D2.2 post: outline

A working outline for the LessWrong post on D2.2, the next in the sequence after *Anchoring one concept in a transformer* (D2.1 post). It collects what we would like the post to say, in the order a reader would meet it, and names where the evidence for each point lives, so the post can be written from the reports. Open points are marked **open**, each with a recommendation. The design doc holds the claims and the recipe in full; this page is about telling them.

The post at a glance

Reader

Someone who may not have read the earlier posts. The D2.1 post opened with a hand-drawn picture of the residual stream and built the recipe up one piece at a time, in about 2,200 words with ten figures. This one can lean on it for the basics, and should stay near that length. Method detail goes in LessWrong footnotes where a sentence or two will do, and in an appendix linked from the site where it won't. The post leaves out our internal names: the claim letters, recipe names such as `τ_ℓ fixed` and "per-visit", and, where practical, terms we coined (spill criterion, target null), saying what each one means in plain words instead.

Headline

We anchored a concept the model has to infer, an operation that is never written down, onto an axis we chose. Editing that axis out removes most of what the model does with the op, in proportion to how much each context needed it, and leaves the other ops alone. On this model a direction fitted after training (LEACE, INLP) removes the op about as well, and the two edits are hard to tell apart. What the anchor adds is a location known before training: no search over where and how hard to edit, an edit that stayed within the spill criterion on every run with no tuning for each model, and a fixed geometry to analyse. Like the post-hoc methods, the edited model ends up a little short of a model that never learned the op, but we have a lead on where the rest of the evidence for the op is kept.

The sentence on fitted directions belongs in the opening summary at the top of the post, since it answers the first question a reader will ask.

What D2.1 promised

The D2.1 post said we would try intervening on the anchored models, add more ops and anchor them, and compare with other methods. This post delivers the first two (with one op anchored of seven), and compares with several of the other methods (directions found after training). Gradient routing and SAE steering stay future work.

Decisions

Settled: the body of the post reports on the per-visit runs, since the reference model, the confusion matrices and the write bound all come from them, and the setup says plainly how the labels were drawn. Over 400 epochs about 34,000 of the 42,891 `difference` contexts carry a label at some point. Each label scheme stands in for something real. The per-visit draw is like a very large corpus, where most contexts are seen once and a fixed share of them is labelled. The fixed draw is like a small labelled set reused over many epochs. The fixed draw gets its own part of the labels section (section 7).

Settled: the post includes the post-hoc comparison, now that its results are in, with `τ_ℓ fixed` and the per-visit draw as its anchored arm (section 6).

Leaning: no further experiments before the post. The comparison carries one caveat instead, that its reading without role labels is a lower bound on the fitters until a position gate is tried (section 6). The decision is being settled in the thread that weighed the comparison.

Settled: the low runs get a part of their own, and the post goes ahead without fixing them.

Open: how much of the path to the recipe to show. Recommendation: one short section with one figure and the two problems and their solutions (section 2), since the path is where most of the work went but little of what a reader needs.

1. From a word to an operation

- D2.1 anchored *red*, a concept with its own token. Most concepts worth controlling are never named in the text; the model works them out. So D2.2 asks whether anchoring works for a concept that has to be inferred (pivot).
- Why an inferred concept matters beyond this toy: the behaviors people most want to steer, such as a persona, are inferred from context too. Africa and Irving's research agenda (Thousand-dimensional structure) treats a persona as a latent the model infers, with a distribution over personas; the op here is a small case of the same shape, with a posterior over ops. Cite it as motivation, since it reports no results of its own.
- The task in one picture: a context of three solved examples and a query, with `?` where the op word would be. The model infers the op from the examples. This is where the grammar illustration goes: each example drawn as `a ? b = y` with `a` and `b` as small images and `y` their combination under the hidden op, then the query with `y` as the model answers it pixel by pixel (backlog).
- Seven ops, `difference` anchored to e_1 of the residual stream of the same small nGPT as before (four blocks, 64 wide).
- When the evidence is mixed: some examples show the answer of another op (replacement op noise), so a context can leave the op in doubt. The posterior over ops says how much each context needs `difference`. The target null° is what an ideal predictor that never knew `difference` would answer, weighing the other ops by how well they fit, and removal is measured as a share of the way to it (design § scoring). This needs only a short paragraph here, since section 4 measures against it; the picture of it waits for section 5.

