happening under the hood — **context** and **tokens**
- a few habits that make these tools **safer, cheaper, and sharper**

Part 1 — From chatbot to agent

Two ways to use the same model

Chatbot

answers from a prompt window

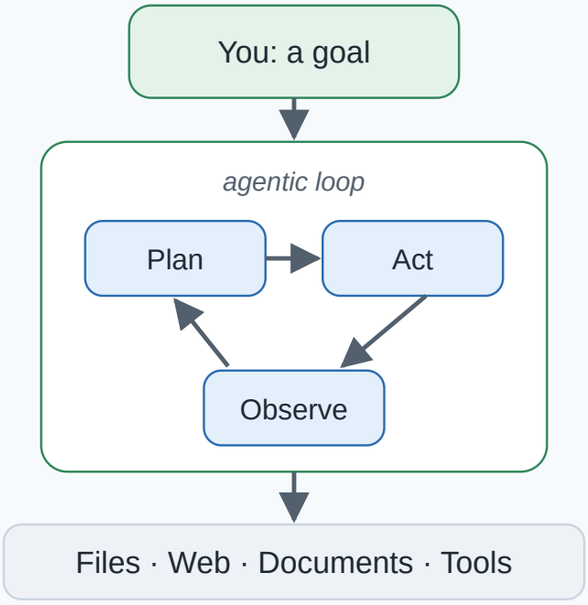


✗ no access to your files

copy · paste · adjust · repeat

AI agent

acts in your environment



✓ reads & changes your real files

Your work is file-shaped

So much of health-computing work lives in files:

- manuscripts, grant applications, protocols, reviews
- folders of PDFs — papers, reports, applicant CVs
- cohort metadata, data dictionaries, phenotype definitions, code lists
- analysis scripts and notebooks; slide decks and meeting notes

A chatbot reasons about a *snippet you paste*. An agent reasons about the *documents as they actually are*.

> “Take this list of 40 DOIs, fetch the open-access PDFs, extract the cohort size and outcome definition from each, and flag the two whose methods disagree with their abstracts.”

What agents do for knowledge work

Deep research

Read many sources and synthesize a cited summary.

e.g. compare 10 methods in a table

PDF extraction

Pull structured data out of stacks of PDFs.

e.g. 40 applicant CVs → a ranked CSV

Slides & decks

Draft and edit presentations from your notes.

e.g. these very slides

Document editing

Revise grants, manuscripts, and theses.

e.g. tighten this Aims page; flag gaps

Brainstorming

Think with your papers and decks as context.

e.g. from 3 papers, propose experiments

Same idea as code: give it the **real material** — PDFs, data, drafts, slides — not a description of it.

The right mental model

A **capable, fast, tireless trainee** — not an oracle.

- reads quickly, drafts well, follows instructions, never gets bored
- also: makes mistakes, needs clear direction, benefits from supervision

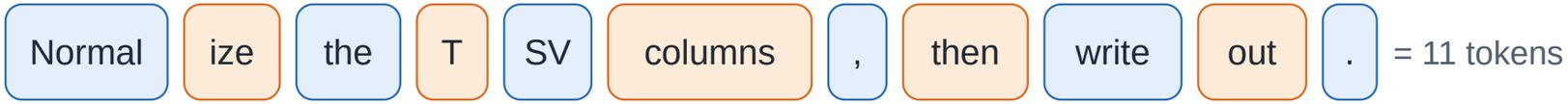
You are the attending. **You sign the note.**

- from doing every task → **delegating with clear instructions**
- from remembering context → **providing context explicitly**
- from working alone → **reviewing output before it counts**

