② Causal Bench: research question

We want to know how well frontier models, both closed and open sourced, perform on causal inference and experiment design tasks: whether they have really mastered these skills, to what degree, and whether they can do the work on real tasks.

② Causal Bench: well-specified causal questions

QUESTION **When a causal question is fully specified, can a model carry out the analysis correctly?**

① Questions

328 questions with an underlying truth known in advance.

16 topics: cover the full range of experimentation and causal inference.

3 ways to generate a question:

(1) generated by our own DGP program

(2) adapted from real study data

(3) taken from a public causal benchmark

② Model testing

Input: the task tells the model what it needs to know.

(1) What the data looks like: the meaning of each column, and the first 5 rows.

(2) The data is clean: no missing values and no other problems.

(3) How to analyze it: the steps and assumptions are stated, or the case is a standard textbook one.

Rules: cannot see the full data, run its script, or search the web.

Output: (1) A JSON file: its conclusion, as an estimate, an interval and a decision. (2) A script: the method behind the conclusion. We run it on hidden data to check that the method is right.

③ Scoring

Graded by a program, not by an LLM or a person, so the scores are stable.

The answer gets several checks per question, such as identification, method, estimate, interval and diagnosis. For example, the estimate check passes only if the model's number is within a preset tolerance of the true value.

The script is run on hidden datasets the model never saw. A script that only prints a fixed number fails.

④ Ranking

Each model gets one pass rate. It is the mean over rounds, then over the questions of a topic, then over the 16 topics with equal weight.

Each score has a 95% confidence interval. It comes from a bootstrap with 2,000 resamples. Related questions are resampled together as one family.

Models are ranked by pass rate, in tiers. For each pair of models we bootstrap the difference on the same questions, with a Holm correction. Models that are not significantly different share a tier.

② Causal Bench: what the 328 questions test

16 topics in four groups. Green: number of questions in the topic. Red: average pass rate of the 11 models.

Running experiments 113

Experiment design

36 — 74%

Before launch, given a business goal: how many users are needed to detect the expected effect, how to split traffic, and how long to run.

Experiment analysis

32 — 87%

After the test, given its data: estimate the effect and its interval, then decide whether to launch.

Diagnosis and root cause

45 — 77%

When something is wrong: the experiment or its data has a problem that the task does not mention, such as an uneven split between groups. The model must find it, explain it, and not report problems that are not there.

Observational data 104

Observational estimation

56 — 80%

No randomization: users chose the treatment themselves, so the groups differ. Adjust for that, then estimate the effect.

Real designed-study data

20 — 93%

The same estimate, on data from real studies, not simulated data.

Bounds and sensitivity

16 — 89%

No single exact answer: report a range, and how strong a hidden factor must be to overturn it.

Heterogeneous effects

12 — 94%

The effect is not the same for everyone: find who benefits most, and decide whom to treat.

Causal structure 65

Causal discovery

26 — 79%

No graph given: from data alone, work out which variable causes which.

Identifiable-or-not pairs

17 — 97%

Two cases that look alike: the effect can be estimated in only one. Tell which, and decline the other.

Graphs and counterfactuals

14 — 98%

Given a causal graph: answer what-if questions about a case that did not happen.

Mediation, time, spillover

8 — 93%

Not a simple effect: it goes through a middle variable, builds over time, or spills over.

Controls and engineering 46

Text and commonsense

19 — 94%

No data: read a situation described in words, and judge cause and effect.

Replication, open analysis

11 — 90%

No steps given: from data and a short brief, do the full analysis.

Audit

10 — 97%

Someone else's code: read it, without running it, and find the errors.

Research design

3 — 95%

Before any data: choose the study design that fits the goal.