Target null. Where a removal should land…

2. Anchoring an op, briefly

- What carried over from D2.1: the anchor and anti-subspace terms, labels on whole contexts, pooling that lets the term choose its states.
- What changed: the pull is pooled over slices as well as positions, so we do not have to say which block holds the op, and the anti term falls off with depth (design § recipe).
- The two problems along the way and their solutions, in a sentence each: a color embedding standing in for the op, solved by holding the anti weight high on the first slices (embedding-lean, late-pull); then block 1 writing e_1 on every op, solved by keeping the pool over slices sharp (slice-terms, soft-pool-scout). One figure of spill° falling across the recipes, perhaps from recipe-compare.

Spill. The side effect of an edit on the other ops…

- The mellowmax pooling picture from D2.1, redrawn with the slice axis added. Its SVG in the repo (pooling.svg) is small and hand-written, with light and dark colors, so adding an axis there is easy; editing the Figma original works too.

3. Did it land? (claim a)

- We did not know where the model would hold the op, only that it had to be somewhere. Pooling over slices and positions let the anchor term find it.
- The anchor sits on the example answers, where a context shows the most about its op, and its alignment° there rises with the posterior on `difference` (fixed-effects; example-evidence).
- It held on every run: 8 of 8 fresh seeds and 40 of 40 runs called early at epoch 64 (fixed-confirm, loss-rate). Seven of eight passed the task gate; the eighth learned two HSV ops late.
- Figure candidates: α by role and slice, which shows where the term found the op; the task change per op.

Alignment. How far a state points along e_1...

4. Taking it out (claim b)

- The full edit removes 0.88 of the way to the target null° on the seed mean (0.83 to 0.91), and spill° stays under the criterion on every run at every dose (fixed-effects, reference-baseline).
- The dose curve: removal is complete by three quarters of the full edit, and the drop on each context follows how much it needed the op.
- Why we edit every slice: an edit at fewer slices has to land between the block that writes the op and the block that reads it, and the reader moves from seed to seed. A larger model has readers at depths we don't choose, so the full edit is the one that carries over (block1-edit).
- Where the op lives, briefly: block 1 writes the evidence of each example onto e_1, one later block reads it at the query `=`, and an edit works when it sits between them (block1-edit). Section 7 builds on this picture.
- Figure candidates: dose curve with spill underneath; share of the way to the target null across posterior bins.

Target null. Where a removal should land…

Spill. The side effect of an edit on the other ops…

5. Where the answers go (claim c)

- The target null° in a picture: one context with its posterior bars, before and after the op is removed, beside what the edited model answers.

Target null. Where a removal should land…

- The edited model sends the mass to the other ops in about the right overall proportions, but context by context it covers about half the distance to the target null (fixed-effects confusion matrices).
- A model trained with no loss on `difference` contexts (sans-diff) comes 0.58 of the way where the edited model comes 0.50, and misses in the same way. Measured against a second sans-diff run, the edited model is about 85% of the way from the control to sans-diff. Name that endpoint whenever the 85% is quoted, since the plain share is about half (filtered-corpus).
- The surprise worth a paragraph: on contexts of other ops whose examples point partly at `difference`, the edited model gets better, because the mass the op took now goes to the right op. The gain grows with the posterior on `difference`; in the top bin the edit gets about 75% of the gain the target null predicts, and sans-diff about 94% (filtered-corpus E4).
- That gain is a small, measured case of intervening on one inferred latent and seeing the mass move where inference says it should, which ties back to the persona framing of section 1.
- Figure candidates: confusion matrices (control, edited, sans-diff); the gain on other ops against the posterior.