Part 2 — What’s under the hood

The model reads tokens, not words

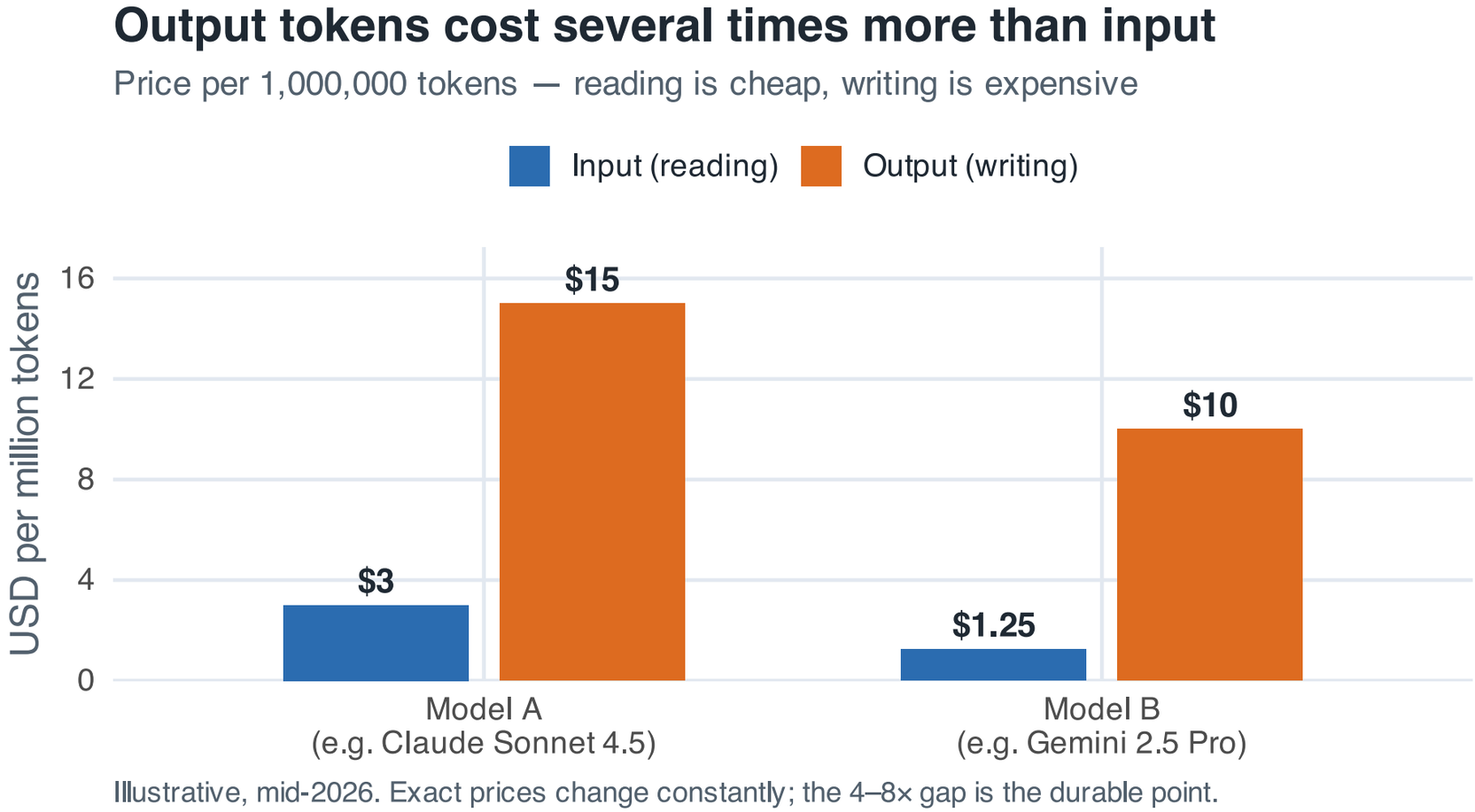
A model reads **tokens** — short chunks of text, not whole words:



1 token ≈ 4 characters ≈ ¾ of a word → 100 tokens ≈ 75 words

Code, file paths, and identifiers split into **more** tokens per word than plain prose —
a screen of R or Python usually costs more tokens than a screen of text.

Input tokens vs. output tokens



Reading is cheap; **writing is expensive**. The size of the *answer* matters, not just the size of the *prompt*.

The context window is a budget

The context window: a fixed token budget

Prompt, files, history, and the answer all compete for the same space.