Other

3 — 100%

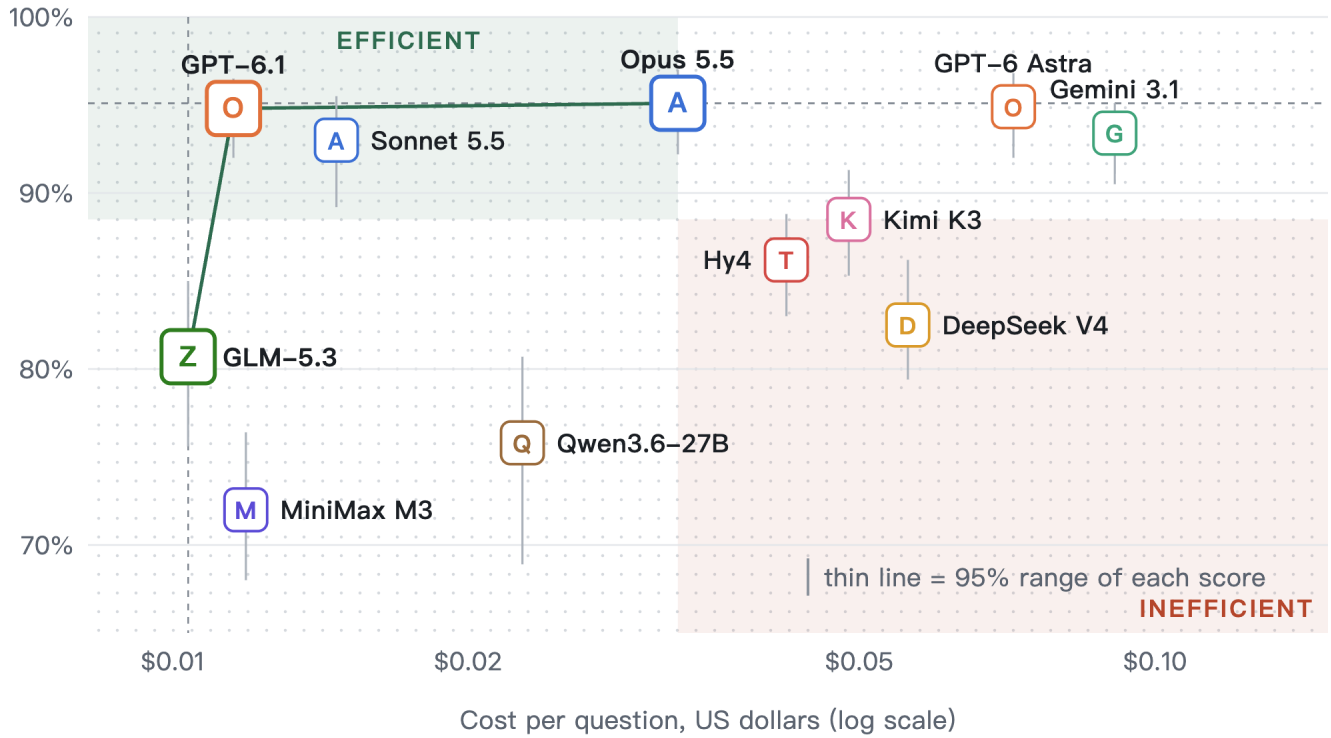
Rare cases that fit no other topic, such as simulated experiments.

HOW THEY WERE MADE **263** generated by our own DGP program · **35** adapted from real study data · **30** taken from a public causal benchmark

② Causal Bench: how 11 models did on 328 questions

The five models at the top cannot be told apart, and half of the questions are too easy to separate models.

Pass rate × cost per question



Tiers: Opus 5.5 ≈ GPT-6 Astra ≈ GPT-6.1 ≈ Gemini 3.1 ≈ Sonnet 5.5 > Kimi K3 ≈ Hy4 > DeepSeek V4 ≈ GLM-5.3 ≈ Qwen3.6-27B > MiniMax M3

① Weaker models are unstable and often fail on engineering

About 40% of their failures are a crashing script, cut-off output or a bad format. Asked the same question again, their result changes in 25% of cases; for the top five, in 9%.

② Code separates strong from weak models

With a script to write: top five 90%, the others 65%. Without one: 93% and 83%.

③ Models know the concepts; applying them is harder

Average of the 11 models: reasoning on graphs and identifiability 97–98%, experiment design 74%, diagnosis 77%. On the 51 questions that give no data, even the weaker models reach 95%. Textbook concepts are no longer what separates models. Working with data is.

④ Top models have solved well-specified tasks

When the data is clean and the task gives the background and the instructions, the top five pass 93–95%. **v1 can no longer tell them apart.**

③ Causal post-training: research question

A model that touches a company's internal data often has to be an open model, deployed locally. Open models still trail the frontier models. We want to know whether post-training can make them better at causal inference and experimentation tasks.

③ Causal post-training: Line A

Three training steps on Qwen3.8-27B, four tests.



① CPT

Data: 85 passages (77,000 tokens) from the DoWhy and EconML concept guides. Plain text only.

