REPEAT OR RETRIEVE? LOCAL COPIES SUPPRESS REMOTE-SOURCE SENSITIVITY

Anonymous authors
Paper under double-blind review

ABSTRACT

Reasoning traces often state an intermediate value twice: once where it is computed and again where it is used. Does the next prediction depend on the earlier source or the nearby copy? We separate these occurrences with source-span activation patching and controlled edits to the copy. Across Gemma-2, OLMo-2, and Qwen3, a correct local copy reduces the mean absolute source-patch effect by more than 99% on controlled arithmetic traces. Randomized trace formats and length-matched filler controls show that the reduction depends on copied values. Controlled training shows that this preference depends on the reliability of copies. In small transformers trained from scratch, corrupting one operand on half of the restated training lines reverses the preference at fixed restatement frequency: copy sensitivity becomes negligible while source sensitivity remains. When copies are reliable, making them more frequent also strengthens source suppression. The preference affects generation: a conflicting copy supplied in an arithmetic prefix makes Gemma-2 and Qwen3 produce its implied result about 90% of the time, and a source change reaches a copy that the model writes itself in 81–94% of samples. These results show how local restatements change which occurrence affects a prediction and identify training conditions that can reverse that change.

1 INTRODUCTION

A chain-of-thought (CoT) trace often writes an intermediate value more than once. We call the line that defines a value its *source* and a later line that uses it a *consumer*. For example, in $x_5 = x_3 - x_1 = 28 - 16 = 12$, the operand 28 is a *local copy* of the value assigned to x_3 at its source. The source and local copy contain the same value but may play different computational roles. Knowing which occurrence affects the next prediction would make reasoning traces easier to interpret. It also matters for auditing: a checker that sees only the consumer line can inspect its arithmetic, but cannot establish whether the copied operands agree with their sources (Korbak et al., 2025; Chen et al., 2025).

Prior work shows that trace edits can change answers (Lanham et al., 2023), written scratchpad states can act as causal registers (Shih et al., 2026), trailing numbers can dominate final-answer readout (Liu, 2026), and steps in an arithmetic chain can depend on the most recent statement of the information they use (Kudo et al., 2026). We ask how two occurrences of the *same intermediate value*, at its source and next use, affect the consumer prediction.

We patch the source line's activations while holding the displayed copy fixed, and separately edit the copy (Fig. 1). On identical arithmetic problems, a correct local copy nearly eliminates source effects across three models; length-matched filler does not. We call this the *local-copy substitution effect*. Conflicting copies also redirect many decoded results, and a source change also reaches a copy that the model writes itself.

Controlled training reverses this preference when copies become unreliable. We train small transformers with fixed source definitions and correct results while varying copied operands. Corrupting one operand on half of the restated training lines reverses suppression at fixed restatement frequency. With reliable copies, increasing their frequency strengthens suppression, including a setting accurate with and without copies. These experiments identify sufficient training conditions, without establishing how deployed models acquired the preference.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

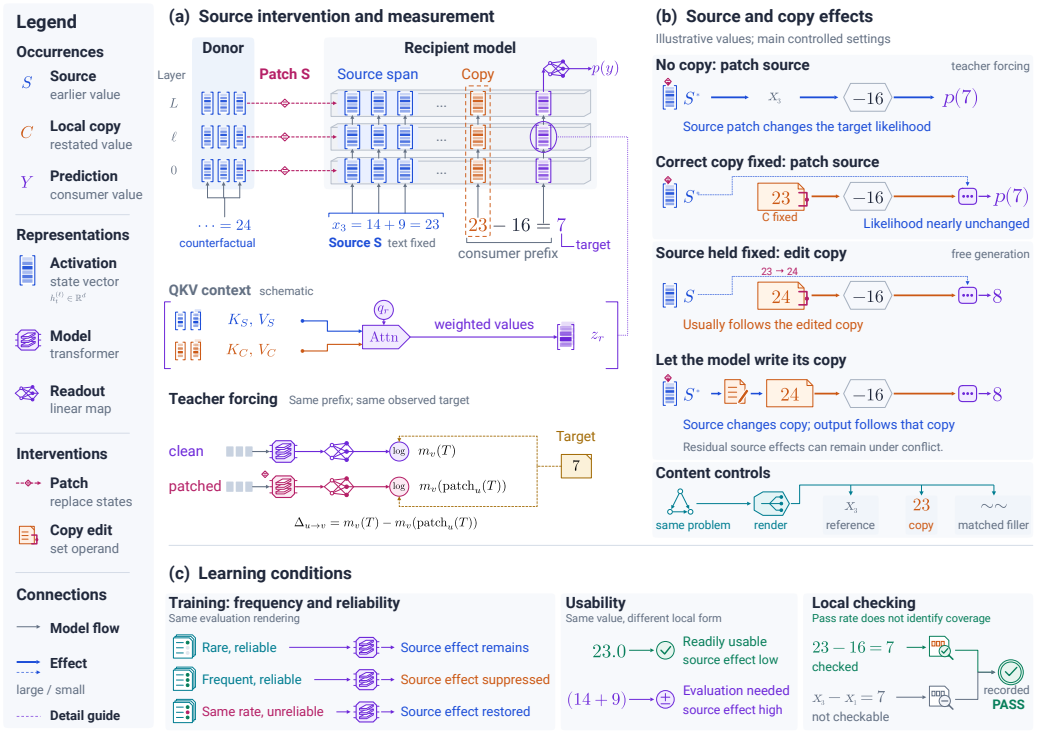

Figure 1: **Separating two occurrences of the same intermediate value.** (a) A counterfactual donor changes source activations while the displayed trace stays fixed. (b) Holding a correct copy fixed suppresses source sensitivity; a conflicting copy supplied in the prefix redirects many arithmetic-template generations. After a source patch that keeps the variable name, Gemma-2, Qwen3, and OLMo-2 write the intended counterfactual copy in 81–94% of samples and the counterfactual result in 96–99% (Sec. 5). (c) Frequent reliable restatements favor copies, whereas corrupting one operand on half or all of the restated training lines reverses suppression at fixed frequency (Sec. 6). The arithmetic expressions illustrate the intervention settings.

We make three contributions:

- We show that a correct local copy suppresses sensitivity to the earlier source. Randomized renderings and content controls establish the effect, while tests across formats, value types, and models identify its boundaries.
- We distinguish predicting from a supplied copy from generating the copy itself. Conflicting copies strongly influence arithmetic-template predictions, and source interventions also propagate through self-written copies.
- We demonstrate that controlled training can reverse the preference by changing copy reliability at fixed frequency. A complementary frequency sweep shows when strong suppression coexists with accurate prediction in both formats.

2 Related Work

Interventions on reasoning traces. Faithfulness studies use truncation, early answering, mistake insertion, biased contexts, and mediation measures (Lanham et al., 2023; Turpin et al., 2023; Paul et al., 2024). Other work treats the whole trace as a mediator (Jia et al., 2026) or studies individual steps through correctness checks, resampling, and premise-dependence tests (Zhao et al., 2026; Bogdan et al., 2025; Macar et al., 2026; Nakamura, 2026). The closest comparisons show that a written state can affect a later transition (Shih et al., 2026), that positional copying can dominate final-answer readout (Liu, 2026), and that patching the problem statement has almost no effect once the chain restates the information (Kudo et al., 2026). We compare two occurrences within one

intermediate dependency, then vary the trace format and training distribution that determine their relative effects.

Text as a computational resource. Models can compute using filler tokens, pauses, or latent states (Pfau et al., 2024; Goyal et al., 2024; Deng et al., 2024; Hao et al., 2025). Expressivity results characterize how externalized tokens support computation (Merrill & Sabharwal, 2024; Li et al., 2024; Pan et al., 2026). Information can also pass through hidden states (Mehrafarin et al., 2026). Our experiments concern dependence on particular written occurrences, rather than whether meaningful text is necessary for reasoning in general.

Trace formats and reasoning circuits. Natural CoT and scratchpads often restate values inline (Wei et al., 2022; Cobbe et al., 2021; Nye et al., 2021); program-style traces refer to earlier results by name (Gao et al., 2023; Chen et al., 2023). We vary this distinction within matched problems. We use activation patching (Vig et al., 2020; Geiger et al., 2021; Meng et al., 2022) with established corruption and metric controls (Zhang & Nanda, 2024; Kramár et al., 2024), building on causal mediation studies of arithmetic (Stolfo et al., 2023). Studies of mid-layer transformations (Dutta et al., 2024), relay and retrieval heads (Cabannes et al., 2024; Wu et al., 2025), attribute extraction (Geva et al., 2023; Biran et al., 2024), and multi-hop pathways (Yang et al., 2024) motivate our partial localization of the effect.

Learning from distributional structure. Corpus statistics, including burstiness and class imbalance (Chan et al., 2022), can shape in-context learning; induction heads provide one mechanistic example (Olsson et al., 2022). We apply this logic to repeated intermediate values, varying the frequency and reliability of copies in a controlled corpus.

3 Measuring Occurrence-Level Dependence

Traces with known dependencies. Our primary testbed uses synthetic arithmetic traces with known dependency graphs. Each line defines an input value or combines earlier values by addition, subtraction, or multiplication. A *variable-form* consumer refers to earlier results by name; a *restated* consumer also writes their values. Interleaving independent branches supplies no-path pairs matched to direct dependencies on step distance, controlling for separation between steps. We also use relational traces and parse model-generated solutions to these problems and GSM8K (Cobbe et al., 2021).

Source intervention. We change the earlier source while holding the consumer's displayed copy fixed. For a source step u and consumer step v in trace T, we construct a counterfactual source line with different values but the same token length. Its activations replace those of the original source span at the embedding output and every layer. Most experiments resample every digit of this line, including the defined variable's index; Sec. A1.3 repeats key analyses with the name kept. The model is evaluated with the original trace tokens supplied, a procedure known as *teacher forcing*. Let $m_v(T)$ be the mean token log-probability of the value stated at the consumer. The *source effect*, measured in nats, is

$$\Delta_{u \to v} = m_v(T) - m_v(\mathrm{patch}_u(T)). \tag{1}$$

A positive effect means that the intervention lowers the original value's probability. The absolute effect, $|\Delta|$, measures the magnitude of the log-probability change in either direction. For the transformer experiments, this all-layer patch is equivalent to the corresponding source-text swap under teacher forcing. It also supplies a reference for the later layer- and head-specific interventions.