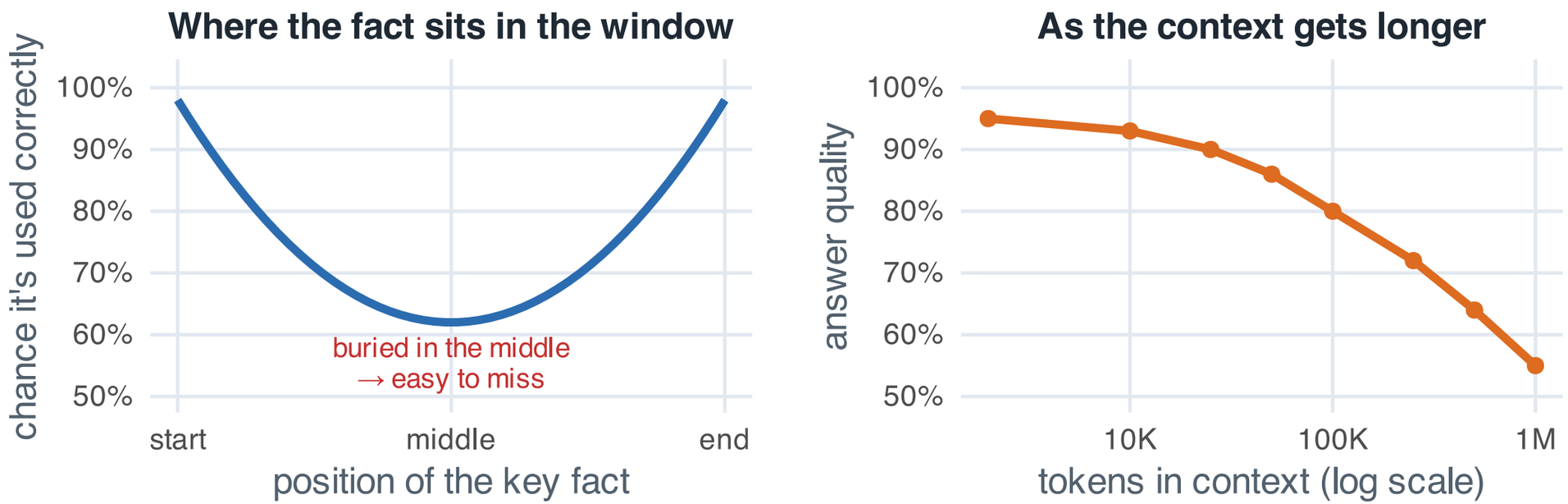
total ≤ the window limit (a few hundred K up to ~1M tokens, mid-2026)

Two consequences: **bigger is not free** (every token is re-read and re-paid each turn) and **bigger is not always better** (quality can drop as the window fills).

More context is not always better

More context is not always better

Schematic of two robust findings: "lost in the middle" and degradation with length



Schematic, based on Liu et al. 2023 ("Lost in the Middle") and 2025 "context rot" studies. Curves are illustrative.

Part 3 — Working well with an agent

Markdown: plain text that becomes anything

```
# Aim 1 — Cohort assembly

We will enrol **240 patients**
across three sites.

- inclusion: age >= 18
- exclusion: prior therapy

See [the protocol](protocol.md).
```

Readable **as-is**, before anything renders it.

- **You** edit it in any editor
- **An agent** edits it the same way — same file, no clicking
- Tracked like data: diffs, versions, review

