

# Does the Greater-Than Circuit Generalize? A Reproducibility Study Across Models and Numerical Entailment

Vinay Setty

University of Stavanger  
vsetty@acm.org

## Abstract

We reproduce the canonical GPT-2-small “greater-than” circuit in its original stack and independently, then test how far it generalizes across eight models (124M–12B) and to numerical entailment. The reported numbers replicate, and the late-MLP structure recurs at matched relative depth well above a size-matched null. Sparse faithfulness is high everywhere too, but only when the edge budget scales with the model’s own graph; since these graphs differ 96-fold in size, a fixed budget makes faithfulness appear to collapse with scale. On numerical entailment, no circuit-bearing model passes the 0.6 gate, while the one model that solves the task well cannot be posed the circuit’s own task.

## 1 Introduction

Mechanistic interpretability (MI) increasingly produces claims of the form “model  $M$  solves task  $T$  using circuit  $C$ ,” supported by a single ablation or patching experiment on a single checkpoint. Such claims are widely cited and reused, as benchmarks for new discovery methods, as textbook examples, and as evidence that transformer internals are legible. Yet they are rarely re-verified independently. [Méloux et al. \(2026\)](#) recently show that circuit-finding is statistically fragile in general, across many circuits and methods. We take the complementary approach: rather than surveying many circuits shallowly, we take *one* canonical, benchmark-status circuit and examine it with the depth its status warrants, re-deriving it, varying what it depends on, and extending it to other models and tasks.

Our subject is the GPT-2-small “greater-than” circuit ([Hanna et al., 2023](#)): a small set of late MLPs and attention heads that the original paper shows compute a numerical inequality (e.g., completing “The war lasted from the year 1732 to the year 17\_\_” with years  $> 32$ ). The result is canonical in the sense that matters for reproducibility: it is

precise, quantitative, path-patching validated, and has been reused as a discovery-method benchmark by ACDC ([Conmy et al., 2023](#)) and edge attribution patching ([Syed et al., 2024](#); [Hanna et al., 2024](#)). Although prior work has reused this circuit heavily, to our knowledge no systematic study exists that (a) independently re-derives the reported numbers end to end, (b) quantifies how much of the circuit’s structure is an artifact of a single prompt template or a single model checkpoint, or (c) asks whether the same algorithmic motif appears in other models or supports the numerical-comparison behavior the circuit is presumed to serve, i.e., entailment over numerical claims. This work addresses four research questions:

- **RQ1:** Do the reported numbers reproduce, in the original code and an independent re-implementation?
- **RQ2:** How much of the circuit is structurally robust rather than a single-template or single-estimate artifact?
- **RQ3:** Does the same motif appear across models and scales, and does a sparse circuit transfer at matched relative depth?
- **RQ4:** Does the circuit support the downstream numerical entailment?

We make four contributions. **(1) Replication.** We reproduce the circuit in the original `rust_circuit` stack and independently in `TransformerLens` via edge attribution patching with integrated gradients (EAP-IG), matching the reference notebook to four decimal places (91.5% of full-model performance) and reproducing the original paper’s own Section 3.2 figures to within 0.6pp, and fix two bugs in the released code. **(2) Robustness.** The circuit’s structure is completely stable (recall 1.0) across five prompt templates, a size-matched random null, and the known less-than failure mode, while its sparse faithfulness varies considerably (0.56–0.70). **(3) Scale.** Extending

circuit-level analysis across model families and to 12B parameters—eight models over a  $96\times$  range (124M–12B), beyond the single-family,  $\leq 2.8B$  scope of prior work—relative-depth localization generalizes (bucketed IoU 0.65–0.81 against a late-layer null of 0.32–0.42), and the apparent collapse of faithfulness with scale is an artifact of a fixed edge budget: at matched sparsity every model is faithful (0.89–1.05), and the worst performers at 200 edges (pythia-6.9b,  $-0.17$ ; pythia-12b,  $-0.00$ ) are among the best once the budget scales with the graph. **(4) Numerical entailment.** No model carrying the circuit performs above chance, while the one model that solves the task cannot be posed the circuit’s own task; we temper this split, since format controls show most models cannot use the probe’s response format at all. Where testable, the circuit’s late MLPs are mildly counterproductive rather than necessary—and at reportable budgets, attribution returns a circuit structurally near-identical to a known-good one (IoU up to 0.82) while anticorrelating with the computation it should explain.

All code, configuration, and per-experiment results files are available at <https://github.com/iai-group/greater-than-repro>.

## 2 Related Work

**Numerical reasoning in language models.** Numerical and arithmetic reasoning in LMs has been probed at several levels: causal mediation analysis locates arithmetic subcomputations (Stolfo et al., 2023); a bag-of-heuristics view argues that arithmetic is solved by many overlapping shortcuts rather than one algorithm (Nikankin et al., 2025); numeracy is legible in embeddings (Wallace et al., 2019); and numeric attributes are linearly, monotonically encoded (Heinzerling and Inui, 2024). This line establishes that numerical competence exists inside LMs, but does not isolate a single reusable comparison circuit or test whether one transfers across models.

**Circuits across models.** Closest to our cross-model question, Lan et al. (2024) find shared sequence-continuation circuits across model families but explicitly do not study greater-than. In contrast, we ask the circuit-level cross-model question directly for one canonical circuit rather than a shared behavior across many, re-deriving it end to end, following its structure across scale, and asking whether it supports the numerical entailment it is

presumed to serve.

## 3 Background and Task

**Circuits, edges, and faithfulness.** A transformer’s forward pass can be written as a directed graph whose nodes are attention heads and MLP blocks and whose edges are the paths along which one component’s output reaches another’s input. A *circuit* is a subgraph of that graph: a subset of components and of the edges between them, claimed to carry out some behavior. An *edge budget* is the number of edges a discovery method is allowed to keep, so a smaller budget means a sparser and more interpretable claim. *Faithfulness* is how much of the model’s behavior a circuit reproduces when everything outside it is ablated, measured as the circuit’s metric over the full model’s metric with both baselined against the corrupted input. Faithfulness near 1.0 means the circuit accounts for the behavior, near 0 means it does not, and negative means that ablating the rest of the model drives behavior in the wrong direction, which is worse than ablating nothing at all.