Occurrence comparison. We render each problem with and without a correct local copy, then compare the mean absolute source effects. If the copy substitutes for the earlier occurrence, changing the source should matter less when that copy is available. Random assignment separates this effect from the model's choice of which values to repeat. We subsequently edit the copy itself and test how the generated result changes. The initial comparison uses exact-value copies; Sec. 4 tests alternative surface forms. Sec. A1.1 gives the formal restatement indicator and contrast.

What the comparison measures. The intervention measures sensitivity at one direct dependency, conditional on the supplied text. A small effect does not establish that the source is absent from the model's internal state or that the full trace is faithful. Holding intermediate text fixed also excludes changes that would propagate through regenerated words. The source may therefore be important for producing a copy even when it has little measured effect after that copy is supplied. We test this distinction by regenerating the consumer line in Sec. 5.

Validation. On variable-form templates, the instrument distinguishes direct edges from distance-matched no-path pairs. The area under the receiver operating characteristic curve (AUROC) ranges from about 0.80 to 0.96 across Gemma-2-2B (Team et al., 2024), OLMo-2-7B (Walsh et al., 2025), and Qwen3-8B (Yang et al., 2025). Restoring clean source activations at all layers fully recovers the original prediction in the Gemma-2 validation. Transitive pairs show no teacher-forced effect because intervening written values remain fixed. These checks validate the measurement of local dependency effects; they do not establish that every dependency in a freely generated trace is used. Sec. A1.1 reports the model-wise estimates, uncertainty, and permutation controls.

4 Correct Local Copies Suppress Source Sensitivity

Observation and format forcing. We begin with arithmetic traces generated by the models themselves. Source effects are generally smaller when the consumer repeats the source value, although Qwen3 retains appreciable sensitivity on these edges (Table A7). Because the model chooses which values to repeat, this observation alone cannot establish the effect of adding a copy.

Prompting for restatement or variable-only references reproduces the contrast, but introduces a compliance problem. All three models reliably add copies when asked. Only Qwen3 reliably omits them when asked; Gemma-2 and OLMo-2 still restate most values (Table A42). Their remaining no-copy edges are therefore a selected subset. This motivates assigning the trace format directly.

Randomization and content controls. We compare assigned renderings of identical problems. Across three models, the mean absolute source effect is 1.33–6.87 nats without a copy and at most 0.011 with a correct local copy (Table A11). Crossing instruction with rendering leaves no detectable instruction effect when a copy is present (Table A12). With a copy, source patching raises the probability of the counterfactual-implied result about as much as that of random values (Sec. A1.3).

The configuration sweep in Fig. 2 tests why the copy matters. Length-matched filler preserves the extra token positions without carrying the values; a bare number repeats the value without its variable binding. Neither reproduces the correct-copy reduction in Gemma-2. A bound duplicate placed after all consumers also leaves source sensitivity essentially unchanged, as expected from its unavailability to those predictions. The reduction thus depends on the copy's content and placement, rather than simply on additional tokens. With a name-preserving counterfactual, a wrong local copy leaves only 9–15% of the effect, less than a correct duplicate after the source (32–63%; Sec. A1.3).

Signed and absolute effects distinguish suppression from cancellation. A correct duplicate placed near the source produces positive changes on some edges and negative changes on others. These changes cancel in the signed average, even though individual predictions remain sensitive to the source (Sec. A2.1). A correct local copy instead reduces both the signed and absolute effects. The contrast explains why a small signed mean alone is insufficient evidence of suppression.

Correct-copy suppression also replicates in DeepSeek-R1-Distill-Qwen-7B, distilled from reinforcement learning with verifiable rewards (RLVR; Guo et al., 2025), and in Gemma-2-27B. Restatement prompts also yield higher task accuracy than variable-form prompts across the tested models (Table A42). Local copies can therefore be useful for prediction even as they weaken sensitivity to an earlier occurrence.

Surface form changes the effect. Exact token identity is unnecessary. Number words, decimals, and currency renderings strongly suppress source sensitivity, whereas zero-padded numbers, spaced digits, and arithmetic expressions leave larger effects (Table A13). Thus, two strings denoting the same value need not have the same influence on a prediction. We compare the same leaf-source edges, keeping the intervened source line identical across styles. Token positions align within each clean–counterfactual pair, but token lengths and positions are not matched across styles. The

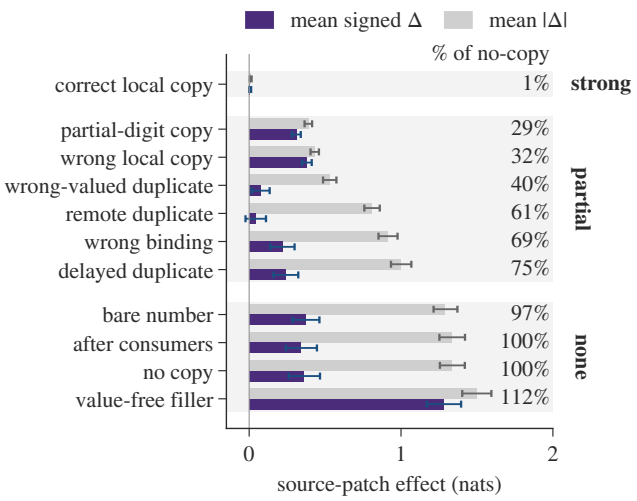

Figure 2: **Source sensitivity across copy configurations.** Mean signed and mean absolute effects in Gemma-2-2B, with 1000 direct edges per configuration and trace-clustered bootstrap 95% CIs. The delayed-duplicate bar pools 543 cases with the copy before its consumer and 457 with the copy after it. A correct, bound, line-local copy removes 99% of the mean absolute no-copy effect. Wrong, partial, remote, mis-bound, or delayed copies reduce it by only 25–71%, while bare numbers, copies placed after their consumers, and value-free filler produce little or no reduction. Right column: mean absolute effect as a percentage of the no-copy baseline. Table A11 gives the cross-model grid.

comparison does not isolate normalization cost from tokenization, length, or familiarity. For person names, uppercasing, adding an honorific, and using a supplied alias also suppress source sensitivity.

Generality and boundaries. Strong suppression also appears when intermediate values are words or person names (Fig. A3). Copies without explicit variable bindings also suppress sensitivity for these value types, whereas bare numbers have little effect. On PrOntoQA deduction (Saparov & He, 2023), rule restatement reduces the mean signed source effect in Qwen3 and OLMo-2, but not in Gemma-2. Correct copies also reduce mean signed source effects in Mamba2 (Dao & Gu, 2024) and RWKV (Peng et al., 2023), extending the observation beyond transformers (Sec. A3.4). These comparisons show both generality and limits: the value type and model change which copy properties matter.

In parsed self-generated GSM8K solutions, source values that appear again later have much smaller signed effects on the final parsed result (Table A41). This is an observational comparison involving few sources without a later occurrence. It does not isolate a copy at one consumer, and conservative parsing limits its scope (Sec. A1.5).

5 Using a Supplied Copy and Generating One

We next ask how the occurrence contrast affects generated values. First, we supply the consumer prefix, including its operand copy, and let the model generate the result. Then we let the model write the entire consumer line, including the copy. The first setting tests prediction from an available operand; the second tests whether a source intervention propagates into a copy the model writes.

A supplied copy controls many decoded outcomes. We cross source activation patching with edits to the local copy. With a correct copy fixed, we ask whether source patching changes how often the original result is produced. With a conflicting copy fixed, we measure the change in how often its implied result is produced, denoted Δ_S. Sec. A2.6 defines these frequency contrasts and reports all four source–copy conditions.

With a clean source and a conflicting copy, template generations produce the copy-implied result in 89–90% of cases (Table 1). When the patched source and copy agree, however, their matching result cannot distinguish the two occurrences. A separate Gemma-2 experiment therefore assigns distinct results to the original source, patched source, and wrong copy. The wrong-copy result still dominates (Sec. A2.5). This supports the behavioral preference without treating outcome frequencies as direct measurements of internal route use.

		counterfactual result			original result		
Model	Format	$S = 0$	$S = 1$	Δ_S	$S = 0$	$S = 1$	n/cell
Gemma-2-2B	template	.891	.891	.000	.002	.003	976
Qwen3-8B	template	.903	.905	.002	.004	.001	1166
Qwen3-8B	<think>	.886	.907	.021	.009	.002	1166
R1-32B	<think>	.863	.919	.057	.048	.003	583
R1-70B	<think>	.785	.931	.146	.146	.003	391
Gemma-2-2B	GSM8K	.877	.878	.001	.026	.026	896
Qwen3-8B	GSM8K	.756	.814	.059	.076	.040	700

Table 1: **Generated-result frequencies with a counterfactual copy fixed ($C = 1$).** $S = 0$ denotes a clean source; $S = 1$ denotes a patched source. In template and <think> rows, patching makes the source agree with the copy. GSM8K instead uses independent source corruption and a designated single-operand candidate (Sec. A3.2). Δ_S is the change in counterfactual-result frequency after source patching. These are decoded outcomes: when source and copy agree, the result cannot identify which occurrence was used. Correct-copy cells appear in Sec. A2.6 and Table A23.

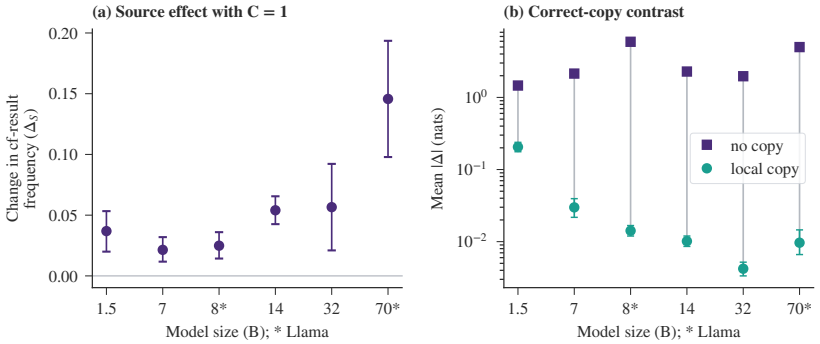

Figure 3: **Six-model comparison.** Left: Δ_S, the source-induced change in counterfactual-result frequency with $C = 1$ fixed. The nominal 95% intervals use a trace-cluster bootstrap through 14B and normal approximations without clustering for the summary-only 32B/70B generation runs. The association with size is descriptive (Spearman $\rho = 0.83$); base architectures differ. Right: mean absolute source-patch effects with and without a correct local copy, with 95% percentile intervals from 2000 paired trace-cluster bootstrap draws. Connectors join the two formats within each model. Starred models are Llama-based; the others are Qwen-based.