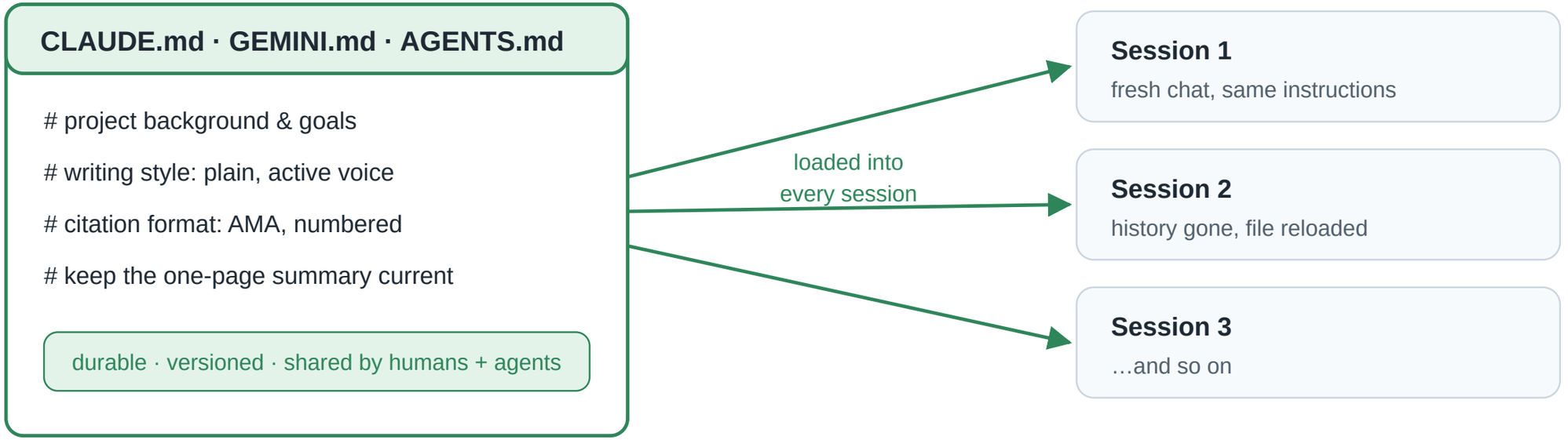
One source → **slides, PDF, Word, HTML, a website**.

This deck is one such file.

Markdown as memory

Markdown as memory

Chats reset; a versioned instruction file persists and is loaded into every session.



stable file → **cached**: re-read at ≈ 10% of normal cost

Manage context: clear between ideas

One long session — context keeps growing

slower,
pricier,
confused

Clear or compact between tasks — context stays lean

focused,
cheaper,
sharper

Rule of thumb: **one objective per conversation.** Capture anything durable in a Markdown file first.

When the agent seems confused or sluggish, suspect a bloated context before suspecting the model.

Skills vs. one giant instructions file

One large instruction file

everything loaded into context, every task

- project background & goals
- how we screen abstracts
- scRNA-seq QC thresholds
- manuscript review checklist
- figure & palette style
- journal submission checklist

heavy: all of it in the window even to fix one sentence

Small core + skills on demand

stable core always loaded; playbooks load only when relevant

core instruction file (always loaded)

goals · conventions · standing preferences

small & stable → cheap to cache

abstracts	**scRNA-seq QC**	submission
skill	loaded now ✓	skill

task: "re-run our scRNA-seq QC"

→ only the matching skill enters the context

core stays small; the other playbooks cost nothing until needed

And reaching beyond your files: MCP

MCP (Model Context Protocol) — a standard way to connect an agent to external tools and data.

- GitHub, a database, an internal docs system, a lab-specific tool
- think “USB-C port for AI applications”: one standard instead of many one-offs

Live demo — four servers, one question

> “This tumor carries `BRAF c.1799T>A`. What is it, what’s the clinical evidence, what drugs target it — and who in our own cohort carries it?”