**Task and data.** We use the year-span task of Hanna et al. (2023) on their released dataset, unchanged: 1,000 prompts of the form “The {noun} lasted from the year XXYY to the year XX”, where the century XX ranges over 11–17 and is not fixed at 17, and the start year YY ranges over 02–98. The model completes the prompt with a two-digit year YY’, scored with the *probability difference* metric  $\text{prob-diff} = \sum_{YY' > YY} p(YY') - \sum_{YY' \leq YY} p(YY')$ , i.e., the model’s aggregate preference for greater-than-consistent completions. The corrupted counterpart of each prompt replaces the start year with 01. Appendix B gives the per-century item counts.

**The claimed circuit.** Hanna et al. (2023) report that this behavior is computed by a small set of late MLPs (layers 8–11, chiefly 9 and 10) at the final position, fed by attention heads (a5.h1, a5.h5, a6.h9, a7.h10, a8.h8, a8.h11, a9.h1) that move the two-digit year forward via a soft, distributed comparison rather than a lookup table. Path patching validates the circuit at  $\sim 90\%$  recovery. They also establish necessity: giving circuit nodes the 01 dataset and everything outside it the normal dataset leaves the model unable to perform the task, at a probability difference of  $-36.6\%$ . We state the sign explicitly because this figure is easy to mis-

Table 1: Reproduction of the GPT-2-small greater-than circuit. Recovery = circuit perf. / baseline. Row 1 is as reported by Hanna et al. (2023), who state 89.5% recovery and give no edge count. Rows 2 and 4 use attribution at a 200-edge budget rather than the paper’s hand-curated path patching, which is why those circuits are larger; our re-implementation matches the reference notebook to four decimal places. Appendix A gives each row’s provenance in full.

| Source                      | Baseline | Circuit perf. | Recovery | Nodes / Edges |
|-----------------------------|----------|---------------|----------|---------------|
| Hanna et al. (2023)         | 0.817    | 0.727         | 0.895    | 11 / —        |
| Reference notebook (EAP-IG) | 0.814    | 0.745         | 0.915    | 29 / 150      |
| rust_circuit repro.         | 0.816    | 0.727         | 0.890    | 11 / —        |
| Our TL / EAP-IG             | 0.814    | 0.745         | 0.915    | 29 / 150      |
| EAP (variant)               | 0.814    | 0.656         | 0.806    | 35 / 165      |
| clean-corrupted (variant)   | 0.814    | 0.686         | 0.842    | 31 / 151      |

read as a 36.6% reduction. The original paper also documents cracks: the circuit fails to invert on a less-than variant and over-generalizes to implausible contexts.

**Discovery methods.** We use EAP (Syed et al., 2024) and its integrated-gradients variant (EAP-IG; Hanna et al., 2024), which approximate patching effect without ACDC’s (Conmy et al., 2023) combinatorial search. EAP-IG better tracks faithfulness (Hanna et al., 2024), so it is our primary method (§4–§7); EAP and clean/corrupted patching serve as robustness checks.

## 4 Reproduction (RQ1)

**Original stack.** Running the released rust\_circuit codebase end to end on GPT-2-small (Radford et al., 2019), we reproduce Hanna et al.’s Section 3.2 evaluation closely: a full-model probability difference of 0.8164 against their reported 81.7%, a circuit probability difference of 0.7270 against their 72.7%, and a recovery of 88.99% against their 89.5%. Cutoff sharpness and the circuit-discovery script’s own figures agree similarly closely (Appendix A).

**Independent re-implementation.** We separately re-implement the task and circuit discovery in TransformerLens (Nanda and Bloom, 2022) using EAP-IG, without reference to rust\_circuit internals. Our baseline prob-diff is 0.8143, and the discovered 29-node / 150-edge circuit achieves 0.7449, a recovery of 91.5%, matching the released reference notebook to four decimal places. Table 1 summarizes all three settings alongside two discovery-method variants (plain EAP and a clean-corrupted patching baseline), which recover somewhat less

(80.6% and 84.2%) but agree qualitatively.

**Node-level agreement.** Beyond aggregate prob-diff, we ask whether EAP-IG rediscovers the *same components* the original hand-curated circuit specifies. It does: recall against both the “core” and “full” hand-curated node sets is 1.0: every hand-identified head and MLP is found. Precision is lower (0.393 core / 0.536 full), because automated discovery additionally includes early-layer components (chiefly embedding-adjacent nodes) that the manual circuit omits; we take this as evidence that the hand-curated circuit is a conservative subset of what attribution methods surface, not that the two disagree.

**Which stack carries the rest of the paper.** We report both for RQ1 but use the TransformerLens re-implementation thereafter, because rust\_circuit is unmaintained and GPT-2-specific while RQ3 and RQ4 need attribution across eight checkpoints plus Qwen2.5-7B, spanning three families. The two stacks agree on GPT-2-small, which is what licenses using the more portable one (Appendix A).

**Two bugs in the released code.** Running the original repository end to end surfaced two bugs, reported here as reproducibility observations rather than criticism: the reverse-patching cell in big\_ds\_experiments.py reuses a sampler group built for  $N=200$  examples against  $N=10,000$  tensors (a shape-mismatch crash at full scale), and, once fixed naively, accumulated circuit references cause an out-of-memory failure. Neither is exercised by the release’s verification notebook.

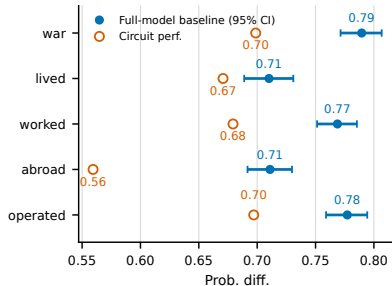


Figure 1: Template robustness for the year-span task across five prompt templates. Points/bars show the full-model baseline prob-diff with bootstrap 95% CIs (0.71–0.79 across templates); overlaid markers show the discovered circuit’s recall (1.0 on every template, i.e., fully template-robust structure) and faithfulness (0.56–0.70, weakest on the “abroad” template). Structural recovery is template-invariant; quantitative faithfulness is not.

## 5 Statistical Robustness (RQ2)

We run three checks that a single prompt template and point estimate cannot: template robustness, a random-circuit null, and less-than inversion.