Correct and conflicting copies have different boundaries. The six-model comparison separates suppression with a correct copy from source sensitivity under conflict (Fig. 3). Correct copies strongly suppress the likelihood-level source effect across the R1-Distill series. With a correct copy fixed, source patching changes original-result frequency by less than two percentage points at every tested size.

With a conflicting copy fixed, the change is larger in some models, reaching about fifteen percentage points in the largest model. Strong suppression with a correct copy can therefore coexist with measurable source sensitivity under conflict. The association with size is descriptive: the ordering is not monotonic among the smaller models, and parameter count varies together with the Qwen or Llama base architecture (Sec. A3.3).

Source sensitivity can also reappear without changing the decoded value. In a separate joint intervention, a wrong copy increases likelihood-level source effects in all four tested models (Table A39). A change in a result's likelihood need not change which value is decoded. Residual source sensitivity is therefore compatible with copy-dominated generation.

Parsed steps from model-generated solutions. We also edit arithmetic steps extracted from self-generated GSM8K solutions, supplying the consumer prefix and regenerating only its result. The designated counterfactual candidate is computed by replacing the target operand in the original parsed

operation. Both Gemma-2 and Qwen3 frequently produce this candidate when the source is clean (Table 1). Source donors are corrupted independently of the copy edit, so source patching does not ensure agreement between the occurrences. These measurements concern selected, locally consistent steps with explicit copied operands; they do not measure unrestricted solution accuracy. Sec. A3.2 details the selection and candidate definition. In full model-written solutions, a supplied copy also outweighs its source: across 12 settings, a correct copy keeps the original result in 94–100% of edges, and in few-shot runs a conflicting copy sets it in 92–100% (Sec. A2.7). In `<think>` runs, conflicting copies influence the consumer less consistently, and continuations usually return to the original final answer even when the consumer result follows the copy.

A no-copy reference. We also evaluate variable-form prefixes that supply no operand copies. Separate decoding and log-probability evaluations show source sensitivity, but the baseline exact-result accuracy is only 17% (Table A40). Changing the source can therefore shift a prediction without enabling reliable calculation from that source. This reference helps separate occurrence sensitivity from task competence.

Generating the copy. Finally, we let Gemma-2, Qwen3, and OLMo-2 generate the entire consumer line, including its operand copies. A separate problem block retains the step definitions, so the target computation remains specified when the supplied consumer line is removed. We use counterfactual source lines that preserve the variable name and remain arithmetically consistent. Across the three models, the intended counterfactual copy appears in 81–94% of patched samples, and the counterfactual result in 96–99% (Table A4). Both rates use all samples, including those without an extractable target copy or result. The source can therefore guide a copy that the model writes, even when it has little effect once a correct copy is supplied.

An earlier digit-resampling intervention also renamed the source variable in most samples, leaving the consumer's reference undefined and producing much lower propagation rates. Sec. A2.8 reports that condition's full outcomes and the single-edge agreement diagnostic.

6 Learning the Preference from Controlled Corpora

The pretrained-model experiments establish the preference but do not identify how it was acquired. We test whether next-token training can produce or reverse it by varying copy frequency and reliability while holding the underlying synthetic problem stream fixed.

Design and readout. We train small, six-layer decoder-only transformers from scratch. Each internal training line restates its operands with probability p. Every model is evaluated on the same traces, with copy and no-copy lines sampled with equal probability and disjoint training and evaluation seeds. Thus, evaluation format is fixed while the training distribution changes. To compare the two evaluation formats, we summarize source suppression as

$$G = 1 - \frac{\mathbb{E}|\Delta|_{\text{restate}}}{\mathbb{E}|\Delta|_{\text{varform}}}. \tag{2}$$

A value near one indicates strong suppression when a copy is present. Zero indicates equal source sensitivity in the two formats; a negative value indicates greater sensitivity on restated lines. We also patch two segments of the consumer line separately: its copied operands and its variable references. These probes distinguish sensitivity to the local value from sensitivity to the name referring to its source. Reporting criteria check that the model has learned leaf predictions and that interventions produce a measurable signal (Sec. A5.1). Consumer accuracy is reported separately, so source sensitivity is not mistaken for calculation competence.

Reliability reverses the preference at fixed frequency. We fix restatement frequency at one half and corrupt one randomly selected operand copy on a fraction of those lines. The correct result and source definitions remain unchanged. Restated lines are equally frequent, but their copied values are less reliable.

Corrupting one operand on half of the restated lines reverses source suppression in both training seeds (Fig. 4(a)). Sensitivity to the copied-operand segment becomes negligible, while source and

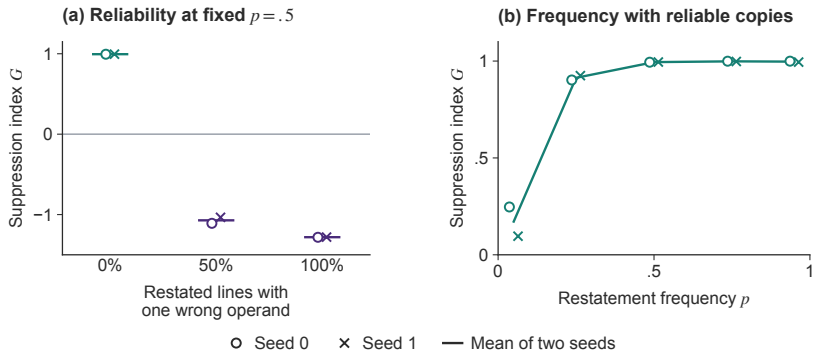

Figure 4: **Training reliability and frequency control source suppression.** (a) At fixed restatement frequency $p = 0.5$, corrupting one operand on 50% or 100% of restated lines reverses the suppression index G. (b) With reliable copies, G increases with restatement frequency. Circles and crosses show two training seeds at $40{,}000$ steps; horizontal lines in (a) and the connected line in (b) show their means. Seed markers are offset horizontally for visibility.

	remote source		local readouts			
p	no copy	with copy	copy	reference	implied	G (two seeds)
0.05	8.97	6.76	0.78	7.03	-8.8	0.25 / 0.10
0.25	12.89	1.26	17.39	0.55	+14.2	0.90 / 0.92
0.50	12.88	0.08	19.66	0.02	+19.1	0.99 / 1.00
0.75	11.69	0.01	23.22	0.05	+23.6	1.00 / 1.00
0.95	5.96	0.00	23.27	0.04	+23.3	1.00 / 0.99

Table 2: **Training frequency changes occurrence sensitivity.** p is the probability that a training line restates its operands. Remote-source columns report mean absolute patch effects in the two evaluation formats. Local probes patch the copied-operand segment or the variable-reference segment on restated lines. The implied column is the log-probability advantage of the result implied by a corrupted copy over the true result. Effects are in nats from seed 0; G is shown for seeds 0 and 1. The probes act at different distances from the prediction; we report their raw effects without normalizing one by another. Near-zero sensitivity is rounded to two decimals; competence diagnostics appear in Table A31.

variable-reference interventions remain effective. Both models retain accurate value prediction on clean restated evaluation lines, so the reversal occurs in a competent condition (Table A32). When we subsequently corrupt a local copy at evaluation, they favor the true result over the result implied by that copy. Corrupting one operand on every restated line produces the same qualitative change. Thus, making copies unreliable reverses suppression even though their frequency is held fixed.

Frequency changes the preference. When only 5% of training lines restate their operands, a supplied copy leaves substantial source sensitivity, while its local-copy effect is much smaller than at higher restatement rates. Increasing restatement to 25% sharply reduces source sensitivity on copied lines and makes the local-copy probe influential. The result implied by a corrupted copy also becomes more likely than the true result. Both seeds show this change (Table 2; Fig. 4(b)).

The quarter-restatement setting provides the clearest case of suppression with competent prediction: value-token accuracy in the expanded diagnostics averages above 98% in both evaluation formats. The endpoints require more caution. When copies are rare, prediction accuracy is moderate in both formats, so learning progress can contribute to the apparent transition. When copies are very frequent, the model predicts copied lines accurately but performs poorly without copies, despite retaining source sensitivity (Table A31). The sweep therefore establishes a training-dependent preference, not a universal frequency threshold or uniformly reliable retrieval.

Training on extra tokens is insufficient. The earlier filler control changed the evaluation trace. Here, we replace copied values with length-matched filler in the training corpus and evaluate the trained model in that same rendering. These models retain strong source sensitivity, showing that extra training tokens alone do not reproduce suppression. With real copies, sensitivity instead depends on whether the evaluation line supplies useful values (Sec. A5.1).

Format training also shifts pretrained models. Supervised fine-tuning (SFT) with low-rank adaptation (LoRA; Hu et al., 2022) on variable-form solutions raises Gemma-2-2B's source sensitivity about fivefold. Restatement training lowers it, and mixed-format training produces intermediate changes. We evaluate every checkpoint on the same held-out variable-form traces, so the comparison does not follow from a change in the generated format. Final-checkpoint task accuracy remains high across seeds (Sec. A5.2). Joint interventions still show little source sensitivity when a correct copy is supplied (Table A34). This extension changes dependence chiefly in the no-copy setting.

7 Discussion

What the evidence supports. Dependence attaches to a particular written occurrence of a value, and controlled training can reverse which occurrence affects the prediction. Auxiliary depth and head interventions partially localize this dependence; head ablation restores only part of the suppressed sensitivity (Sec. A4).