6. Compared with directions found after training

- Open with the same sentence as the summary: on this model a direction fitted after training removes the op about as well as the anchored edit, and the two edits are hard to tell apart on grading, landing and where the answers go. Then what the anchor still offers (post-hoc compare).
- What was compared: the difference in means, a ridge probe, LEACE and INLP (linear erasers), and a direction found by gradient search, each fitted on the controls with every label and tuned per run over slice, site and dose, and edited through the same scorer as the anchor.
- The numbers, in two tables side by side. With role labels and a per-run grid, LEACE and INLP remove 0.94 to 0.97 of the way within the spill° criterion, against 0.88 for the anchor, and land a little closer to the target null° (0.54 to 0.55 against 0.50) with the same confusion pattern. Pooled over positions, which needs no role labels, LEACE removes 0.91 and INLP 0.89, about level with the anchor; at matched removal they spill past the criterion on 2 runs of 8 against 0 of 8, within chance of each other. The difference in means, a common choice for steering (to verify against recent papers before calling it the usual one; candidates include contrastive activation addition and the refusal-direction work), falls behind once role labels are taken away.

Spill. The side effect of an edit on the other ops…

Target null. Where a removal should land…

- So claims (b) through (e) describe what removing one direction does in this model. They hold for the anchored edit and for a fitted one, and the post should present them that way, in sections 4 and 5 as well as here.
- The case for the anchor is a different cost profile, as an alternative to an eraser:
    - The cost: an extra training term, a task score 0.003 lower, and a recipe that took some thirty experiments to settle. Post-hoc methods apply to a model that already exists, which is why most people use them. Say this plainly.
    - No search: the location is known before training, so there is no sweep over slice, site and dose for each model, and the full edit stayed within the criterion on all 8 runs with no tuning for each model. A fitted configuration chosen on the other seven runs missed the slice on one run in eight (seed 712), which kept 0.36 to 0.43 of its removal. Tell that in a sentence with the sample size beside it.
    - A fixed geometry to analyse: because e_1 was chosen, the write bound could be read off the clean model before any edit, and one end of the arc of section 7 had a name we chose, which made the arc quick to read. A model trained without the anchor holds the same arc, found from the class means after training, but a short one: the two means lie about 21° apart at slice 3 in the controls against 83° in the anchored runs (arc-in-controls probe). So the post should say the anchor stretches the arc and names one end in advance, and leave out "only visible because". This is where the program is heading (several concepts, designed subspaces and circuits), and the post can name it as the direction of travel.
- What to leave out: label efficiency as an edge, since the fitters stay flat down to one label in fifty; the ceiling hypotheses (H3 and the gradient-searched direction), since they turn on the baseline and invite a question a footnote can't answer; and the verdict words of the report. Say what happened in plain words and link the preregistration.
- What to put in, briefly: a sentence or two of these is enough, said plainly and precisely, so they don't take over the section. Which analyses were not preregistered (the pooled reading, the configuration chosen on other runs, the axis-baseline reading of H4); the pooled table beside the main one; and above it the caveat that the pooled reading is a lower bound on the fitters, since a per-position gate from the fitted direction's own score might recover much of the role-label edge without role labels. Two smaller caveats to carry: the seed 712 story is one run in eight, and the LEACE probe result is an open puzzle in the report. If the free re-analysis of the configuration chosen on other runs (picked at a small spill margin in place of at the criterion) runs before the post, the no-search point takes its number.
- Figures: one frontier panel per method, as seed-mean frontiers with a sans-diff mark; the confusion matrices with the anchored row set off by a bold label.

7. How few labels? (claim e)

- One `difference` context in fifty carries a label, the shape of label we expect for a concept in natural language, and about a quarter of the labelled contexts have a posterior below one half on `difference`, so many labels mark an op the context does not show (design § e).
- The fixed draw: the anchor held at 1/50, 1/25 and 1/10 on all 24 runs, and most runs removed the op about as well as under the per-visit draw. At 1/25 removal matched (with one late spill°); at 1/50 and 1/10 it averaged 0.81, pulled down by a low tail of two or three runs in eight. No trend with label count between about 900 and 4,300 labelled contexts (fixed-labels).

  Spill. The side effect of an edit on the other ops…
- The fitters in section 6 were as flat down to one label in fifty, so scarce labels are enough for both, and the post shouldn't present them as an edge for the anchor.
- Retraining the same labels and seed moved which runs ended low, so whether a run ends low comes down to small numerical differences along training (fixed-labels E5).