Task: predict the next word, as in pre-training. Loss: cross-entropy, mean 1.39 (perplexity about 4).

Training: LoRA, 10 steps, learning rate 1e-5. Only a small add-on is updated.

Output: merged into the model. SFT starts from it.

Result: **+0.0025** on our in-domain test. No gain.

② SFT

Data: 13,817 question-and-answer pairs of three kinds: causal graphs, full cases and tool use.

Task: write the standard answer. Loss: cross-entropy on the answer only, from about 1.0 down to 0.03.

Training: LoRA, learning rate 5e-5, one pass of 1,727 steps.

Output: the version at step 1,350, best on our in-domain test.

Result: against the base model, **+0.28** on our in-domain test, **−0.30** on CLadder, **−0.21** on tool calling.

③ RL

Data: 100 simulated tasks from 3 of our own environments. The model acts step by step, and a program gives the reward.

Task: GRPO. The model writes several answers per task; answers above the group average are reinforced.

Training: LoRA, learning rate 1e-5, KL penalty 0.02, 100 rounds.

Output: the model we sent to Causal Bench.

Result: against the SFT version, **+0.0002** on our in-domain test, **+0.06** on CLadder, no change on tool calling.

Four tests: base → final

(1) In-domain test, 3,567 tasks like the training data: **0.59 → 0.87**.

(2) General skills, 6 public tests: **2 of 6 drop**. Tool calling **0.86 → 0.65**.

(3) CLadder, 1,500 outside questions: **0.76 → 0.52.**

(4) Causal Bench, 300 blind questions: **0.70**, in the last tier.

Rule: reject a version if two or more general skills drop. **Line A is rejected.**

③ Causal post-training: what Line A tells us

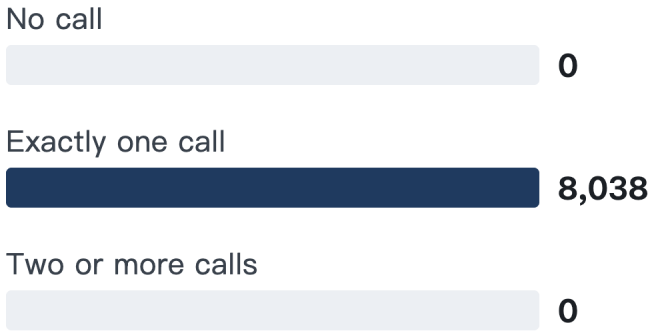
1 **Almost all of the change comes from SFT.**
On the in-domain test: CPT +0.0025, SFT +0.28, RL +0.0002.

2 **The model learned our format, not only the reasoning.**
Our own test went up by 0.28. CLadder, a test written by others, went down by 0.30. One reason: after training, the model answers in the format of our training data, even when CLadder asks for a different one.

3 **The tool-calling loss has one cause, in the data.**
Every training answer that had a tool makes exactly one call. The model learned that a tool means one call. It fails when no call is needed, and when several are needed at once.

4 **RL did not help.**
Our own test moved by +0.0002: no effect. A likely reason is three cautious settings together: LoRA trains only a small add-on, the learning rate is small (1e-5), and a KL penalty keeps the model close to the SFT version. So the model could change very little.

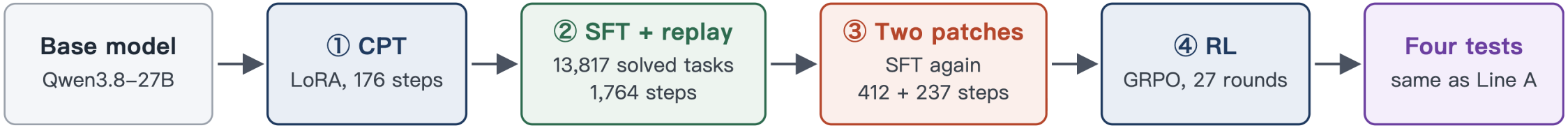
Tool calls in the training answers
the 8,038 answers that had a tool



NEXT We wrote about 1,600 counterexamples for the two missing cases, and tested them in a second, independent run: Line B.

③ Causal post-training: Line B

An independent repeat of Line A, with replay and two patches. The same four tests.



① CPT

Change: a larger corpus. The result is saved as a new base for all later steps.

Data: 1,411 passages of text, 1.57M tokens. About 20 times Line A's.

Task: predict the next word. Loss on held-out text: 1.23 before training, 1.18 after (perplexity 3.4, then 3.2).