**Template robustness.** We rewrite the year-span prompt into five surface variants. All five preserve the two year mentions, their order, and the final truncated century, and vary only the surrounding clause: the subject, the verb of duration, and the preposition pair (“from ... to” versus “from ... until”). The original “war” template and its four paraphrases are given verbatim in Appendix B. Figure 1 shows, per template, the bootstrap 95% CI on the baseline prob-diff (0.71–0.79) alongside circuit recall and faithfulness. Structure is completely template-robust (recall 1.0 on every template); faithfulness varies considerably (0.56–0.70), weakest on “abroad”.

**Method agreement.** On the war template, our three discovery methods (EAP-IG, clean-corrupted, EAP) agree reasonably well in circuit faithfulness (0.699, 0.663, 0.658)<sup>1</sup>, so our headline numbers are not an artifact of one attribution method’s idiosyncrasies. This is consistent with the observation of Zhang and Nanda (2024) that patching-based metrics can be sensitive to method choice.

<sup>1</sup>gpt2-small’s faithfulness appears as three values in this paper because they measure different things: 0.699 here, on the war template alone; 0.885 at the 200-edge budget of Table 2; and 0.915 for the 150-edge reference-notebook circuit of Table 1.

**Random-circuit null.** We sample 25 size-matched random circuits and prune each with the same procedure used for the real circuit. All 25 prune to the empty graph; no random circuit of this size supports any behavior. This places the real circuit at the 100th percentile. A null where every sample collapses to zero is degenerate, however, and licenses only a percentile statement, not an effect size; a connectivity-preserving null (degree- and depth-matched subgraphs) is needed for a stronger claim.

**Less-than inversion.** Hanna et al. (2023) report that their circuit fails to invert on a less-than variant; we reproduce this quantitatively. Full-model prob-diff on the inverted task is  $-0.79$  and the circuit’s is  $-0.70$ , both strongly wrong-directioned, with the circuit tracking the broken full-model behavior closely. The circuit is a template-and-direction-specific mechanism, not a general comparator, and the original authors’ own failure mode reproduces exactly.

## 6 Cross-Model Generalization (RQ3)

The original circuit was found in one 124M-parameter model. Tigges et al. (2024) study greater-than circuits across Pythia scale and training checkpoints using EAP-IG, the same method we use, reporting Jaccard similarity of circuit node sets and a positive correlation between circuit size and model scale. Their analysis is confined to a single model family, which they note leaves open whether their findings are family-specific, and they omit Pythia-6.9b and 12b from circuit finding on cost grounds. We extend the picture in four ways: across model families, to those two largest checkpoints, against a size-matched late-layer null for depth-bucketed overlap, and by normalizing edge budgets to graph size.

**Behavioral gate.** Before asking about circuits we check that each candidate can perform the year-span task at all, gating on prob-diff  $\geq 0.3$ . All eight pass, from 0.431 (pythia-160m) to 0.854 (pythia-1.4b; Table 2, and Biderman et al., 2023). Qwen2.5-7B (Qwen Team, 2024) *cannot be posed the task at all*: its tokenizer splits multi-digit numbers into per-digit tokens rather than the two-digit chunks the task assumes, so neither the prompt nor the metric is well defined for it. That is a finding in itself about tokenizer-dependent task portability, and Appendix A records a related pitfall that silently

Table 2: Cross-model year-task circuits. Edge counts span  $96\times$ , so absolute budgets are incomparable: *Faith.*@200 uses a fixed 200-edge budget, *Faith.*@match 0.616% of each model’s own graph (the gpt2-small equivalent). IoU uses  $K=11$  depth deciles; *Null* is the mean over 1,000 size-matched random late-layer node sets. † pythia-160m’s above-1.0 value is an artifact of its noisy 0.431 baseline.

| Model        | Layers | Edges     | Gate  | Faith.@200 | Match $n$ | Faith.@match | Late-MLP $n$ | IoU (Null)    |
|--------------|--------|-----------|-------|------------|-----------|--------------|--------------|---------------|
| gpt2-small   | 12     | 32,491    | 0.791 | 0.885      | 200       | 0.885        | 6            | 1.000 (—)     |
| gpt2-medium  | 24     | 231,877   | 0.834 | 0.699      | 1,427     | 0.924        | 12           | 0.714 (0.365) |
| pythia-160m† | 12     | 32,347    | 0.431 | 1.051      | 199       | 1.051        | 6            | 0.810 (0.424) |
| pythia-410m  | 24     | 231,493   | 0.824 | 0.435      | 1,425     | 0.944        | 12           | 0.714 (0.359) |
| pythia-1.4b  | 24     | 231,493   | 0.854 | 0.703      | 1,425     | 0.925        | 11           | 0.762 (0.378) |
| pythia-2.8b  | 32     | 1,591,857 | 0.764 | 0.079      | 9,799     | 0.959        | 15           | 0.714 (0.339) |
| pythia-6.9b  | 32     | 1,591,857 | 0.797 | -0.171     | 9,799     | 1.024        | 12           | 0.650 (0.328) |
| pythia-12b   | 36     | 3,131,263 | 0.816 | -0.004     | 19,275    | 0.981        | 16           | 0.650 (0.324) |

corrupts results rather than crashing.

**Localization generalizes.** We run EAP-IG discovery independently per model and compare circuits by relative depth, so models of different depths are comparable (Table 2). In every model the circuit concentrates late MLPs at matched relative depth, with depth-bucketed overlap (IoU) against gpt2-small of 0.65–0.81. Because coarse buckets and late-MLP-heavy circuits could make that trivial, we compare against a null of size-matched random node sets drawn from each model’s own late layers: the null means 0.32–0.42 with a 95th percentile of at most 0.53, placing every observed IoU at the 100th percentile of its null and 5.9–6.8 standard deviations above its mean (Appendix C). The late-MLP, relative-depth motif Hanna et al. (2023) describe for GPT-2-small is not an artifact of that one checkpoint.

**Sparse faithfulness must be measured against graph size.** At a fixed 200-edge budget faithfulness does fall steeply with scale, from 0.885 (gpt2-small) to  $-0.171$  (pythia-6.9b), the largest models scoring worse than ablating nothing at all. But 200 edges is not a comparable ask: these graphs differ in size  $96$ -fold, so the same count is 0.616% of gpt2-small’s graph and 0.006% of pythia-12b’s, asking the largest model for a circuit roughly 100 times sparser. Given instead the budget matching gpt2-small’s canonical 200 edges as a *fraction* of its own graph, every model is faithful (0.885–1.051) and the ordering inverts, the three checkpoints that look worst at 200 edges becoming three of the four best (Table 2, Figure 2). The finding is therefore not that circuits degrade with scale but that *absolute edge budgets do not port across model sizes*: comparing models at a fixed edge count measures graph size more than circuit quality.