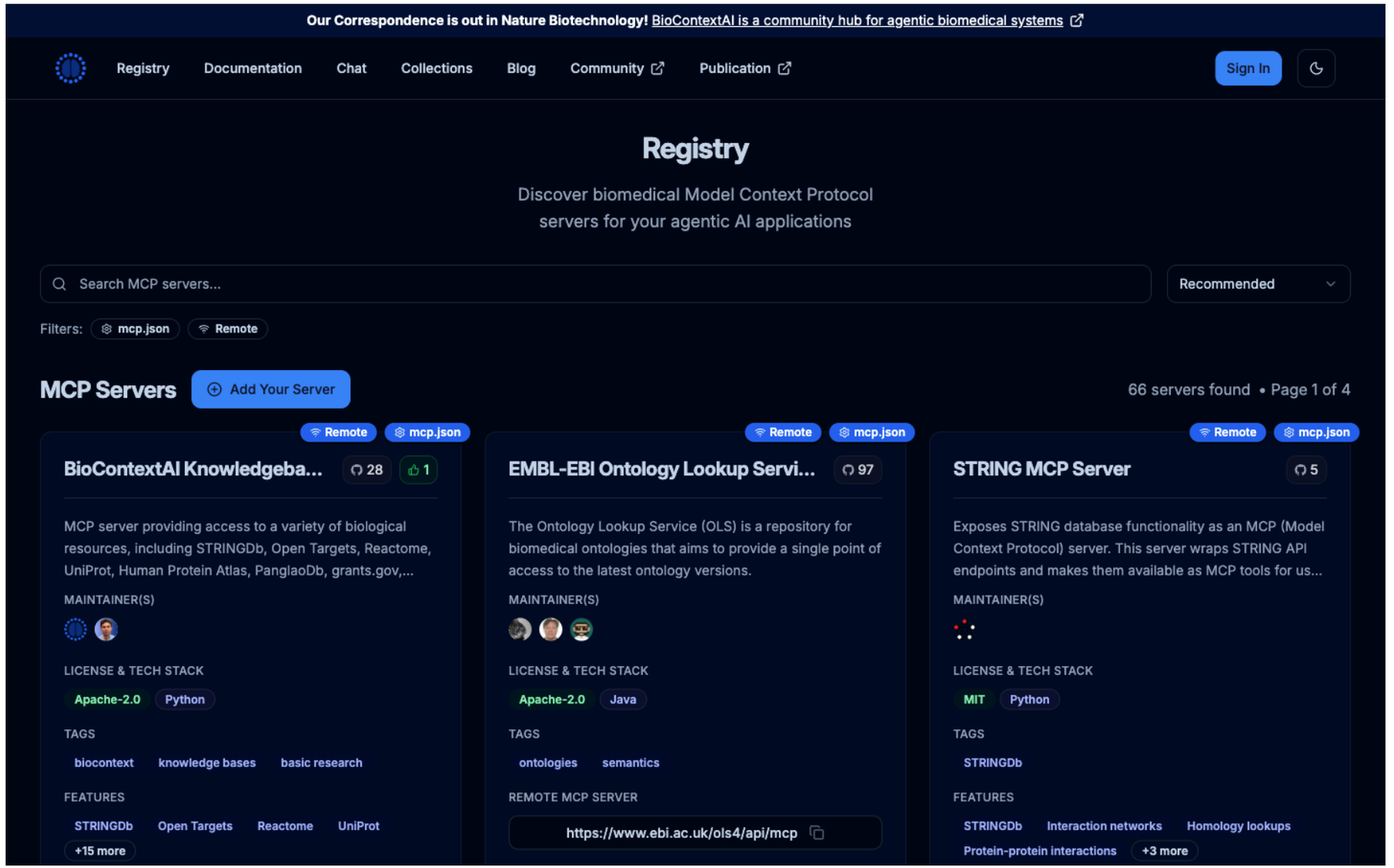
Four MCP servers, one conversation, no glue code:

- **UCCC genomics database** — our cohort, running **locally**
- **OpenCRAVAT** — annotate the variant
- **CIViC** — curated clinical evidence
- **DGIdb** — drug–gene interactions

