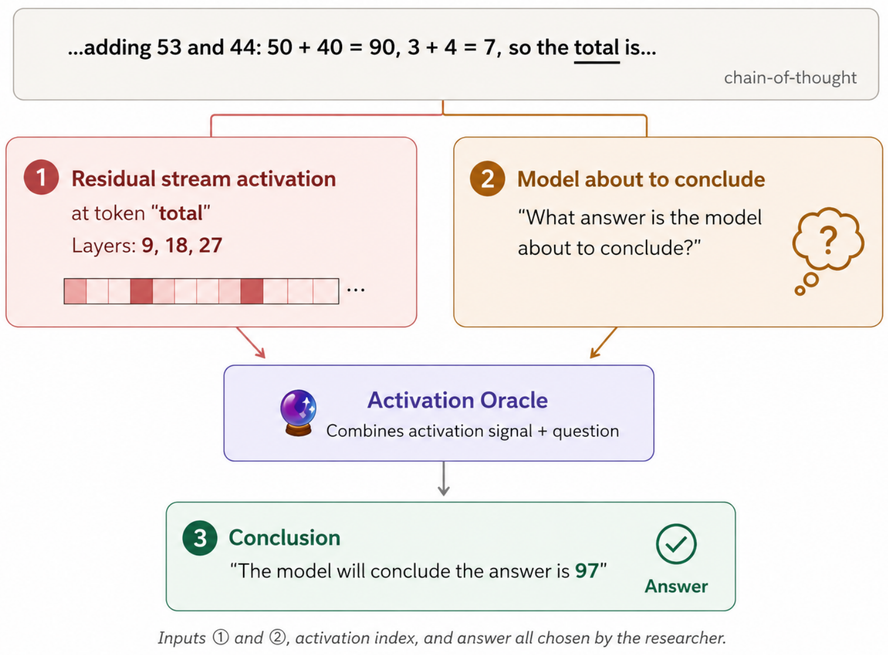


Inputs ① and ②, activation index, and answer all chosen by the researcher.

The Activation Oracle receives residual-stream activations plus a natural-language question, and answers about the model state they represent.

Train your Activation Oracle on questions that are actually answerable from activations.

A conversational dataset built for **solvability**, together with multi-layer feeding, on-policy lens data, and stronger injection, makes AOs **less vague** and **less hallucinatory**, with **large gains** on AObench.

Building Better Activation Oracles

Jan Bauer[*,1,2] Celeste De Schamphelaere[*,1,3] Adam Karvonen[4] Niclas Luick[1,5] Neel Nanda

[1]MATS [2]Gatsby Unit, UCL [3]Ghent University [4]Independent [5]University of Hamburg [*]equal contribution, order random

1 TL;DR

MAIN 3 TAKEAWAYS

- **Activation Oracles (AOs)** are LLMs finetuned to answer natural-language questions about another model's activations. Current AOs **hallucinate**, are **vague**, and evaluations are confounded by **text inversion**.
- We rebuild the training recipe; the biggest win is a conversational dataset built for **solvability**: questions answerable from the *activations* rather than the surrounding *text*.
- We release **AObench**, the first comprehensive AO evaluation suite. The full recipe **substantially raises** the chance-adjusted score and **reduces** both hallucination and vagueness.

2 Background / Motivating question

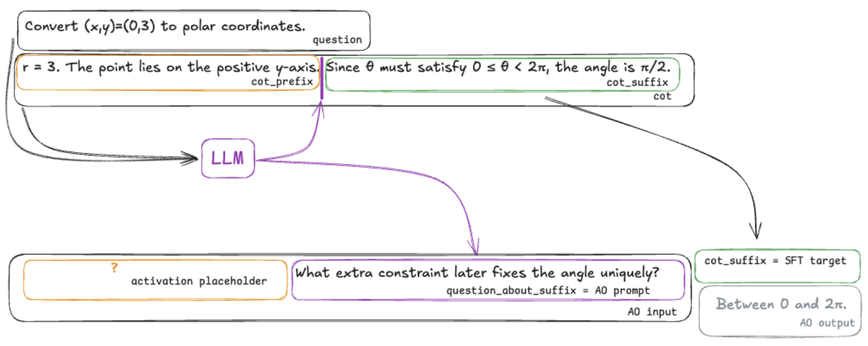
AOs **[Karvonen et al., 2025]** map model internals to natural-language explanations. The standard recipe: the *LatentQA* dataset **[Pan et al., 2024]**, binary classification tasks, and a past/future-lens objective on FineWeb, with norm-matched additive injection after layer 2. In practice, AOs are hard to use and evaluate **[Jakkli et al., 2026]**:

- **Hallucination**: invented, unsupported specifics.
- **Vagueness**: generic, unfalsifiable non-answers.
- **Text inversion**: the AO succeeds by reconstructing nearby text and answering from it, like any black-box oracle could.

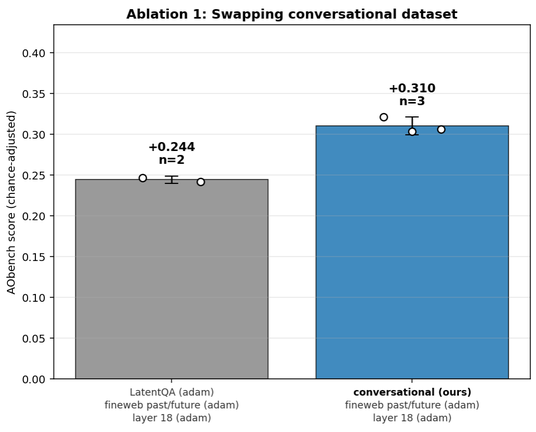
Can better training data fix this?

3 Better training data: solvable and on-policy

LatentQA's answers are often *not retrievable from activations at all*, which rewards text inversion and guessing. Its questions also target the user prompt instead of the model's own reasoning. We replace it:



An LLM splits the target model's own chain-of-thought into prefix and suffix, then writes a question about the *suffix* that is hard to answer from the prefix's text (no text inversion) but plausibly answerable from the prefix's activations (solvability).

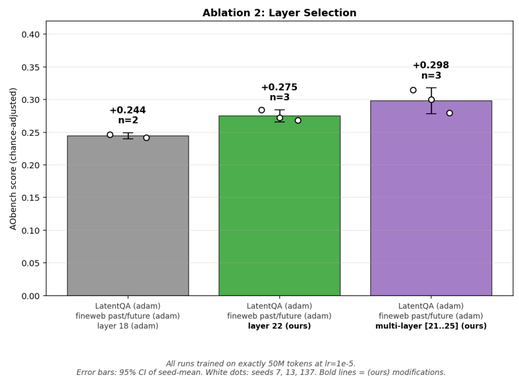


All runs trained on exactly 50M tokens at lr=1e-5.
Error bars: 95% CI of seed-mean. White dots: seeds 7, 13, 137. Bold lines = (ours) modifications.

Swapping *only* the conversational dataset is the **largest single step** in our recipe ($n = 3$ seeds). Responses become more specific, less vague, and follow instructions better.

On-policy lens data. Predicting FineWeb continuations mostly needs the *prior text* rather than the model's state. Sourcing the past/future-lens corpus from the interpreted model's *own* CoT rollouts instead gives a **small but real gain**.

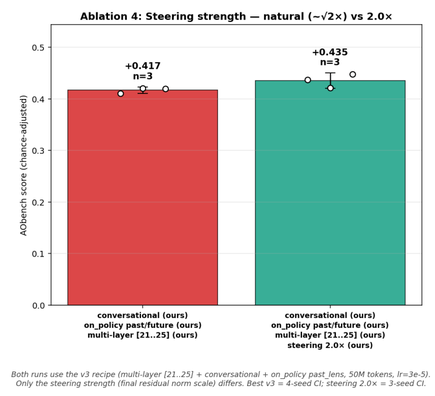
4 Layer choice & injection strength



All runs trained on exactly 50M tokens at lr=1e-5.
Error bars: 95% CI of seed-mean. White dots: seeds 7, 13, 137. Bold lines = (ours) modifications.