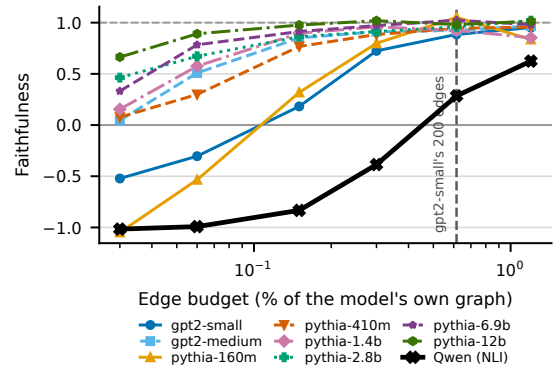


Figure 2: Circuit faithfulness vs. edge budget as a percentage of each model’s own graph; dashed rule at 0.616%, the fraction gpt2-small’s 200-edge circuit represents. Curves largely coincide, so the apparent scale-dependent collapse arises only when edge count, not fraction, is held fixed. Qwen entailment (Section 7) turns positive only past the matched fraction.

One caveat cuts the other way: pythia-160m’s circuit *exceeds* its full model’s faithfulness (1.05 at matched sparsity), which we read as an artifact of its low, noisy baseline (0.431) and flag in Table 2 so it is not mistaken for the best-generalizing model.

## 7 Numerical Entailment (RQ4)

If the greater-than circuit computes a reusable numerical comparison, it should support natural-language entailment over numerical claims. We probe this with a synthetic comparison-NLI item, “Premise: {name} has {x} {noun}. Hypothesis: {name} has more than {y} {noun}. ... Answer:”, scored on the single-token Yes/No logit difference, which is tokenization-robust across every family we study ( $N=500$  items per model and condition; Appendix D gives construction details, and all intervals in this section are bootstrap intervals over

evaluation items).

**What we observe.** Applying §6’s gate, the split is clean. All six models we test on this probe sit at chance (0.500–0.576, zero- and two-shot alike; pythia-6.9b and 12b were not run on the entailment probe), while Qwen2.5-7B, the one model that cannot be posed the year-span task at all, solves the probe perfectly zero-shot (1.000, logit-difference 3.83, 95% CI [3.71, 3.94]). Taken at face value the models that carry the circuit do not do well at numerical entailment, and the model that does well is the one we could not test for the circuit.

**Why we do not read that as a dissociation.** Two considerations temper it, and both bound what this experiment can establish. First, a model at chance on a Yes/No probe may be unable to use the response format at all rather than declining to recruit a comparison circuit. A digit-free control sharing the probe’s scaffolding and scoring exactly (Table 3) shows this is the operative limit for most of them: six of the eight score 0.492–0.502 when the task is reduced to asking whether a hypothesis restates the premise’s attribute, which is strictly easier than any numerical comparison, so their numerical accuracy cannot be attributed to numeracy specifically. Only pythia-2.8b and Qwen2.5-7B clear the floor convincingly,<sup>2</sup> and in both an ordinal control over alphabet positions is no easier than the numerical version (Table 3). Establishing whether the circuit is recruited for numerical entailment therefore needs models that can engage this response format, or a probe these models can engage; we leave that to future work.

Second, Qwen’s per-digit tokenization makes the year-span task unposable for it, so we did not test it for a greater-than circuit and cannot conclude that it lacks one. Confirming the other half of the observation would require reformulating the task at the digit level, which we also leave to future work (§9). The contrast between the smaller models and Qwen additionally confounds model family, parameter count, and tokenizer at once, so it isolates no single cause.

**Necessity does not transfer.** Pythia-2.8b is the only model that clears the floor and still falls short of its own ceiling, so it is the only one in which the circuit’s causal role is testable. Ablating its late MLPs drops accuracy to chance, but so does

<sup>2</sup>pythia-12b (0.550) sits about two standard errors above chance on 500 items, too weak to interpret either way.

Table 3: Format controls for the entailment probe: *Identity* is lexical identity, *Ordinal* a letter-order comparison, both digit-free and sharing the numerical probe’s scaffolding and scoring; *Numerical* is the best of zero- and two-shot accuracy. Chance is 0.5. A model at chance on *Identity*, the easiest of the three, cannot use the response format at all.

| Model           | Identity | Ordinal | Numerical |
|-----------------|----------|---------|-----------|
| gpt2-small      | 0.500    | 0.504   | 0.504     |
| gpt2-medium     | 0.502    | 0.500   | 0.504     |
| pythia-160m     | 0.500    | 0.500   | 0.504     |
| pythia-410m     | 0.500    | 0.500   | 0.518     |
| pythia-1.4b     | 0.500    | 0.500   | 0.502     |
| pythia-2.8b     | 0.650    | 0.518   | 0.576     |
| pythia-6.9b     | 0.492    | 0.506   | —         |
| pythia-12b      | 0.550    | 0.476   | —         |
| Qwen/Qwen2.5-7B | 1.000    | 0.986   | 1.000     |

an equally-sized random-MLP ablation: no differential necessity. Node ablation is the wrong test, however, since what the circuit contains is the *edges into* those MLPs, as one reviewer contains. At the edge level the result inverts: removing the 37 circuit edges entering those MLPs *raises* accuracy from 0.576 to 0.624 [0.582, 0.666] while size-matched random-edge ablations stay at baseline (0.554–0.574), and removing all 9,813 edges into them raises it to 0.728 [0.689, 0.767]. The year-task late MLPs are mildly counterproductive for entailment rather than merely unnecessary, consistent with the naturalistic result below. We do not read the 0.048 circuit-edge gain on 500 items as a strong effect.

### Structural overlap without functional faithfulness.

Running EAP-IG on Qwen2.5-7B directly on the entailment task yields a circuit *structurally* near-identical to the year-span circuits: depth-bucketed IoU 0.77–0.82, recall roughly 0.9, concentrated in 11 late MLPs, the same motif as §6. At sparse budgets it is *anti-faithful*, from  $-1.02$  at 200 edges to  $-0.77$  at 1600, driving behavior in the wrong direction rather than failing to recover it.