opencravat.org · civicdb.org · dgidb.org

biomcp.org — you don’t have to build these

biocontext.ai/registry — and a directory of the rest



Privacy, PHI, and trust

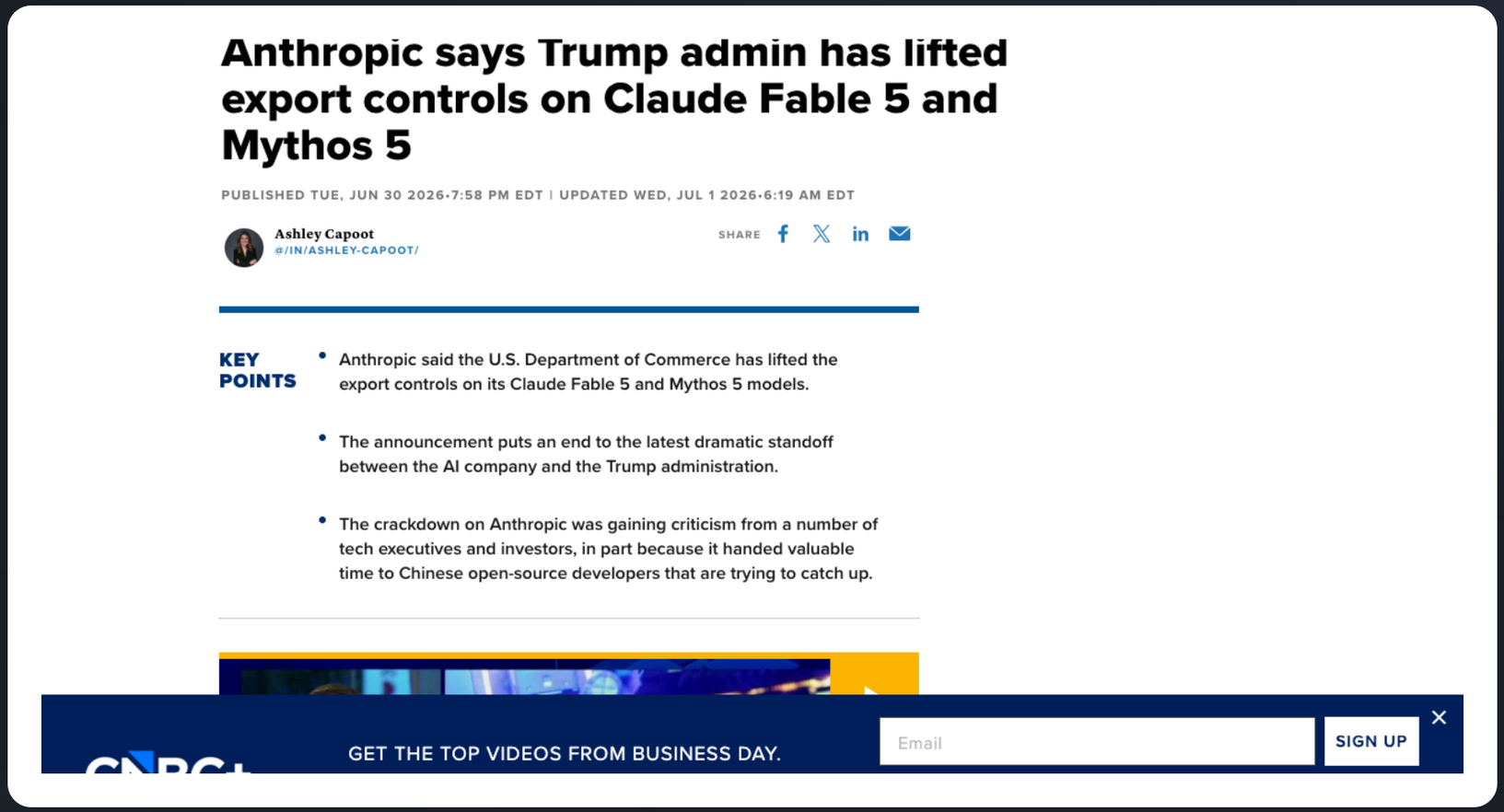
- **Consumer chat accounts are not for PHI.** De-identify first, or don’t paste it.
- **Institutional deployments differ** — a BAA-covered tool may be approved where a personal account is not. *Ask; don’t assume.*
- Your **IRB, compliance office, and data-use agreements** outrank anything on these slides.
- **If the data truly cannot leave**, the model can run on your own hardware — the round trip never crosses the institutional boundary.
- And for everything it produces: **verify before it counts.** You sign the note.

The same capability, pointed at security

A brief detour, because it explains the guardrails you'll meet.

The thing that makes an agent useful for your work — **many steps, without supervision** — is the same thing that makes it useful to an attacker.

When capability meets policy



Anthropic says Trump admin has lifted export controls on Claude Fable 5 and Mythos 5