Interpreting repeated values. The results separate three questions: whether a result follows the displayed operands, whether those operands match earlier definitions, and whether changing either occurrence changes the prediction. A line can pass the first check while failing the second; neither check alone identifies causal dependence. Intervention reports should therefore specify the edited occurrence, the text held fixed, and the regenerated span.

Implication for line-local checking. A line-local arithmetic checker can recompute a result only when all required operands appear on that line. Its *coverage* is the fraction of parsed, variable-referencing internal lines meeting this condition, averaged over traces. Two Qwen3 selection-and-fine-tuning loops illustrate why coverage should accompany pass rate (Emmons et al., 2025). Both retain only passing traces, with missing operands causing no rejection. Random selection raises coverage; selecting the least-restating trace lowers it, while every retained trace passes. An unchanged pass rate can therefore conceal a change in how much arithmetic the checker can inspect. These outcomes follow the selection rule and do not establish a consequence of neural substitution. Sec. A6 reports the checkers, accuracy, selection controls, and the separate behavior of source references.

Limitations. The main causal analyses use 2–8B instruction-tuned models and controlled tasks. Larger models, non-numeric values, and other architectures broaden the evidence but do not establish the effect in unrestricted reasoning. Surface-form comparisons do not match token length across styles. The training sweep establishes sufficiency under one training setup; moderate low-frequency accuracy and poor high-frequency no-copy accuracy preclude a universal frequency threshold. The instrument measures output sensitivity rather than internal route-use frequencies and cannot fully separate changes in value dependence from confidence recalibration. Closed-weight models cannot be tested with this activation instrument.

Reproducibility statement. The supplement provides seeded code and saved-result summaries for our synthetic, PrOntoQA, and GSM8K experiments. Code and results: `https://anonymous.4open.science/r/paper-RepeatOrRetrieve-A154`. The Appendix specifies checkpoints, hyperparameters, parsing coverage, intervals, reporting criteria, and pre-specified extensions.

Conclusion. Reliable local copies can shift dependence away from earlier sources. Controlled training reverses this preference when copies become unreliable. In arithmetic templates, supplied copies strongly influence the next result; self-written copies usually follow source changes that preserve variable names. This local influence need not determine the final answer: `<think>` continuations often return to their original answer. Interpreting traces therefore requires distinguishing written occurrences, their effects on predictions, and the dependencies a checker can verify.

AI USE STATEMENT

We used generative AI tools for language editing and code debugging. We did not use generative AI tools to form research ideas or hypotheses, to design the experiments or the preregistered decision rules, to run any experiment, or to produce, select, or interpret any reported result. No number, table entry, interval, or figure in this paper was produced by a generative AI tool.

REFERENCES

Usman Anwar, Tim Bakker, Dana Kianfar, Cristina Pinneri, and Christos Louizos. Analyzing and improving chain-of-thought monitorability through information theory. *arXiv preprint arXiv:2602.18297*, 2026.

Eden Biran, Daniela Gottesman, Sohee Yang, Mor Geva, and Amir Globerson. Hopping too late: Exploring the limitations of large language models on multi-hop queries. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 14113–14130, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.781. URL https://aclanthology.org/2024.emnlp-main.781/.

Paul C Bogdan, Uzay Macar, Neel Nanda, and Arthur Conmy. Thought anchors: Which LLM reasoning steps matter? *arXiv preprint arXiv:2506.19143*, 2025.

Vivien Cabannes, Charles Arnal, Wassim Bouaziz, Alice Yang, Francois Charton, and Julia Kempe. Iteration head: A mechanistic study of chain-of-thought. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 109101–109122. Curran Associates, Inc., 2024. doi: 10.52202/079017-3463. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/c50f8180ef34060ec59b75d6e1220f7a-Paper-Conference.pdf.

Stephanie Chan, Adam Santoro, Andrew Lampinen, Jane Wang, Aaditya Singh, Pierre Richemond, James McClelland, and Felix Hill. Data distributional properties drive emergent in-context learning in transformers. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 18878–18891. Curran Associates, Inc., 2022. doi: 10.52202/068431-1371. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/77c6ccacfd9962e2307fc64680fc5ace-Paper-Conference.pdf.

Wenhu Chen, Xueguang Ma, Xinyi Wang, and William W. Cohen. Program of thoughts prompting: Disentangling computation from reasoning for numerical reasoning tasks. *Transactions on Machine Learning Research*, 2023. ISSN 2835-8856. URL https://openreview.net/forum?id=YfZ4ZPt8zd.

Yanda Chen, Joe Benton, Ansh Radhakrishnan, Jonathan Uesato, Carson Denison, John Schulman, Arushi Somani, Peter Hase, Misha Wagner, Fabien Roger, et al. Reasoning models don't always say what they think. *arXiv preprint arXiv:2505.05410*, 2025.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Tri Dao and Albert Gu. Transformers are SSMs: Generalized models and efficient algorithms through structured state space duality. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 10041–10071. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/dao24a.html.

Yuntian Deng, Yejin Choi, and Stuart Shieber. From explicit CoT to implicit CoT: Learning to internalize CoT step by step. *arXiv preprint arXiv:2405.14838*, 2024.

Subhabrata Dutta, Joykirat Singh, Soumen Chakrabarti, and Tanmoy Chakraborty. How to think step-by-step: A mechanistic understanding of chain-of-thought reasoning. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. URL https://openreview.net/forum?id=uHLDkQVtyC.

Scott Emmons, Roland S Zimmermann, David K Elson, and Rohin Shah. A pragmatic way to measure chain-of-thought monitorability. *arXiv preprint arXiv:2510.23966*, 2025.

Luyu Gao, Aman Madaan, Shuyan Zhou, Uri Alon, Pengfei Liu, Yiming Yang, Jamie Callan, and Graham Neubig. PAL: Program-aided language models. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 10764–10799. PMLR, 23–29 Jul 2023. URL https://proceedings.mlr.press/v202/gao23f.html.

Atticus Geiger, Hanson Lu, Thomas Icard, and Christopher Potts. Causal abstractions of neural networks. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 9574–9586. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/4f5c422f4d49a5a807eda27434231040-Paper.pdf.

Mor Geva, Jasmijn Bastings, Katja Filippova, and Amir Globerson. Dissecting recall of factual associations in auto-regressive language models. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 12216–12235, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.751. URL https://aclanthology.org/2023.emnlp-main.751/.

Sachin Goyal, Ziwei Ji, Ankit Singh Rawat, Aditya Krishna Menon, Sanjiv Kumar, and Vaishnavh Nagarajan. Think before you speak: Training language models with pause tokens. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=ph04CRkPdC.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. DeepSeek-R1 incentivizes reasoning in LLMs through reinforcement learning. *Nature*, 645(8081):633–638, 2025.

Shibo Hao, Sainbayar Sukhbaatar, DiJia Su, Xian Li, Zhiting Hu, Jason E Weston, and Yuandong Tian. Training large language models to reason in a continuous latent space. In *Second Conference on Language Modeling*, 2025. URL https://openreview.net/forum?id=Itxz7S4Ip3.

Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

Jinghan Jia, Joe Benton, and Eric Easley. Faithfulness as information flow: Evaluating and training faithful chain-of-thought reasoning. *arXiv preprint arXiv:2605.24286*, 2026.

Tomek Korbak, Mikita Balesni, Elizabeth Barnes, Yoshua Bengio, Joe Benton, Joseph Bloom, Mark Chen, Alan Cooney, Allan Dafoe, Anca Dragan, et al. Chain of thought monitorability: A new and fragile opportunity for AI safety. *arXiv preprint arXiv:2507.11473*, 2025.

János Kramár, Tom Lieberum, Rohin Shah, and Neel Nanda. AtP*: An efficient and scalable method for localizing LLM behaviour to components. *arXiv preprint arXiv:2403.00745*, 2024.

Keito Kudo, Yoichi Aoki, Tatsuki Kuribayashi, Shusaku Sone, Masaya Taniguchi, Ana Brassard, Keisuke Sakaguchi, and Kentaro Inui. LLMs faithfully and iteratively compute answers during CoT: A systematic analysis with multi-step arithmetics. In Vera Demberg, Kentaro Inui, and Lluís Marquez (eds.), *Findings of the Association for Computational Linguistics: EACL 2026*, pp. 1114–1153, Rabat, Morocco, March 2026. Association for Computational Linguistics. ISBN 979-8-89176-386-9. doi: 10.18653/v1/2026.findings-eacl.59. URL https://aclanthology.org/2026.findings-eacl.59/.

Tamera Lanham, Anna Chen, Ansh Radhakrishnan, Benoit Steiner, Carson Denison, Danny Hernandez, Dustin Li, Esin Durmus, Evan Hubinger, Jackson Kernion, et al. Measuring faithfulness in chain-of-thought reasoning. *arXiv preprint arXiv:2307.13702*, 2023.

Zhiyuan Li, Hong Liu, Denny Zhou, and Tengyu Ma. Chain of thought empowers transformers to solve inherently serial problems. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=3EWTEy9MTM.

Bryce Little. Length penalties make chain-of-thought less monitorable. *arXiv preprint arXiv:2607.09786*, 2026.

Ming Liu. The readout shortcut: Positional number copying dominates arithmetic CoT readout in small language models. *arXiv preprint arXiv:2605.22870*, 2026.

Uzay Macar, Paul C. Bogdan, Senthooran Rajamanoharan, and Neel Nanda. Thought branches: Interpreting LLM reasoning requires resampling. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=bVsAuIOvJ5.

Matt MacDermott, Qiyao Wei, Rada Djoneva, and Francis Rhys Ward. Reasoning under pressure: How do training incentives influence chain-of-thought monitorability? *arXiv preprint arXiv:2512.00218*, 2025.

Thomas McGrath, Matthew Rahtz, Janos Kramar, Vladimir Mikulik, and Shane Legg. The hydra effect: Emergent self-repair in language model computations. *arXiv preprint arXiv:2307.15771*, 2023.

Houman Mehrafarin, Amit Parekh, and Ioannis Konstas. When chain-of-thought fails, the solution hides in the hidden states. *arXiv preprint arXiv:2604.23351*, 2026.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in GPT. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 17359–17372. Curran Associates, Inc., 2022. doi: 10.52202/068431-1262. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/6f1d43d5a82a37e89b0665b33bf3a182-Paper-Conference.pdf.