Training: LoRA, learning rate 1e-5, one pass of 176 steps.

Result: **+0.006** on our in-domain test, **−0.05** on CLadder.

② SFT + replay

Data: the same 13,817 question-and-answer pairs as Line A, 2.06M answer tokens.

Change: 5% of the training tokens are replay. The base model answers general tasks, and we keep the answers a checker confirms.

Training: LoRA, learning rate 5e-5, one pass of 1,764 steps. Loss ends at 0.05.

Result: **+0.28** on our in-domain test, **−0.15** on CLadder, **−0.10** on tool calling. One skill still drops.

③ Two patches

Patch 1: 412 more SFT steps on 3,300 examples: 800 that need several calls at once, 300 that need no call, and 2,200 ordinary ones, so the model does not over-correct.

Result: tool calling **0.77 → 0.85**, code **0.87 → 0.84**.

Patch 2: 237 more steps on half of each kind, plus 50 code examples.

Result: code **0.84 → 0.86**, tool calling **0.83**. Both are still slightly below the base model.

Both: our in-domain test does not move. CLadder **0.60 → 0.67**.

④ RL, and the four tests

RL: the same recipe as Line A, 27 rounds. No effect.

Base → final: in-domain test **0.59 → 0.88**, tool calling **0.86 → 0.83**, CLadder **0.75 → 0.67**.

Rule: no general skill drops past our threshold. **Line B passes.**

③ Causal post-training: what Line B tells us

1 **The same failure appears again.**

After SFT, tool calling falls to 0.77, even with replay. Of 97 new errors, 95 are the same kind: several calls were needed, and the model made one.

2 **Replay helps only where it has enough examples.**

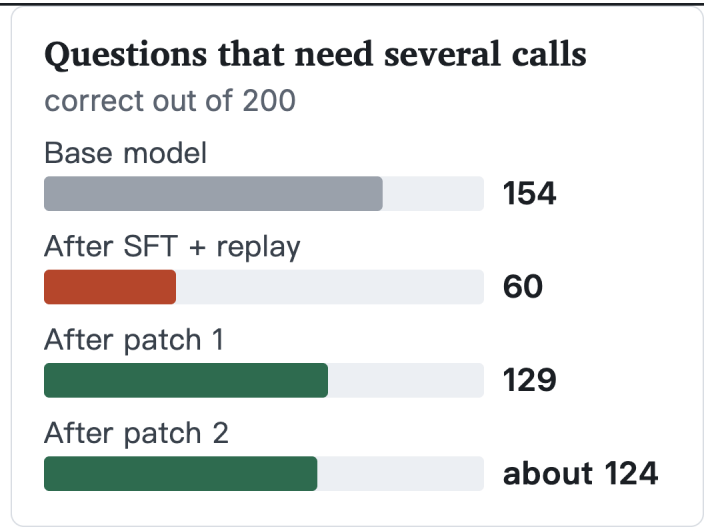
Replay is 5% of the training tokens, but tool examples are only 0.4%. That is too few to protect tool calling. Questions that need several calls at once fall from 154 to 60 correct, out of 200.

3 **A small change to the data fixes it.**

Two patches, 649 steps in all, bring tool calling back to 0.83. Our in-domain test does not move: 0.875, 0.876, 0.876.

4 **At least half of the outside loss was format.**

CLadder recovers from −0.15 to −0.08 while the causal score stays the same. The model already knew these answers. It was not writing them in the format asked for.



STILL OPEN On CLadder, three question types recover after the patches. One does not: backdoor adjustment falls from 0.47 to 0.03 after SFT and stays there. The cause is not known yet. RL again had no effect.

③ Causal post-training: conclusions

1 **Post-training works on the target.**
On our in-domain test the score rose from 0.59 to 0.87, in both lines.

2 **One test would have hidden the cost.**
The same model lost on tool calling and on an outside causal test. Only the four tests together showed it.

3 **The damage came from the data.**
Every training task with a tool used exactly one call. Two independent lines showed the same failure.

4 **A small, targeted fix was enough.**
About 1,600 counterexamples and 649 steps. No general skill drops past our threshold, and the in-domain score did not move.

> **STILL OPEN** **RL added nothing** in either line. **Backdoor adjustment** questions on CLadder fell from 0.47 to 0.03 in Line B. On **Causal Bench** the Line A model is still in the last tier.