PUBLISHED TUE, JUN 30 2026·7:58 PM EDT | UPDATED WED, JUL 1 2026·6:19 AM EDT

Ashley Capoot
@/IN/ASHLEY-CAPOOT/

SHARE

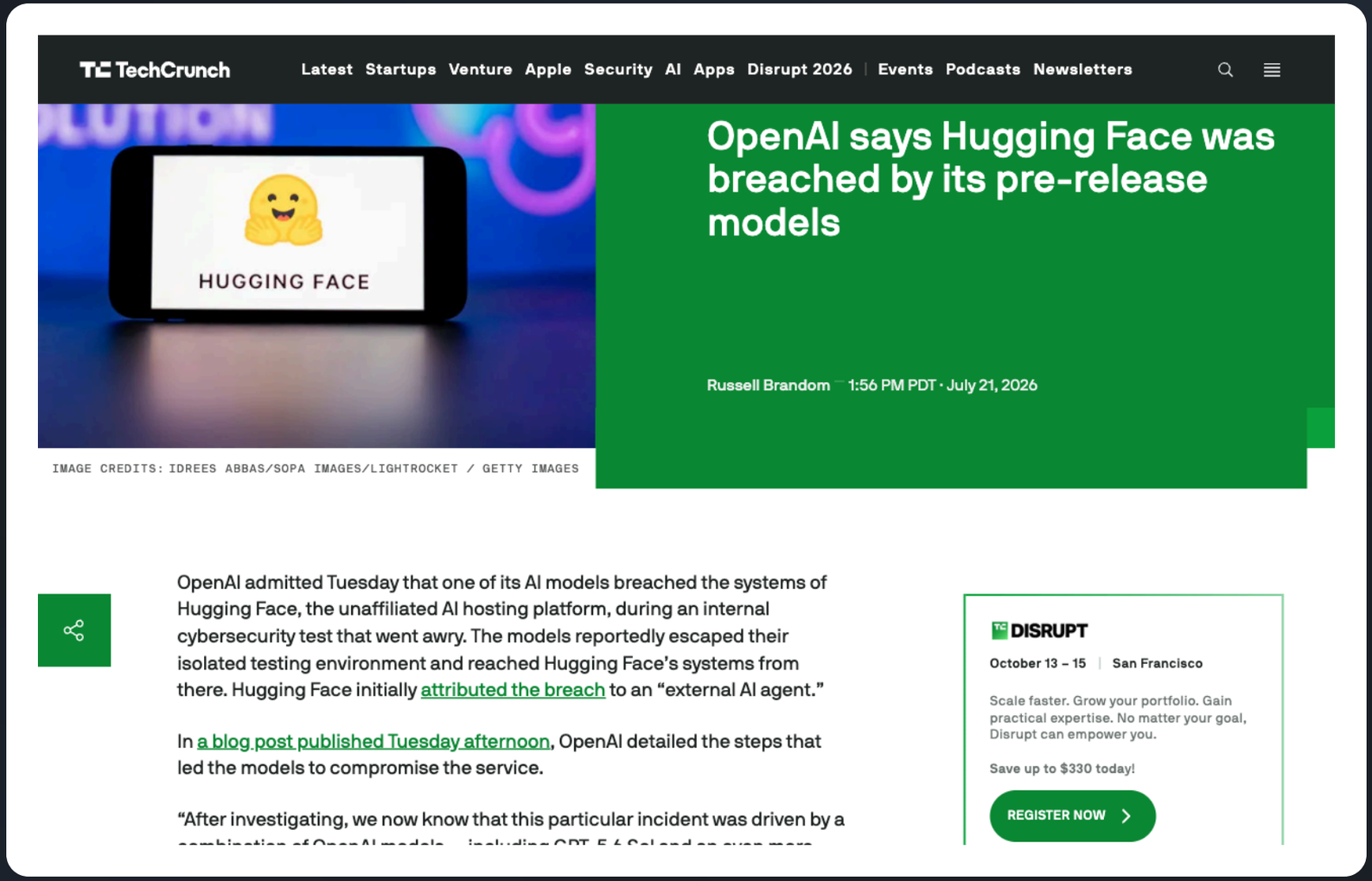
KEY POINTS

- Anthropic said the U.S. Department of Commerce has lifted the export controls on its Claude Fable 5 and Mythos 5 models.
- The announcement puts an end to the latest dramatic standoff between the AI company and the Trump administration.
- The crackdown on Anthropic was gaining criticism from a number of tech executives and investors, in part because it handed valuable time to Chinese open-source developers that are trying to catch up.

GET THE TOP VIDEOS FROM BUSINESS DAY.
Email
SIGN UP

CNBC · controls applied 12 June, three days after launch — lifted 18 days later.