William Merrill and Ashish Sabharwal. The expressive power of transformers with chain of thought. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=NjNGlPh8Wh.

Hironao Nakamura. Interventional grounding audits: Black-box premise-dependency tests for LLM chain-of-thought via predicate substitution. *arXiv preprint arXiv:2607.13069*, 2026.

Maxwell Nye, Anders Johan Andreassen, Guy Gur-Ari, Henryk Michalewski, Jacob Austin, David Bieber, David Dohan, Aitor Lewkowycz, Maarten Bosma, David Luan, et al. Show your work: Scratchpads for intermediate computation with language models. *arXiv preprint arXiv:2112.00114*, 2021.

Catherine Olsson, Nelson Elhage, Neel Nanda, Nicholas Joseph, Nova DasSarma, Tom Henighan, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Dawn Drain, Deep Ganguli, Zac Hatfield-Dodds, Danny Hernandez, Scott Johnston, Andy Jones, Jackson Kernion, Liane Lovitt, Kamal Ndousse, Dario Amodei, Tom Brown, Jack Clark, Jared Kaplan, Sam McCandlish, and Chris Olah. In-context learning and induction heads. *Transformer Circuits Thread*, 2022. https://transformer-circuits.pub/2022/in-context-learning-and-induction-heads/index.html.

Liangming Pan, Jason Liang, Jiaran Ye, Minglai Yang, Xinyuan Lu, and Fengbin Zhu. Opening the black box: A survey on the mechanisms of multi-step reasoning in large language models. *arXiv preprint arXiv:2601.14270*, 2026.

Debjit Paul, Robert West, Antoine Bosselut, and Boi Faltings. Making reasoning matter: Measuring and improving faithfulness of chain-of-thought reasoning. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 15012–15032, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-emnlp.882. URL https://aclanthology.org/2024.findings-emnlp.882/.

Judea Pearl. Direct and indirect effects. In Jack S. Breese and Daphne Koller (eds.), *UAI '01: Proceedings of the 17th Conference in Uncertainty in Artificial Intelligence, University of Washington, Seattle, Washington, USA, August 2-5, 2001*, pp. 411–420. Morgan Kaufmann, 2001.

Bo Peng, Eric Alcaide, Quentin Anthony, Alon Albalak, Samuel Arcadinho, Stella Biderman, Huanqi Cao, Xin Cheng, Michael Chung, Leon Derczynski, Xingjian Du, Matteo Grella, Kranthi Gv, Xuzheng He, Haowen Hou, Przemyslaw Kazienko, Jan Kocon, Jiaming Kong, Bartłomiej Koptyra, Hayden Lau, Jiaju Lin, Krishna Sri Ipsit Mantri, Ferdinand Mom, Atsushi Saito, Guangyu Song, Xiangru Tang, Johan Wind, Stanisław Woźniak, Zhenyuan Zhang, Qinghua Zhou, Jian Zhu, and Rui-Jie Zhu. RWKV: Reinventing RNNs for the transformer era. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 14048–14077, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-emnlp.936. URL https://aclanthology.org/2023.findings-emnlp.936/.

Jacob Pfau, William Merrill, and Samuel R. Bowman. Let's think dot by dot: Hidden computation in transformer language models. In *First Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=NikbrdtYvG.

Abulhair Saparov and He He. Language models are greedy reasoners: A systematic formal analysis of chain-of-thought. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=qFVVBzXxR2V.

Benjamin Shih, John Winnicki, and Eric Darve. Do models read what they write? causal registers in scratchpad reasoning. *arXiv preprint arXiv:2606.29522*, 2026.

Alessandro Stolfo, Yonatan Belinkov, and Mrinmaya Sachan. A mechanistic interpretation of arithmetic reasoning in language models using causal mediation analysis. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 7035–7052, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.435. URL https://aclanthology.org/2023.emnlp-main.435/.

Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, et al. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*, 2024.

Miles Turpin, Julian Michael, Ethan Perez, and Samuel Bowman. Language models don't always say what they think: Unfaithful explanations in chain-of-thought prompting. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 74952–74965. Curran Associates, Inc., 2023. doi: 10.52202/075280-3275. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/ed3fea9033a80fea1376299fa7863f4a-Paper-Conference.pdf.

Jesse Vig, Sebastian Gehrmann, Yonatan Belinkov, Sharon Qian, Daniel Nevo, Yaron Singer, and Stuart Shieber. Investigating gender bias in language models using causal mediation analysis. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 12388–12401. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/92650b2e92217715fe312e6fa7b90d82-Paper.pdf.

Evan Pete Walsh, Luca Soldaini, Dirk Groeneveld, Kyle Lo, Shane Arora, Akshita Bhagia, Yuling Gu, Shengyi Huang, Matt Jordan, Nathan Lambert, Dustin Schwenk, Oyvind Tafjord, Taira Anderson, David Atkinson, Faeze Brahman, Christopher Clark, Pradeep Dasigi, Nouha Dziri, Allyson Ettinger, Michal Guerquin, David Heineman, Hamish Ivison, Pang Wei Koh, Jiacheng Liu, Saumya Malik, William Merrill, Lester James Validad Miranda, Jacob Morrison, Tyler Murray, Crystal Nam, Jake Poznanski, Valentina Pyatkin, Aman Rangapur, Michael Schmitz, Sam Skjonsberg, David Wadden, Christopher Wilhelm, Michael Wilson, Luke Zettlemoyer, Ali Farhadi, Noah A. Smith, and Hannaneh Hajishirzi. 2 OLMo 2 furious (COLM's version). In *Second Conference on Language Modeling*, 2025. URL https://openreview.net/forum?id=2ezugTT9kU.

Kevin Ro Wang, Alexandre Variengien, Arthur Conmy, Buck Shlegeris, and Jacob Steinhardt. Interpretability in the wild: a circuit for indirect object identification in GPT-2 small. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=NpsVSN6o4ul.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed Chi, Quoc V Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 24824–24837. Curran Associates, Inc., 2022. doi: 10.52202/068431-1800. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/9d5609613524ecf4f15af0f7b31abca4-Paper-Conference.pdf.

Wenhao Wu, Yizhong Wang, Guangxuan Xiao, Hao Peng, and Yao Fu. Retrieval head mechanistically explains long-context factuality. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=EytBpUGB1Z.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

Sohee Yang, Elena Gribovskaya, Nora Kassner, Mor Geva, and Sebastian Riedel. Do large language models latently perform multi-hop reasoning? In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 10210–10229, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.550. URL https://aclanthology.org/2024.acl-long.550/.

Fred Zhang and Neel Nanda. Towards best practices of activation patching in language models: Metrics and methods. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=Hf17y6u9BC.

Zheng Zhao, Yeskendir Koishekenov, Xianjun Yang, Naila Murray, and Nicola Cancedda. Verifying chain-of-thought reasoning via its computational graph. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=CxiNICq0Rr.

APPENDIX

Roadmap. The appendix follows the order of the main text. Sec. A1 gives the full intervention definitions, statistical conventions, prompts, and parsing procedures. Sec. A2 reports the copy-content controls, cross-model replications, and full generation-time results. Sec. A3 tests the effect in native thinking formats, with non-numeric values, across six model sizes, and in non-transformer architectures. Sec. A4 localizes the effect in depth and in attention heads. Sec. A5 reports the training experiments and the criteria for including from-scratch runs. Sec. A6 defines the monitoring metrics and separates coverage changes from changes in explicit provenance. Sec. A7 reports the outcomes of pre-specified extensions, additional intervention results, and enlarged figures.

A1 THE INSTRUMENT AND ITS VALIDATION

A1.1 INSTRUMENT: FULL DEFINITIONS AND VALIDATION

Notation. A trace is $T = (s_1, \ldots, s_m)$. Each step s_u has a token span $\pi(s_u)$ and a stated value $\mathrm{val}(s_u)$ whose token positions are g_u. The dependency graph expressed in the trace has edges $E_T \subseteq \{(u, v) : u < v\}$, taken from the template or from variable references parsed in self-generated text. The step metric averages the teacher-forced log-probability over the stated value's tokens,

$$m_v(T) \;=\; \tfrac{1}{|g_v|} \textstyle\sum_{t \in g_v} \log p_\theta(y_t \mid y_{<t}(T)) \,. \tag{3}$$

Restatement contrast. The restatement indicator is $r(u, v) = \mathbf{1}[\mathrm{val}(s_u) \in \mathrm{lits}(s_v)]$, where $\mathrm{lits}(s_v)$ contains the consumer's literal values, excluding variable names. For canonical exact-value renderings, the controlled comparison tests whether

$$\mathbb{E}[|\Delta_{u\to v}| \mid r = 1] \approx 0 \quad \ll \quad \mathbb{E}[|\Delta_{u\to v}| \mid r = 0]. \tag{4}$$

This is a textual indicator, not a complete test of semantic equivalence. The surface-form experiments in Sec. A2.4 test alternative ways of expressing the same value.

Rollout interventions and graph alignment. Rollout interventions regenerate the trace after the corrupted line and record downstream value changes: $\Delta^{\mathrm{tot}}_{u\to v} = \mathbf{1}[\mathrm{val}_v(\mathrm{regen}) \neq \mathrm{val}_v(T)]$. This quantity measures propagation through regenerated text rather than the teacher-forced source-patch effect. Alignment between the externalized graph and source-patch effects is the discrimination statistic

$$\mathrm{Align}(T) \;=\; \mathrm{AUROC}(\Delta[E_T],\ \Delta[\mathcal{N}_T])\,, \tag{5}$$

where $\Delta[S] = \{\Delta_{u\to v}\}_{(u,v)\in S}$. The set $\mathcal{N}_T$ contains no-path pairs matched to E_T on step distance, following the confound controls used in resampling studies (Bogdan et al., 2025). We compare AUROC with permutation-null distributions.