The budget caveat of Section 6 applies here too: Qwen’s graph has 935,747 edges, so 1600 of them is 0.17% against gpt2-small’s 0.616%. Swept at matched fractions the circuit recovers monotonically and turns positive, reaching  $+0.28$  at the matched budget (Figure 2), so anti-faithfulness here is again confined to undersized budgets.

Two things survive. The entailment circuit is markedly less sparse-recoverable than the year circuits, reaching 0.28 at matched sparsity where they

reach 0.89–1.05 and needing roughly 0.5% of its graph merely to stop anticorrelating with the behavior. And, independently of budget, at the sparsity a circuit paper would actually report, attribution returned a structure near-identical to a benchmark circuit while anticorrelating with the computation it should explain: a reader given only that circuit and its overlap would have drawn a confident and wrong conclusion. Structural overlap is not by itself evidence of shared mechanism, echoing the statistical fragility of circuit discovery (Méloux et al., 2026) and the interpretability-illusion literature (Makelov et al., 2024).

**Naturalistic checks.** Qwen’s perfect synthetic accuracy does not transfer. On EQUATE (Ravichander et al., 2019) it scores 0.602 (RTE-Quant) and 0.553 (AWPNLI, which requires arithmetic); on NumPert (Aarnes and Setty, 2025) it scores 0.620 and is robust to label-preserving rewrites ( $\pm 0.03$ ) but collapses to 0.009 under digit masking and to chance under sign flips. The synthetic-NLI circuit fares no better: on 167 length-matched NumPert pairs its faithfulness is  $-0.01$ , and zero-ablating its late MLPs is not selectively harmful on EQUATE while *improving* NumPert accuracy ( $0.62 \rightarrow 0.70$ ) where a size-matched control collapses it (0.48). Sparse-circuit conclusions drawn from templated probes do not carry over to naturalistic numerical inference (Appendix D).

## 8 Discussion

What survives of the “established” greater-than circuit? **RQ1:** the numbers reproduce, in the original codebase and independently, to four decimals; the two bugs we fixed are process observations, not errors in the claim. **RQ2:** node identity is completely template-robust while sparse faithfulness is not (0.56–0.70). This pattern recurs throughout: structural identity is more robust than quantitative faithfulness. **RQ3:** the circuit generalizes in structure. Relative-depth localization transfers across a  $96\times$  parameter range at IoU values far above a late-layer null, and sparse faithfulness is high everywhere once the edge budget is normalized by graph size; measured at a fixed edge count it instead appears to collapse with scale, which is a property of the measurement rather than of the circuits. **RQ4:** the models carrying the circuit do not do well at numerical entailment, and the model that does cannot be posed the circuit’s task. We report that split but stop short of calling it a dis-

sociation, because most of these models fail an equally easy non-numerical probe in the same format, which leaves the question open rather than answered; confirming it needs models that can engage the format, and testing Qwen for the circuit needs a tokenizer-independent formulation of the task. Where the circuit’s role is testable, in pythia-2.8b, its late MLPs show no differential necessity and at the edge level are mildly counterproductive. Structural resemblance to a benchmark circuit can co-occur with anti-faithful behavior at the sparsity a circuit paper would report.

## 9 Conclusions and Future Work

We re-derived the GPT-2 greater-than circuit in its original stack and in an independent re-implementation, checked how it holds up under template variation, a random null, and its known failure mode, and followed it across eight models spanning 124M to 12B parameters and a downstream numerical-entailment task. The original numbers hold, and the circuit’s late-MLP structure recurs at matched relative depth across a  $96\times$  parameter range, well above a size-matched null.

Two measurement choices drive what a circuit study concludes. Fixed edge budgets compared across model sizes manufacture a scaling collapse that vanishes under size-normalized budgets — our graphs span a 96-fold range. And a behavioural null is uninterpretable until the response format is controlled: models that seem unable to do numerical entailment also fail an equally hard non-numerical task in the same format. Circuit papers should report faithfulness against *size-normalized* budgets, gate circuit claims on task behavior, control response format before reading a null as mechanistic evidence, audit tokenization before porting tasks across families, and release code that runs end to end at scale.

Whether the greater-than circuit serves numerical entailment is unresolved, since the models that carry it cannot use our probe’s format; settling it needs instruction-tuned models at comparable scale or a probe they can engage. And whether Qwen2.5-7B has the circuit is untested, not answered: its tokenizer emits one token per digit, so the task as posed is undefined. Scoring the tens digit instead would be tokenizer-independent and comparable across families. Future work should add seed-level replication, a connectivity-preserving null, and circuit discovery on naturalistic data.

## Limitations

Our discovery experiments use a single random seed per model/task combination; the confidence intervals we report are bootstrap intervals over evaluation items, not over discovery-seed variation, so we cannot rule out that a different seed would surface a somewhat different sparse circuit (though §5’s template-robustness results suggest the qualitative structure would be stable). We restrict circuit discovery to EAP and EAP-IG-family attribution methods; we do not run a path-patching cross-check beyond gpt2-small, so our cross-model and entailment circuits have not been validated by a second, independent discovery method. Our necessity probe on pythia-2.8b tests a weak, above-chance-only signal (0.576 accuracy), and a null result on a weak signal is less informative than one on a strong effect. We exclude Llama-3-8B from the cross-model study because its weights are gated and unavailable to us under our compute access.

Our format controls (§7) show that six of the eight year-circuit models cannot use the entailment probe’s Yes/No response format on even a lexical-identity task. We therefore report the entailment split we observe but do not read it as evidence about circuit use: for those models the question remains open rather than answered negatively, and we cannot attribute their failure to numeracy specifically. We likewise did not test Qwen2.5-7B for a greater-than circuit, so its absence there is untested rather than established. and addressing it would need either instruction-tuned models of comparable scale or a probe whose response format these models can use. Our circuit *discovery* for entailment uses a synthetic, template-based probe; the naturalistic checks (§7) are behavioral and ablation-based, plus a single positional evaluation enabled by NumPert’s length-preserving perturbations; broader naturalistic circuit discovery remains open. The ablation controls include early MLPs, which are generically disruptive to remove, so our “necessary but not selective” readings are conservative and tentative.

Finally, our random-circuit null (§5) is degenerate (all samples collapse to the empty graph) and licenses only a percentile statement, not an effect size; a connectivity-preserving null is needed for a stronger quantitative robustness claim.