What the edit misses, and why some runs end low

- Every run writes the evidence that an example fits `difference` as one vector with two parts: along e_1, where the anchor put it, and along the direction a non-fitting answer has, which the loss made on its own. The readers (attention, at block 3 or 4) take some of the evidence from each part, and the edit reaches only the first (low-runs).
- Anchoring as an interpretability tool: fixing one end makes the other visible by contrast. The evidence is one arc on the unit sphere, from the non-fitting direction to e_1, and each example sits along it by how well it fits. Without e_1 in place these would be two unnamed directions. Figure candidate: the answer states of one run projected onto the plane of e_1 and the non-fitting direction, fitting and non-fitting examples in two colors, slices 2 and 3 side by side, with the arc drawn.
- A cheap check before writing: whether the non-fitting direction is shared across ops or differs by op (stored checkpoints, no training).
- Which aspect of the evidence the readers lean on is invisible to training: twins with the same labels, seed and batches match in loss and α° for the whole run and part only in removal.

  Alignment. How far a state points along e_1…
- What this says about anchoring: it controls where one aspect of the evidence goes, and the edit is bounded by how much the readers use that aspect. The fixed-label low runs are the stress case; under the per-visit draw the same thing costs a few hundredths.
- The worry that an edit pushes a behavior into dimensions we can't see is the central risk in Thousand-dimensional structure. Here the remainder sits along a direction the loss made during training, so it was there before the edit and the edit missed it. The write bound and spill check directly for anything the edit itself pushed elsewhere. A worked case of that distinction may be useful to readers of the agenda. Treating the remainder as a lead is also what found the arc, a habit the already-clean essay argues for.
- The headline says "most of what the model does with the op" and points here for the rest.

8. A change of prior

A short reflection in the first person, drafted from a note in review and tied to Cammarata's essay. It sits after the results and before next steps: the change of prior came before the post-hoc comparison (from the example answers, the writer and reader blocks, and the low runs), so it closes the evidence as a whole and leads into the circuit work of section 9. Draft:

> When I started this program, I assumed that the representations in neural networks were tangled by default, and that a concept we cared about would be hard to find unless we gave the model somewhere to put it. SCA was meant to give that signal early in training. So when we started D2.2, we designed a grammar with what we thought would be a hard-to-find latent concept: an operation that is never written down.
>
> That isn't quite what we found. The anchor settled on the example answers, and the post-hoc methods found the same evidence in models trained with no anchor at all. They all found a judgement about each example: does this one fit `difference`? Block 1 writes that judgement, and a later block reads it where the query is answered. We went looking for the op and found a quality of each example, something like how much it has of `difference` about it: a noun, just not the one we expected. That is close to how Nick Cammarata reads neurons in *Neural Networks Are Already Clean*: as distinctions that later parts of the network use, more often than the named things we bring to them.[1]
>
> We don't yet know whether the op itself is represented anywhere, or whether the answer is composed from those judgements by the blocks that read them. If it is, what we call `difference` may be a behaviour that emerges from them, unsatisfying though that may be; looking at what the readers write at the query would tell us. But the evidence is pretty cleanly represented: several arcs that share one end, on the axis we chose, and one other component, the direction of a non-fitting answer. If networks often hold evidence this cleanly on their own, the anchor is less about making a concept clean and more about deciding where it goes before training.
>
> [1] Cammarata argues from neurons in much larger vision and language models that this is the usual case. The essay is a position piece and still debated, and our result is one concept in one small synthetic model, so we offer it as related company rather than support.

Notes for the draft: it avoids "disentangled" and "superposition", which carry several meanings and invite questions the result can't answer. "Spread over several sites" is composition across the examples and slices, the kind of spread the essay counts as clean. The essay's caution is about fitting a replacement basis after training (a sparse autoencoder, say), which can lose the hints a neuron's odd groupings give. It credits training choices with cleaner neurons (InceptionV1 retrained "with more reasonable modern regularization" reads cleaner), and nothing found in it argues against a term at training time. The last sentence of the draft is where SCA and the essay meet: the model's own units are worth reading, and the anchor chooses where one of them sits.