Scope and exceptions. The controlled arithmetic experiments support the absolute-effect contrast in Eq. 4. The observational summaries in Table A7 instead report signed means and show model-dependent reductions. Qwen3's observational restated edges retain an effect of 0.86 nats, still threefold below its non-restated edges, and Gemma-2-27B generates no restated edges in the one-shot condition. The contrast varies by model and is not uniformly near zero on restated edges. Without a copy, prompt, template, and training can change the source effect by factors of 2–10; its numerical scale should therefore be compared within a matched construction.

A weaker regularity concerns *copy availability*. Let $a(u, v) = 1$ when an intact, variable-bound textual copy of $\mathrm{val}(s_u)$ appears outside the patched span and before the value at consumer v is predicted. The copy-configuration comparison shows that such a copy reduces the mean signed effect even when it is remote. Among the tested arithmetic configurations, only the correct line-local copy reduces the mean absolute effect to near zero in the sense of Eq. 4.

Format effects under direct randomization. Crossing instruction I with rendered format R in a 2×2 design gives a copy effect of 0.39–0.47 nats, with 95% confidence intervals (CIs) $[0.26, 0.50]$ and $[0.35, 0.60]$. With a copy present, the instruction's controlled direct effect (CDE) (Pearl, 2001) is 0.0007, CI $[-0.001, 0.002]$. Without a copy, it is small but nonzero at 0.086, CI $[0.025, 0.152]$; the Qwen3 replication gives 0.002 against a copy effect of 1.82. We therefore detect no direct instruction effect when a copy is present in these conditions.

Additional validation. On variable-form templates, Gemma-2-2B distinguishes direct edges from distance-matched no-path pairs with AUROC 0.795 ± 0.013 (mean $\pm$ sample standard deviation over three data-generation seeds). The single-seed AUROCs for OLMo-2-7B and Qwen3-8B are 0.84 and 0.96, respectively. All three models have permutation-null $z \geq 26$. For the transformer experiments, an all-layer span patch under teacher forcing is mathematically equivalent to the corresponding text swap. Patching clean activations into the corrupted run should likewise restore the original prediction. All-layer restoration recovers the original log-probability exactly in the Gemma-2 validation. Restoring only layers 8–13 recovers a median 98% of the lost log-probability, measured on 878 direct edges with an absolute corruption effect of at least 0.1 nats. An attention-mask deletion reference, which removes u's span from later positions' attention, serves as a conflict-free check. On variable-form templates, its direct-edge effects are 1.35, 2.54, and 7.08 nats in Gemma-2, Qwen3, and OLMo-2 (no-path floors 0.29, 0.37, and 1.53), comparable to the source patch (1.18, 4.86, and 8.00). In prompt-generated arithmetic traces, deletion gives mean absolute effects of at most 0.008 nats on restated edges, versus 2.7–8.1 nats on non-restated edges, so the correct-copy contrast does not depend on creating a conflict.

Counterfactual generators. Three generators build counterfactual source lines of equal length. The consistent name-preserving generator keeps the defined variable, operator, and operand names and draws new operands with a consistent result; the dependency validation, the surface-form comparison,

the margin run in Sec. A1.2, and the specificity and self-generation replications in Sec. A1.3 use it, and the word, name, and PrOntoQA experiments substitute values in the same spirit. A second name-preserving generator resamples only numerals outside variable names and requires the line's own value to change, without keeping the line consistent; the other replications in Sec. A1.3 use it. The digit-resampling generator replaces every digit in the line, including variable indices; the remaining patching experiments use it, including the copy-configuration factorial, the observational and prompt-forced splits, the generation crosses, the joint and directional experiments, the native-format, scale, and architecture runs, the depth and head interventions, steering, and the self-generation experiment. With single-digit indices it changes the defined variable in most lines (927 of 1000 in the factorial traces), so these patches alter both the source value and the variable the line defines. The model-written-solution experiment edits values directly and keeps every name. Whole-line and numeral-only counterfactuals agree in sign, ordering, and separation but differ in scale (1.00 versus 0.36 nats in direct-edge mean). We therefore make every reported contrast within a construction.

A1.2 Saturation Checks for the Correct-Copy Headline

With a correct copy, the stated value typically remains highly probable both before and after source patching. Its log-probability can change little while unlikely alternatives undergo large relative changes. Table A1 reports two checks. The first restricts no-copy edges to confident clean predictions, $\log p(y_0) > -0.1$. Their mean absolute Δ exceeds the all-edge mean, so low confidence without a copy does not produce the headline reduction. The second uses a separate run on the same problems whose counterfactual source lines keep the variable name and are arithmetically consistent, and records the change in the margin $\log p(y_0) - \log p(y_{\mathrm{cf}})$ together with the decoded value. With a correct copy, source patching leaves the decoded value unchanged on at least 99.3% of edges but raises $\log p(y_{\mathrm{cf}})$ by 2.7–6.2 nats, while y_{cf} keeps less than 1% of the probability mass. On the margin, the copy reduces the mean absolute source effect by 55–79% relative to confident no-copy edges; relative to all no-copy edges, the reduction is 51–62% in Qwen3 and OLMo-2 and absent in Gemma-2, whose no-copy predictions are least often confident (7% of edges). A first-token check in Gemma-2 agrees: with a copy, source patching raises the counterfactual-versus-original logit margin by 3.45 nats, compared with 4.31 on baseline-correct no-copy edges. A random-value control in Sec. A1.3 shows that this gain is not specific to y_{cf}: random values of matched length rise about as much on average.

	mean $\lvert\Delta\rvert$ on $\log p(y_0)$			mean $\lvert$margin change$\rvert$			margin reduction (%)		correct copy present		
Model	no copy	conf.	copy	no copy	conf.	copy	all	conf.	y_{cf} gain	y_{cf} mass	kept
Gemma-2-2B	1.33	2.42	0.009	2.36	6.07	2.74	-16 [-25, -8]	55 [49, 59]	2.73	0.005	0.993
Qwen3-8B	2.33	2.53	0.011	10.71	12.28	5.28	51 [48, 53]	57 [55, 59]	5.27	0.007	0.997
OLMo-2-7B	6.87	9.42	0.001	16.64	30.18	6.25	62 [58, 66]	79 [77, 81]	6.22	0.000	1.000

Table A1: **Saturation checks for the correct-copy headline.** Mean absolute effects in nats; "conf." restricts no-copy edges to clean $\log p(y_0) > -0.1$. The $\lvert\Delta\rvert$ columns use the run behind Table A11; the margin columns use a separate run on the same problems with name-preserving, arithmetically consistent counterfactual source lines (919–922 edges per cell), in which $\lvert\Delta\rvert$ falls from 1.21–8.23 nats without a copy to at most 0.018 with one. Margin reductions give 95% trace-cluster bootstrap intervals. The last three columns report, with a correct copy present, the mean gain in $\log p(y_{\mathrm{cf}})$, the probability mass on y_{cf} after patching, and the fraction of edges whose decoded value is unchanged.

A1.3 Name-Preserving Replications

The digit-resampling counterfactual changes both a source's value and, usually, the variable it defines (Sec. A1.1). We therefore repeat four analyses with name-preserving counterfactuals in bfloat16 on one A100 (two for R1-Distill-Llama-70B) with the pinned model revisions of Sec. A1.5. The patched forward pass equals a forward pass on the counterfactual text, so these runs edit the text directly.

Specificity of the margin shift. The first replication rebuilds the edges of the margin run in Sec. A1.2 and scores two random control values per edge, with the same digit count and number of value tokens as y_{cf} and absent from both the original and the counterfactual source line. Without a copy, source patching raises $\log p(y_{\mathrm{cf}})$ by 1.0–7.6 nats more than it raises the control values

(Table A2). With a correct copy, y_{cf} and the controls rise by similar amounts, and their difference lies between -0.57 and 0.15 nats. The residual margin shift under a copy therefore reflects a general rise of improbable values, not source-specific information.

Model	Condition	n	gain of y_{cf}	gain of random values	difference
Gemma-2-2B	no copy	919	1.12 [1.02, 1.23]	0.12 [0.08, 0.16]	1.00 [0.90, 1.10]
	correct copy	922	2.72 [2.64, 2.81]	2.80 [2.72, 2.88]	-0.07 [-0.12, -0.02]
Qwen3-8B	no copy	919	5.84 [5.54, 6.14]	0.92 [0.78, 1.07]	4.91 [4.68, 5.15]
	correct copy	922	5.25 [5.02, 5.48]	5.11 [4.87, 5.33]	0.15 [0.08, 0.22]
OLMo-2-7B	no copy	919	8.39 [7.82, 8.98]	0.77 [0.51, 1.01]	7.62 [7.11, 8.15]
	correct copy	922	6.22 [5.81, 6.63]	6.79 [6.36, 7.20]	-0.57 [-0.71, -0.43]

Table A2: **Specificity of the log-probability gain under source patching.** Mean gain in nats with trace-clustered 95% CIs for the counterfactual-implied value y_{cf} and for the mean of two random control values of matched length; the last column is their per-edge difference. Counterfactual source lines keep the variable name and are arithmetically consistent.

Copy configurations. The second replication repeats the configuration factorial with a counterfactual that resamples only numerals outside variable names and requires the source's own value to change. It adds two configurations that restate a single operand on the consumer line, such as `x5 = x3 - x1 = 23 - x1 = 7`, and records distance-unmatched no-path floors from later lines that do not depend on the source. The correct local copy again leaves at most 1.8% of the no-copy effect, remote duplicates leave 32–63%, and filler and duplicates placed after the consumer leave nearly all of it (Table A3). A wrong local copy leaves 9–15% of the absolute effect and no signed effect, whereas under digit resampling it leaves 18–32% and a signed effect (Table A11); a wrong value on the consumer line thus suppresses the source more than a correct duplicate elsewhere. A local copy of the source operand alone leaves 17–52%, whereas a local copy of the other operand alone leaves the source effect intact or larger. Complete suppression thus requires the consumer line to carry all its operands, but a local copy of the source value alone removes most of the effect in Qwen3 and OLMo-2.