## Ethics Statement

This work reproduces and extends existing interpretability analyses on publicly released, pretrained language models (GPT-2, Pythia, Qwen2.5) using synthetic numerical prompts; it does not involve human subjects, crowdsourced annotation, or personally identifiable data. We do not foresee dual-use or harm concerns beyond those already associated with the base models we study, all of which are publicly available research artifacts. We view the reproducibility and bug-reporting components of this work as a positive-sum contribution to the interpretability community, and we report the code issues we found in the original release respectfully and with the intent of helping, not discrediting, the original authors, whose result we substantially reproduce.

## References

- Peter Røysland Aarnes and Vinay Setty. 2025. NumPert: Numerical perturbations to probe language models for veracity prediction. In *Proceedings of the IJCNLP-AACL 2025 Student Research Workshop*, IJCNLP-AACL '25, pages 78–95.
- Stella Biderman, Hailey Schoelkopf, Quentin Gregory Anthony, Herbie Bradley, Kyle O'Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, USVSN Sai Prashanth, Edward Raff, Aviya Skowron, Lintang Sutawika, and Oskar van der Wal. 2023. Pythia: A suite for analyzing large language models across training and scaling. In *Proceedings of the 40th International Conference on Machine Learning*, ICML '23, pages 2397–2430.
- Arthur Conmy, Augustine N. Mavor-Parker, Aengus Lynch, Stefan Heimersheim, and Adrià Garriga-Alonso. 2023. Towards automated circuit discovery for mechanistic interpretability. In *Advances in Neural Information Processing Systems 36*, NeurIPS '23.
- Michael Hanna, Ollie Liu, and Alexandre Variengien. 2023. How does GPT-2 compute greater-than?: Interpreting mathematical abilities in a pre-trained language model. In *Advances in Neural Information Processing Systems 36*, NeurIPS '23.
- Michael Hanna, Sandro Pezzelle, and Yonatan Belinkov. 2024. Have faith in faithfulness: Going beyond circuit overlap when finding model mechanisms. In *Proceedings of the First Conference on Language Modeling*, COLM '24.
- Benjamin Heinzerling and Kentaro Inui. 2024. Monotonic representation of numeric attributes in language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, ACL '24, pages 175–195.
- Michael Lan, Philip Torr, and Fazl Barez. 2024. Towards interpretable sequence continuation: Analyzing shared circuits in large language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, EMNLP '24, pages 12576–12601.
- Aleksandar Makelov, Georg Lange, Atticus Geiger, and Neel Nanda. 2024. Is this the subspace you are looking for? an interpretability illusion for subspace activation patching. In *Proceedings of the Twelfth International Conference on Learning Representations*, ICLR '24.
- Maxime M  loux, Fran  ois Portet, and Maxime Peyrard. 2026. Mechanistic interpretability as statistical estimation: A variance analysis. In *Proceedings of the 43rd International Conference on Machine Learning*, ICML '26.
- Neel Nanda and Joseph Bloom. 2022. TransformerLens. <https://github.com/TransformerLensOrg/TransformerLens>. Software.
- Yaniv Nikankin, Anja Reusch, Aaron Mueller, and Yonatan Belinkov. 2025. Arithmetic without algorithms: Language models solve math with a bag of heuristics. In *Proceedings of the Thirteenth International Conference on Learning Representations*, ICLR '25.
- Qwen Team. 2024. [Qwen2.5 technical report](#). Preprint, arXiv:2412.15115.
- Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. Technical report, OpenAI.
- Abhilasha Ravichander, Aakanksha Naik, Carolyn Rose, and Eduard Hovy. 2019. EQUATE: A benchmark evaluation framework for quantitative reasoning in natural language inference. In *Proceedings of the 23rd Conference on Computational Natural Language Learning*, CoNLL '19, pages 349–361.
- Alessandro Stolfo, Yonatan Belinkov, and Mrinmaya Sachan. 2023. A mechanistic interpretation of arithmetic reasoning in language models using causal mediation analysis. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, EMNLP '23, pages 7035–7052.
- Aaquib Syed, Can Rager, and Arthur Conmy. 2024. Attribution patching outperforms automated circuit discovery. In *Proceedings of the 7th BlackboxNLP Workshop: Analyzing and Interpreting Neural Networks for NLP*, BlackboxNLP '24, pages 407–416.
- Curt Tigges, Michael Hanna, Qinan Yu, and Stella Biderman. 2024. LLM circuit analyses are consistent across training and scale. In *Advances in Neural Information Processing Systems 37*, NeurIPS '24, pages 40699–40731.
- Eric Wallace, Yizhong Wang, Sujian Li, Sameer Singh, and Matt Gardner. 2019. Do NLP models know numbers? probing numeracy in embeddings. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, EMNLP-IJCNLP '19, pages 5307–5315.
- Fred Zhang and Neel Nanda. 2024. Towards best practices of activation patching in language models: Metrics and methods. In *Proceedings of the Twelfth International Conference on Learning Representations*, ICLR '24.

## Appendix

### A Implementation and Reproducibility Details

**Code and artifacts.** All code, configuration, SLURM scripts, and per-experiment JSON results are available at <https://github.com/iai-group/greater-than-repro>; every number in this paper is traceable to a committed results file, and the repository’s README maps each script to the table or figure it produces. Experiments ran on NVIDIA A100 GPUs under SLURM; each experiment completes within a few GPU-hours.

**Original-stack reproduction (RQ1).** We built `rust_circuit` from source with Python 3.10 and torch 2.4.1+cu121 and ran the released `circuit_discovery.py` unchanged. Two fixes were required to run the remaining released scripts end to end: (1) the reverse-patching loop in `big_ds_experiments.py` reuses a batch-size-200 computation graph with full  $N=10,000$  tensors and crashes; we batched it to mirror the forward loop; (2) the plotting path needs `plotly<6` with `kaleido==0.2.1` and two output directories the scripts assume but do not create. The reverse loop also exhausts GPU memory unless the forward loop’s graphs are freed first.

**Original-stack figures in full.** Our end-to-end run of the released `rust_circuit` stack reproduces Hanna et al. (2023)’s Section 3.2 evaluation as follows, ours first and theirs second: full-model probability difference 0.8164 (SD 0.1930) against 81.7% (SD 19.3%); circuit probability difference 0.7270 against 72.7%; recovery 88.99% against 89.5%; cutoff sharpness  $0.0595 \rightarrow 0.0778$  against  $6.0\% \rightarrow 8\%$ . The release’s circuit-discovery script separately prints its whole-circuit probability difference (0.7261), its ratio to that script’s own baseline of 0.8189 (88.67%), and its ratio to an earlier patched figure (98.56%), the last matching the 98.3% of their Appendix B. These three are one circuit measured three ways rather than three circuits, which is worth stating because only the middle value is a recovery.