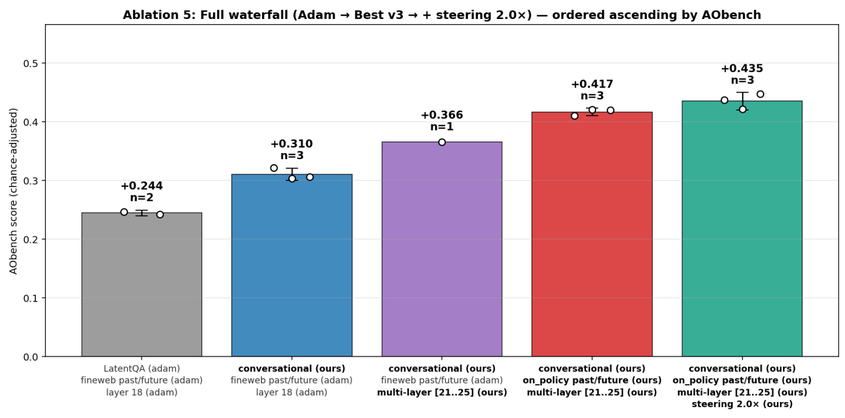
Layer choice. The baseline reads layers at 25/50/75% depth. Performance peaks at layer 22 (≈62% depth); feeding **5 contiguous layers** (21–25) adds a **further gain**, largest on model diffing.

Injection strength. Norm-matched *additive* injection after layer 2 consistently beats NLA-style embedding replacement. A 2× injection strength gives a marginal overall gain, but **noticeably improves hallucination**, so tune this carefully.



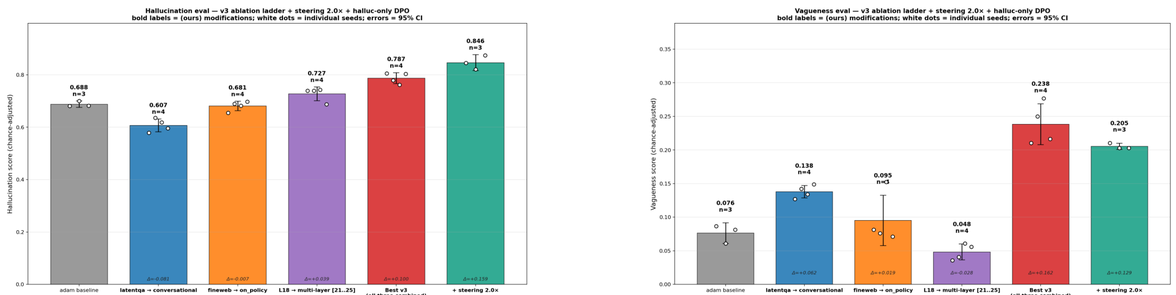
Both runs use the v3 recipe (multi-layer [21..25] + conversational + on_policy past_lens, 50M tokens, lr=3e-5).
Only the steering strength (final residual norm scale) differs. Best v3 = 4-seed CI; steering 2.0× = 3-seed CI.

5 Results: the ablation ladder



All runs trained on 50M tokens at lr=1e-5 (Best v3 + steering 2.0× use lr=3e-5).
Error bars: 95% CI of seed-mean. White dots: individual seeds. Bold lines = (ours) modifications.

Each bar adds one intervention to the previous recipe. All runs: exactly 50M tokens, matched learning rates, multiple seeds (95% CI).



Hallucination improves monotonically once accounting for the model making more specific (hence falsifiable) claims; vagueness improves correspondingly (chance-adjusted).

6 Limitations and discussion

- The on-policy ablation is not perfectly clean (our conversational data is also on-policy); LLM judges are noisy. The recipe *ordering* is stable across seeds and variations.
- Narrow post-training [Ivanova et al., 2026] **fails to beat linear probes**; AOs still hallucinate, and often the CoT alone gives the same insight. They shine on **open-ended questions** and **latent-reasoning models**.
- **NLAs** [Fraser-Taliente et al., 2026] are a promising AO pretraining route, and should benefit from the solvability principles introduced here.

code

models & datasets

celestedeschamphelaere@protonmail.com · japhba@gmail.com