Configuration	Gemma-2-2B	Qwen3-8B	OLMo-2-7B
no copy (nats)	1.25	3.51	8.68
correct local copy	1.8	1.4	0.0
wrong local copy	15.0	8.9	15.3
partial-digit local copy	18.5	—	—
local copy of the source operand only	51.6	16.7	21.7
local copy of the other operand only	186.4	98.9	137.3
length-matched filler	133.9	91.2	114.2
correct remote duplicate	63.3	31.5	52.4
wrong remote duplicate	36.8	—	—
mis-bound remote duplicate	75.9	—	—
bare number after the source	82.5	—	—
duplicate two steps later	80.6	—	—
duplicate after all consumers	99.9	—	—
no-path floor, no copy (nats)	0.296	0.475	1.318
no-path floor, correct copy (nats)	0.0012	0.0029	0.0001

Table A3: **Copy configurations under a name-preserving counterfactual.** The first row gives the no-copy mean absolute source effect in nats; the other configuration rows give the mean absolute effect as a percentage of that row. Single-operand rows pool the two single-operand renderings by whether the source is the restated operand. Gemma-2 runs all configurations on 100 traces; Qwen3 and OLMo-2 run the core configurations on 80 traces. No-path floors are mean absolute effects on later internal lines that do not depend on the source.

Self-generated copies. The third replication repeats the G3 self-generation experiment (Sec. A2.8) with a name-preserving, arithmetically consistent counterfactual on the same 150 traces, in Gemma-2, Qwen3, and OLMo-2. Source patching now makes the models write the counterfactual copy in 81–94% of samples and the counterfactual result in 96–99% (Table A4), compared with 23.5–32% and 26–34% when the counterfactual also renames the source variable. The low propagation in G3 therefore reflected the lost binding rather than a failure to propagate a changed value.

Outcome (source patched)	Gemma-2-2B	Qwen3-8B	OLMo-2-7B
original target copy	0.000 [0.000, 0.000]	0.000 [0.000, 0.000]	0.000 [0.000, 0.000]
counterfactual target copy	0.813 [0.789, 0.836]	0.940 [0.922, 0.957]	0.938 [0.924, 0.953]
other extracted target copy	0.001 [0.000, 0.004]	0.003 [0.000, 0.007]	0.001 [0.000, 0.004]
no extracted target copy	0.186 [0.163, 0.210]	0.058 [0.041, 0.076]	0.060 [0.046, 0.074]
original result	0.001 [0.000, 0.004]	0.000 [0.000, 0.000]	0.000 [0.000, 0.000]
counterfactual result	0.980 [0.969, 0.989]	0.957 [0.942, 0.971]	0.991 [0.984, 0.996]
other or missing result	0.019 [0.010, 0.028]	0.043 [0.029, 0.058]	0.009 [0.004, 0.016]
digit-resampling counterfactual			
counterfactual target copy	0.235 [0.207, 0.262]	0.263 [0.233, 0.292]	0.323 [0.295, 0.353]
counterfactual result	0.271 [0.242, 0.302]	0.263 [0.234, 0.293]	0.339 [0.310, 0.369]

Table A4: **Models write the consumer line after a name-preserving source patch.** Proportions in the patched arm with trace-clustered 95% CIs over 747 samples per model; categories are defined as in Table A18. In the clean arm, every model writes the original target copy in at least 98.9% and the original result in at least 99.3% of samples. The last two rows repeat the experiment with the digit-resampling counterfactual, which usually renames the source variable (741 samples per model).

Generation crosses, scale series, and other architectures. The fourth replication reruns the source–copy generation crosses of Table 1, the six-model series of Table A23, and the non-transformer checks of Sec. A3.4 with the same name-preserving counterfactual (Table A5 and Table A6). Δ_S stays within 0.011 of its digit-resampling value in every model and reaches 0.140 in R1-Distill-Llama-70B, now with a trace-clustered 95% CI of $[0.111, 0.177]$; CDE_0 stays at most 0.029. With a correct local copy, the native-format source effect falls to 0.4–20% of its no-copy value. In Mamba2, the mean absolute effect falls from 1.04 to 0.053 nats with a correct local copy, and the share of negative per-edge effects falls from 37% to 4%, so the signed reduction in Sec. A3.4 is not an artifact of cancellation. In RWKV, the correct local copy removes the signed effect (0.143 to -0.001 nats) but leaves 56% of the mean absolute effect (0.455 to 0.254 nats), and 35–39% of per-edge effects are negative in both conditions, so the near-zero signed effect in RWKV mostly reflects cancellation, and the copy only partly suppresses the source.

			cf. result, $C = 1$					
Model	Format	$\lvert\Delta\rvert$ rest. / non-rest.	S clean	S patched	Δ_S [95% CI]	Δ_S digits	CDE_0	n/cell
Gemma-2-2B	template	—	0.893	0.894	0.001 [0.000, 0.003]	0.000	-0.001	1000
Qwen3-8B	template	—	0.905	0.910	0.005 [0.001, 0.010]	0.002	0.001	1193
Qwen3-8B	<think>	0.023 / 3.70	0.878	0.910	0.032 [0.023, 0.042]	0.021	0.003	1193
R1-Qwen-1.5B	<think>	0.319 / 1.61	0.764	0.801	0.036 [0.021, 0.051]	0.037	0.029	1193
R1-Qwen-7B	<think>	0.047 / 3.26	0.877	0.899	0.023 [0.014, 0.032]	0.021	0.010	1193
R1-Llama-8B	<think>	0.073 / 6.65	0.898	0.925	0.027 [0.016, 0.038]	0.025	0.000	1193
R1-Qwen-14B	<think>	0.034 / 3.18	0.860	0.909	0.049 [0.037, 0.064]	0.054	0.001	1193
R1-Qwen-32B	<think>	0.024 / 3.00	0.861	0.909	0.049 [0.034, 0.066]	0.057	0.003	595
R1-Llama-70B	<think>	0.027 / 6.87	0.777	0.917	0.140 [0.111, 0.177]	0.146	0.003	399

Table A5: **Generation crosses and native-format source effects under a name-preserving counterfactual.** The third column gives mean absolute source-patch effects (nats) on restated and non-restated edges in native <think> renderings. Counterfactual-result frequencies use a counterfactual copy ($C = 1$) with a clean or patched source; Δ_S has a trace-clustered 95% CI, and "Δ_S digits" repeats the digit-resampling value from Table 1 or Table A23. CDE_0 is the loss of the original result under source patching with a correct copy fixed.

	no copy			correct local copy		
Model	signed	mean $\lvert\Delta\rvert$ [95% CI]	neg.	signed	mean $\lvert\Delta\rvert$ [95% CI]	neg.
Mamba2-2.7B	0.535	1.037 [0.949, 1.134]	0.37	0.039	0.053 [0.043, 0.065]	0.04
RWKV-4-3B	0.143	0.455 [0.415, 0.499]	0.39	-0.001	0.254 [0.219, 0.294]	0.35

Table A6: **Non-transformer architectures under a name-preserving counterfactual.** Mean signed and mean absolute source-patch effects (nats) on direct edges, with trace-clustered 95% CIs, and the fraction of negative per-edge effects.

A1.4 Statistical Detail: Sample Sizes, Clustered Confidence Intervals, and Predictor Baselines

Model	restated	non-restated	restated share	AUROC
Gemma-2-2B	0.016	1.95	93%	0.744
OLMo-2-7B	0.0004	7.30	80%	0.620
R1-Distill-7B	0.03	2.52	69%	0.765
Qwen3-8B	0.86	2.70	11%	0.933
Gemma-2-27B	—	3.00	0%	0.989

Table A7: **Observational split in self-generated arithmetic traces.** Source-patch effects are signed means in nats. AUROC distinguishes direct edges from no-path pairs within distance strata. Rows are ordered by the share of parsed direct edges that restate the source value. The prompt induces different format preferences across models, so the source-effect contrast is interpreted within each model. Gemma-2-27B generates no restated edges in this condition; its copy contrast is instead tested with assigned templates (0.003 versus 2.73 nats).

Edges within a trace share a prompt and are not independent. Our default for edge-level analyses is a nonparametric bootstrap over traces (2,000 cluster resamples; percentile 95% intervals), retaining all edges from each sampled trace. Exceptions are identified with their results: the original Gemma-2 generation cross uses Wilson intervals, and the 32B/70B summary-only generation runs use normal approximations. Those summaries do not retain the trace clustering needed for cluster-aware generation intervals; their nominal intervals should not be treated as equivalent to the clustered estimates. Training plots show individual seeds rather than sampling confidence intervals. Bootstrap intervals for outcomes never observed in the saved traces can be degenerate; they do not establish a zero population rate. Table A8 gives per-cell sample sizes, trace counts, and clustered CIs for the central cells of Table A7 and for the variable-form prompt conditions in Table A42; its caption summarizes the negative-effect rates. The restated and non-restated intervals are well separated in Gemma-2 and OLMo-2. Qwen3's intervals also do not overlap, but the source effect remains appreciable on restated edges. Its restated cell has the widest interval (0.86 [0.51, 1.51]), based on only 153 edges whose effects have a heavy-tailed distribution. This cell is also the main observational exception to the otherwise near-zero correct-copy effects.

Cell	mean	n	traces	95% CI
Observational (self-generated, one-shot; Table A7)				
Gemma-2 rest.	0.016	1791	185	[0.011, 0.022]
Gemma-2 non-rest.	1.95	137	31	[1.71, 2.19]
OLMo-2 rest.	0.0004	1442	157	[0.0001, 0.0009]
OLMo-2 non-rest.	7.30	349	27	[6.54, 8.60]
Qwen3 rest.	0.86	153	53	[0.51, 1.51]
Qwen3 non-rest.	2.70	1187	127	[2.62, 2.78]
Variable-form prompts, non-restated edges (Table A42)				
Gemma-2	2.37	290	—	[2.20, 2.54]
Qwen3	2.68	1328	—	[2.60, 2.78]
OLMo-2	7.20	117	—	[6.18, 9.23]

Table A8: Trace-clustered bootstrap 95% CIs for the mean signed effects in the central cells of Table A7 and Table A42. Negative per-edge effects are rare ($\leq 3\%$) except in OLMo-2's restated cell. There, 12% are marginally negative, but all magnitudes are below 10^{-2} nats and consistent with the measurement floor.