**Independent re-implementation.** Our TransformerLens stack uses Python 3.11, TransformerLens 2.15.4, the EAP-IG library of Syed et al. (2024) as maintained by Hanna et al. (2024), torch 2.4 (CUDA 12.1 wheels), and transformers 4.x. The full dependency lock ships with

the code; on older glibc systems (ours is 2.27), pinning `pyarrow<17`, `wandb<0.18`, and `datasets<3` avoids source builds and incompatible wheels. GPU nodes without internet access require pre-syncing the environment and setting `HF_HUB_OFFLINE=1`.

**Circuit discovery settings.** Attribution uses EAP-IG on inputs with 5 integrated-gradient steps; plain EAP and clean-corrupted patching serve as method controls. The metric is prob-diff for the year task and gold-answer logit-diff for entailment. Circuits are the top- $n$  edges by absolute score, pruned to a connected subgraph; we sweep  $n \in \{100, 200, 400, 800\}$  cross-model and up to 1600 for the Qwen entailment circuit. Models enter the cross-model study only if full-model prob-diff is at least 0.3. Faithfulness is circuit metric over full-model metric, both corrupted-baselined.

**Pitfalls we hit.** Three issues silently corrupt results rather than crash, so we flag them explicitly. First, TransformerLens tokenizers prepend BOS for some model families (Pythia, Qwen); any string-level token arithmetic must pass `add_special_tokens=False`, or clean and corrupted sequences misalign by one position. Second, attribution on grouped-query-attention models (Qwen2.5) requires ungrouping key-value heads before building the edge graph. Third, corruption pairs must match in token length exactly; per-digit tokenizers break pairings that are valid for two-digit-chunk tokenizers, which is also why the year task cannot be posed to Qwen2.5. Qwen attribution at 7B scale required a batch size of 2.

**Data.** The year task uses the released `greater_than_data.csv` plus four paraphrase templates; the synthetic entailment probe poses two-number comparisons as premise and hypothesis with Yes/No answers, with corruptions that flip the gold label while preserving token length. Naturalistic evaluations use EQUATE (RTE-Quant and AWPNI) and NumPert; the 167 positional pairs come from NumPert’s random-replacement perturbation restricted to equal-digit-count substitutions.

### B Task Data and Templates

**Year-span dataset.** We use Hanna et al. (2023)’s released `greater_than_data.csv` unchanged. It contains 1,000 prompts drawn from 34 event nouns, with centuries distributed 11 (73 items), 12 (168),

Table 4: NLI gate: zero-shot and two-shot accuracy per model. Only Qwen2.5-7B passes the gate (highlighted); GPT-2 and Pythia models cluster at chance ( $\approx 0.5$ ).

| Model           | Zero-shot acc. | Two-shot acc. | Pass       |
|-----------------|----------------|---------------|------------|
| gpt2-small      | 0.504          | 0.500         | No         |
| gpt2-medium     | 0.504          | 0.500         | No         |
| pythia-160m     | 0.500          | 0.504         | No         |
| pythia-410m     | 0.500          | 0.518         | No         |
| pythia-1.4b     | 0.502          | 0.500         | No         |
| pythia-2.8b     | 0.536          | 0.576         | No         |
| Qwen/Qwen2.5-7B | 1.000          | —             | <b>Yes</b> |

13 (146), 14 (144), 15 (154), 16 (149), and 17 (166); start years span 02–98. The corrupted counterpart of every prompt replaces the start year with 01, which is the corruption the original release uses. Because the century varies, the task is not specific to the 1700s, and the metric is defined over the 100 two-digit year tokens rather than over absolute years.

**Prompt templates.** The five surface variants of §5. Each is instantiated with a noun, a century, and a start year exactly as in the main task; only the “war” template shares the original release’s subject and verb.

- **war** (original): “The {noun} lasted from the year {century}{yy} to the year {century}”
- **lived**: “The couple lived there from the year {century}{yy} until the year {century}”
- **worked**: “She worked at the company from the year {century}{yy} to the year {century}”
- **abroad**: “He was abroad from the year {century}{yy} until the year {century}”
- **operated**: “The mine operated from the year {century}{yy} to the year {century}”

The four paraphrases vary the subject (“The couple”, “She”, “He”, “The mine”), the verb of duration (“lived there”, “worked at the company”, “was abroad”, “operated”), and the preposition pair (“to” for war, worked, and operated; “until” for lived and abroad), while holding the numeric structure and the final truncated century fixed. Paraphrase templates draw from the 14-noun subset of the released noun list given in our `gtlib/datasets.py`.

## C Cross-Model Details (RQ3)

**Depth bucketing and the IoU null.** Nodes are bucketed by relative depth,  $\text{layer}/(n_{\text{layers}} - 1)$  rounded to one decimal, giving  $K=11$  deciles held

constant across every model regardless of layer count and keyed by component type (so  $m@0.9$  and  $a@0.9$  are distinct). The null draws 1,000 random node sets per model, size-matched to that model’s discovered circuit and sampled only from its own late layers (layer fraction  $\geq 0.5$ ), which is the regime the real circuits occupy and therefore the conservative comparison. Observed IoU against gpt2-small, followed by the null mean, its standard deviation, and the observed value’s distance above the null mean in standard deviations, is 0.714 (0.365, 0.060,  $5.9\sigma$ ) for gpt2-medium, 0.810 (0.424, 0.063,  $6.1\sigma$ ) pythia-160m, 0.714 (0.359, 0.057,  $6.2\sigma$ ) pythia-410m, 0.762 (0.378, 0.062,  $6.2\sigma$ ) pythia-1.4b, 0.714 (0.339, 0.055,  $6.8\sigma$ ) pythia-2.8b, 0.650 (0.328, 0.053,  $6.0\sigma$ ) pythia-6.9b, and 0.650 (0.324, 0.050,  $6.5\sigma$ ) pythia-12b. Every observed value is at the 100th percentile of its null.