9. What we would do next

- The ones to name: circuit design, perhaps in D2.3; gradient routing, a training-time method to compare with; and the D2.3 verification measurements.
- Where post-hoc methods are known to struggle, as the place an anchor might show an edge. Two kinds of case come up in the erasure literature, and both can appear in small models. One is a concept correlated with something else in the data, where a fitted direction takes some of the correlated feature with it (Kumar et al., 2022, on probing classifiers for concept removal). The other is a concept held nonlinearly, which a linear eraser leaves recoverable (Ravfogel et al., 2022, on kernel concept erasure). Citations to check before publishing. On *red*, an earlier check (ex-2.2.1) applied fitted directions to every operand state and found they cost the other colors about as much as they removed. We read that as a limit of unconditional linear erasure on a concept coded continuously (in the un-anchored model *red* is a corner of the color cube, so the fitted direction grades every color), and a gated edit has not been tried there; with a gate, the fitters would likely come close to the anchor on *red* too. The gated comparison on *red* belongs after the post, as one preregistered design with the position gate for the op. A confound built into the color grammar (say, `difference` contexts favoring certain colors) would test the correlated case directly.
- The comparison that tests what the anchor claims: a recipe transfer on fresh seeds, counting how often a fitted configuration chosen on earlier runs misses the slice against how often the recipe of record misses, preregistered after the post. The post can say it is planned.
- Briefly, if there is room: the position gate for the fitters; an edit-aware term that makes the readers depend on e_1 (backlog); the anchor and anti terms at the mid-block states; more ops anchored at once (design § future work).
- A caution on the edit-aware term: training on the removal measurement would put optimization pressure on the channel we use to check the edit, which is what the agenda's "gentle measurement" and the warning against training on interpretability tools ("the most forbidden technique") both advise against. The low-runs twins show how this could go: training can satisfy the anchor term while the readers lean on another channel. If the term is tried, its removal should be checked on a measurement it was never trained on.

Appendix, linked from the site

> The write bound, in plain terms. In the autoencoder the bound was geometric: removing the e_1 part of a state moves it by no more than that part. In a transformer the edited states feed later blocks, which could write e_1 back onto the states of other ops and move them further than the edit did. The write bound says how much e_1 the later blocks may put on those states: no more than the clean model already has there. Because we chose the axis, that limit can be read off the clean model before any edit, and the measured writes stay within it past the embedding, apart from a few sites at slice 1 on one run in eight (write-floor, fixed-effects).

The reference baseline (median over the quiet axes) and why Adam makes it fair; the absolute tolerance on the write bound; comparing conditions as two samples with a Welch t interval; perhaps the early α° screen and why it cannot see a low run (eng/determinism.md).

Alignment. How far a state points along e_1…

Compute overhead, a question readers will ask. On an L40S the terms add no step time we can tell from the noise between containers (2.01 ms with both terms against 1.97 without, with a spread of 1.90 to 2.11). By the compiler's count they add about 0.1% of the arithmetic and 3% of the bytes moved; the label draw runs in every variant, so that share is the two terms themselves, each about a third of it alone. Both terms read one coordinate per state, so their share shrinks as the model widens. The anchor term matters only on batches with a labelled context and could be skipped on the rest, while the anti term reads every state at about the same small cost. Anchored runs learn the task more slowly at first (a median of 140 epochs to reach 80% of the control's final score, against 104), and catch up by about epoch 250, on the same 400-epoch schedule (overhead probe, PR #345). If the post quotes the pace, it should show the training trajectories (task score by epoch, anchored against control, seeds 712–719, from the stored records); otherwise leave the sentence out.

Figures

The D2.1 post took its figures from a page of their own (D2.1 figures), drawn from published results and uploaded from the bundle. A D2.2 figures page in the same shape is the natural next step once the sections above settle. It would hold the grammar images of section 1, the pooling picture of section 2, the arc figure of section 7 drawn from stored checkpoints, the training trajectories if the appendix quotes the pace, beside the figures taken from the reports.

Sources still moving

- The post-hoc comparison (post-hoc compare) is merged, and how to present it is agreed (section 6).