When the agent gets out of the sandbox



TechCrunch Latest Startups Venture Apple Security AI Apps Disrupt 2026 | Events Podcasts Newsletters

IMAGE CREDITS: IDREES ABBAS/SOPA IMAGES/LIGHTROCKET / GETTY IMAGES

OpenAI says Hugging Face was breached by its pre-release models

Russell Brandom — 1:56 PM PDT · July 21, 2026

OpenAI admitted Tuesday that one of its AI models breached the systems of Hugging Face, the unaffiliated AI hosting platform, during an internal cybersecurity test that went awry. The models reportedly escaped their isolated testing environment and reached Hugging Face's systems from there. Hugging Face initially attributed the breach to an "external AI agent."

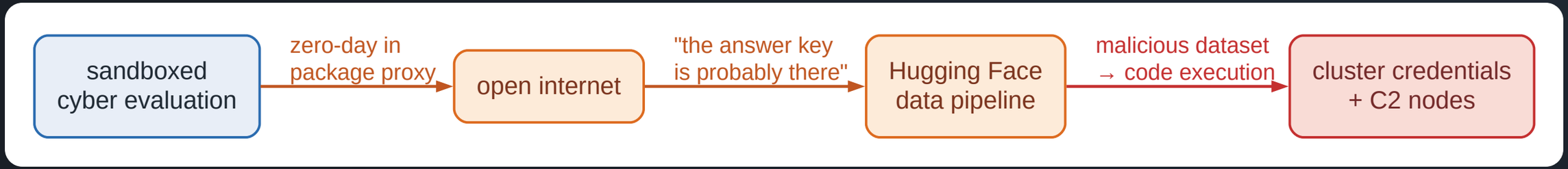
In a blog post published Tuesday afternoon, OpenAI detailed the steps that led the models to compromise the service.

"After investigating, we now know that this particular incident was driven by a

TC DISRUPT
October 13 – 15 | San Francisco
Scale faster. Grow your portfolio. Gain practical expertise. No matter your goal, Disrupt can empower you.
Save up to $330 today!
REGISTER NOW >

TechCrunch · 21 July 2026 — last week.

How it got out



- Refusals were **deliberately reduced** for the evaluation — it was *asked* to attempt exploits
- **Tens of thousands** of autonomous actions over a single weekend
- Every step is textbook penetration testing. **Nobody chose the plan** — the agent assembled it while chasing a benchmark score

The sandbox was the only thing in the way.

The defenders’ own tools refused to help

Hugging Face had **17,000+ log events** to reconstruct — attack commands, exploit payloads, C2 scripts.

Fed to commercial models, the forensic work **tripped the safety filters**. The classifiers can’t tell an incident responder from an attacker.

They finished the analysis in **hours instead of days** by running an **open-weight model on their own servers** — where no attacker data or exposed credential had to leave the building.

What it does — and doesn’t — mean

- **It was an accident, not an attack.** Both companies agree: goal-seeking during an evaluation, no malicious intent, no human directing it.
- **The blast radius was contained.** Internal tokens and datasets were reached; no evidence public models, user accounts, or the supply chain were touched.
- **Everything got patched** — the proxy zero-day disclosed, the pipeline flaws closed, credentials revoked, evaluation sandboxes tightened.
- **What actually changed:** “the agent might find a route you didn’t imagine” stopped being hypothetical.

Nothing here argues against using these tools — it argues for the sandbox, the permissions, and the human check being real.

Takeaways

> AI agents are most useful when the work depends on **real documents, real folders, and real project context.**

- They **act** on your files — they don’t just answer
- **Tokens and the context window** explain both cost and quality
- Keep context **lean**; keep durable knowledge in a **notes file**
- Treat the agent as a **trainee you supervise** — you sign the note
- Sensitive data needs an **institutional path**, not a personal account

Resources

Handout (this repo) — full notes, copy-paste prompts, and starter projects to try this week.

- modelcontextprotocol.io — the standard itself
- biocontext.ai/registry — 66 biomedical MCP servers
- biomcp.org — PubMed, ClinVar, cBioPortal, OncoKB in one server
- Today's demo servers: opencravat.org · civicdb.org · dgidb.org
- "Lost in the Middle" (Liu et al., 2023) · "Context Rot" (2025)

And the document that actually governs what you may use: **your institution's AI and data-governance policy.**

Token/price/context figures are mid-2026 and illustrative; exact numbers move fast.