**Absolute edge budgets.** Table 2 and Figure 2 report budgets as a fraction of each model’s own graph, for the reasons given in §6. We retain the absolute view here because the discrepancy between the two is itself the reproducibility lesson.

At a fixed 200 edges, faithfulness is 0.885 (gpt2-small), 0.699 (gpt2-medium), 1.051 (pythia-160m), 0.435 (pythia-410m), 0.703 (pythia-1.4b), 0.079 (pythia-2.8b),  $-0.171$  (pythia-6.9b), and  $-0.004$  (pythia-12b), a pattern that reads as a collapse with scale steep enough to turn negative. At budgets matched to 0.616% of each graph it is 0.885, 0.924, 1.051, 0.944, 0.925, 0.959, 1.024, and 0.981 respectively, a pattern with no scale trend at all. The two views are computed from the same attribution scores on the same cached graphs and differ only in where the top- $n$  threshold falls.

The practical consequence for circuit papers is that an edge budget is only interpretable alongside

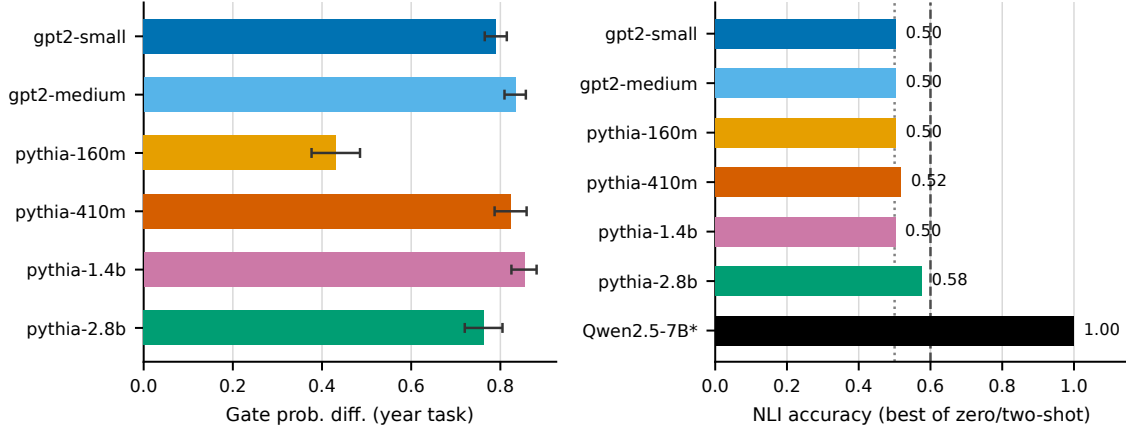


Figure 3: Year-span gate performance (left, prob-diff) against NLI accuracy (right; dotted line = chance 0.5, dashed = gate 0.6). All six year-circuit models tested on the probe cluster at NLI chance and Qwen2.5-7B (\*, year task unposable) is the only model above the gate. The panel invites reading a dissociation between circuit possession and circuit use, but Table 3 shows that for six of the eight left-hand models the NLI result reflects inability to use the response format at all.

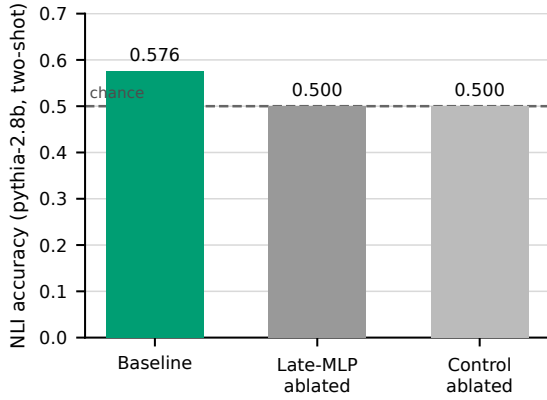


Figure 4: Node-level necessity on pythia-2.8b. Ablating the year circuit’s late MLPs collapses accuracy to chance, but so does an equally-sized random-MLP control: no differential necessity. The edge-level probe in §7 supersedes this test.

the size of the graph it was drawn from. We therefore report the fraction and the total edge count in Table 2; reporting neither, as is common, is what lets the artifact through unnoticed.

## D Entailment Probe Details (RQ4)

**Synthetic probe construction.** Items instantiate “Premise: {name} has {x} {noun}. Hypothesis: {name} has more than {y} {noun}. Question: Is the hypothesis true? Answer:” with  $N=500$  balanced items per model and condition, scored as the single-token  $\text{logit}(\text{“Yes”}) - \text{logit}(\text{“No”})$  at the final position, sign-flipped for “No” items

so that positive always means the model favored the correct answer. Accuracy is the fraction of items with a positive value. Corruption replaces  $x$  with a value on the opposite side of  $y$  having the same digit count, so clean and corrupted sequences stay length-matched; per-digit tokenizers otherwise break the pairing, which is also why the year task cannot be posed to Qwen2.5-7B. All numbers must be single-token except where noted.

**Format controls.** Both control probes reuse the numerical probe’s scaffolding programmatically rather than by copying it, so the prompt shape cannot drift between probe and control. The *format floor* substitutes a colour attribute for the numeric comparison (“... has red {noun}” versus “... has blue {noun}”), answered Yes exactly when the attributes match; the counterpart attribute is chosen to have the same token length. The *ordinal* probe replaces numerals with alphabet positions (“... is at letter K” versus “... is past letter F”), preserving the greater-than structure. Neither probe contains a digit anywhere. Table 4 gives the numerical gate per model and condition, which Table 3 summarizes as its best condition.

**Necessity probes.** The node-level probe zero-ablates pythia-2.8b’s late MLPs (layer fraction  $\geq 0.6$ ) against a size-matched random-MLP control, both under the two-shot condition that produced its 0.576 accuracy. The edge-level probe holds dataset, condition, and accuracy definition fixed and instead clears `in_graph` on the edges en-

tering those MLPs: 37 of them belong to the discovered circuit, 9,813 in total. Random-edge controls sample the same count from all edges whose child is any MLP, so they are matched in count and kind.

**Naturalistic evaluation.** EQUATE subsets are RTE-Quant and AWPNLI. NumPert claim-evidence pairs are posed as premise and hypothesis; the positional circuit evaluation uses the 167 pairs that NumPert’s random-replacement perturbation leaves exactly length-matched, which is what makes a positional clean/corrupted comparison well defined on naturalistic text.