Text-only baselines for the predictor. We compare a fixed restatement rule with edge distance and clean-token confidence. This task predicts a large measured source effect, rather than the presence of a dependency edge as in the graph-validation task above. The positive label is a signed source-patch effect of at least 0.25 nats, and the rule predicts this label when the source value is not restated. The target is a direct-edge consumer except in GSM8K, where it is the final parsed result. The GSM8K restatement feature records whether the source value appears again anywhere later in the trace.

AUROC gives half credit to equal positive and negative scores. For the rule, the score is fixed at $1 - r$, with no fitted orientation or threshold. Balanced accuracy is $\frac{1}{2}[\mathrm{TP}/(\mathrm{TP} + \mathrm{FN}) + \mathrm{TN}/(\mathrm{TN} + \mathrm{FP})]$, where positives satisfy the source-effect threshold. For this binary score, AUROC equals the rule's balanced accuracy in Table A19. We report either metric as undefined if an actual label class is absent.

The rule generally exceeds distance in AUROC (Table A9). The exception is Gemma-2 GSM8K: rule AUROC is 0.67 versus 0.69 for distance. For Qwen3 GSM8K, the rule's balanced accuracy is 0.79, compared with 0.75 for the oracle-selected distance threshold. Clean-token confidence can have high AUROC, but it requires a model forward pass and is one term in the source-effect definition. We include it as a model-based reference. Tables A9 and A19 use identical samples and are generated from raw records by `01-code/summarize_metrics.py`, which records input paths and hashes. For GSM8K, these are the original predictor subsets; the pooled effect estimates in Table A41 additionally include later Gemma-2 and OLMo-2 runs.

Domain	rule	dist.	dist.*	conf.
Arithmetic self-generated	0.97 / 0.99 / 0.86	0.58 / 0.57 / 0.56	0.58 / 0.57 / 0.56	1.00 / 0.98 / 0.83
Arithmetic variable-form prompt	0.99 / 0.99 / 0.64	0.60 / 0.54 / 0.52	0.58 / 0.58 / 0.55	0.99 / 0.98 / 0.61
k-hop	0.79 / 0.75 / 0.82	0.53 / 0.56 / 0.58	0.53 / 0.55 / 0.58	0.87 / 0.76 / 0.82
GSM8K	0.67 / 0.87 / 0.79	0.69 / 0.71 / 0.75	0.68 / 0.68 / 0.75	0.71 / 0.76 / 0.65

Table A9: Predictor baselines for source-patch effects of at least 0.25 nats; each cell reports Gemma-2 / OLMo-2 / Qwen3. The target is a direct-edge consumer except in GSM8K, which uses the final parsed result and groups sources by later value reappearance. "rule" is AUROC for the fixed score $1 - r$; "dist." is AUROC for edge distance (a source-step-position proxy for GSM8K); "dist.*" is balanced accuracy at an oracle-selected distance threshold; and "conf." is AUROC for clean-token confidence. Distance and confidence select the better score orientation within each dataset; the rule keeps its pre-specified orientation. Sample counts appear in Table A19.

A1.5 Prompts, Templates, and Parsing

The arithmetic templates are directed acyclic graphs formed from binary trees of depth 3–4 over $\{+, -, \times\}$, with values below 1000. The released code contains the exact one-shot and format-forcing prompts, the length-matched filler renderer, the trace parser, and validity statistics for each condition. The parser specifies the line grammar, extracts dependencies, and identifies restated values while excluding variable names. Counterfactual lines preserve token counts by digit- and name-matched resampling; traces failing token-alignment assertions are skipped and counted.

Free-format restatement rates. On the arithmetic task, Gemma-2 restates 650 of 654 direct edges under free-format generation (99.4% across 61 parsed traces), and Qwen3 restates all 2000 edges across 200 traces. OLMo-2's free-format traces do not match our grammar and are excluded; its one-shot restatement rate is 80%.

Reproducibility settings. The Hugging Face model IDs are `google/gemma-2-2b-it`, `google/gemma-2-27b-it`, `allenai/OLMo-2-1124-7B-Instruct`, `Qwen/Qwen3-8B`, and `deepseek-ai/DeepSeek-R1-Distill-Qwen-7B`. We use the Hub revisions current in July 2026, which are the latest commits of each repository: `gemma-2-2b-it` `299a856`, `gemma-2-27b-it` `aaf20e6`, `OLMo-2-1124-7B-Instruct` `470b1fb`, `Qwen3-8B` `b968826`, `Qwen/Qwen3-32B` `9216db5`, `deepseek-ai/DeepSeek-R1-Distill-Qwen-{1.5B,7B,14B,32B}` `ad9f0ae`/`916b56a`/`1df8507`/`711ad2e`, `deepseek-ai/DeepSeek-R1-Distill-Llama-{8B,70B}` `6a6f4aa`/`b1c0b44`, `AntonV/mamba2-2.7b-hf` `ef54270`, and `RWKV/rwkv-4-3b-pile` `7fdda3c`. We use Transformers 5.14.1, PyTorch cu130 wheels from July 2026, and PEFT 0.19.1. Evaluation generation is greedy (`do_sample=false`); expert-iteration training candidates are sampled at $T = 0.8$ as specified below. The `max_new_tokens` values are 8 for value-position cells, 30 for single-line regeneration, 200–420 for self-generation, and 320 for GSM8K. All LoRA arms use $r = 16$, $\alpha = 32$, dropout

0.05, and the q/k/v/o/gate/up/down projection targets. We train with AdamW at learning rate 10^{-4} for two epochs, using batch size 4 and gradient accumulation 4. Each format arm contains 3,000 pairs, except the brevity-selected arm, which contains 549. GSM8K parsing is conservative: the enlarged Gemma-2 runs recover 174 answer-dependency records from 600 problems. The released result files report valid and parsed counts for every run.

Reward-guided expert iteration under length pressure. We run three rounds on Gemma-2-2B, sampling $k = 4$ solutions at $T = 0.8$ for each of 400 problems per round. The generic arm keeps the shortest correct solution and applies LoRA SFT to the selected set. The targeted arm uses the same loop but selects the least-restating correct sample. We evaluate held-out traces after each round for accuracy, source sensitivity, and the information available to a checker. The selection outcomes appear together with the monitoring comparisons in Sec. A6.2.

Manual parser audit. We manually compared raw generations with the parser's extraction. The synthetic-task audit found no extraction errors in 20 Gemma-2 generations containing 190 direct edges. The parser correctly stops at the first generated answer and preserves lines in which the stated literal differs from the value of the referenced variable. For GSM8K, we audited 40 generations and all 28 extracted step lines. The parser identifies 26 line results correctly; both errors treat a trailing price constant ("for $2.50") as the result. Among 48 later-use string matches, one is coincidental because a divisor shares the value's digits, and one true restatement is missed because of a decimal-format variant ("10" versus "$10.00"). These mismatches limit the precision of the small non-restated subsets. The audit is too small to determine the direction or magnitude of their effect on the contrast.

Ethics statement. All experiments use synthetic templates, the public PrOntoQA generator, or the public GSM8K dataset; no human subjects, personal data, or crowdworkers are involved. The monitoring analysis characterizes the information available to a line-local checker. Its practical implication is defensive: developers can audit whether training selection reduces explicit operand provenance before deployment.

Compute. Pretrained models run in bfloat16 without quantization, and activation-patching runs use eager attention; the from-scratch models train in fp32. Every model runs on a single A100-80GB GPU except DeepSeek-R1-Distill-Llama-70B, which is sharded across two.

A2 Establishing the Substitution Effect: Controls and Replications

A2.1 Length and Copy-Content Controls

Table A10 compares five copy configurations on one Gemma-2-2B problem set (n=1000 teacher-forced direct edges per condition). The configurations test copy presence, copy correctness with digit-count matching, value-free length-matched filler, and a remote duplicate of the correct value. For the remote condition, we insert "`(x3 = 28)`" after the source line while keeping consumer lines in variable form; the intervention patches only the original source span.

Template	mean	median	IQR
correct local copy (restate)	0.006	0.001	$[.001, .003]$
remote duplicate	0.042	0.019	$[-0.57, 0.56]$
no copy (variable-form)	0.362	0.344	$[-0.46, 1.52]$
wrong local copy	0.380	0.327	$[0.09, 0.58]$
value-free filler	1.281	1.243	$[0.32, 2.23]$

Table A10: Signed source-patch effects (nats) and interquartile ranges (IQRs) under five teacher-forced templates for one Gemma-2-2B problem set. The correct local copy reduces both signed and absolute effects to near zero (mean $|\Delta| = 0.009$). The remote duplicate has a near-zero mean signed effect (0.042, CI $[-0.03, 0.11]$) but a substantial mean absolute effect (0.81, CI $[0.76, 0.86]$): 27% of edges exceed $+0.5$ and 28% fall below -0.5. Wrong local copies retain predominantly positive effects (32% above $+0.5$ and 0.7% below -0.5).

The shared conditions reproduce an earlier length-control run, with source effects of 0.360, 0.004, and 1.291 nats for variable-form, correct-copy, and filler traces, respectively. The filler's larger effect than variable-form traces suggests that placeholder positions may provide useful computation slots (Pfau et al., 2024).

A2.2 Copy Configurations and Cross-Model Replication

We combine five core cells with six follow-up configurations in Gemma-2-2B. Table A10 reports the core cells, and Fig. 2 visualizes all eleven configurations. The follow-ups add a bare number, a copy with the wrong variable binding, a copy placed after all consumers, a partial-digit copy, a wrong remote duplicate, and a delayed duplicate. For a consumer $x_5 = x_3 - x_1$ with $x_3 = 28$ and $x_1 = 16$, the no-copy line is `x5 = x3 - x1 = 12` and the correct local copy is `x5 = x3 - x1 = 28 - 16 = 12`. A wrong local copy replaces both operands with random values of the same digit count, a partial-digit copy changes one digit of each, and length-matched filler writes `~~ - ~~` in their place. The remote duplicate adds the line `(x3 = 28)` immediately after the source line, the bare number adds the line `28`, and the mis-bound duplicate adds `(y3 = 28)`; the consumer stays in variable form. The after-consumer and delayed conditions move the bound duplicate after every consumer or two steps later. The first four give mean signed / mean absolute effects of $0.374/1.289$, $0.219/0.914$, $0.341/1.334$, and $0.310/0.389$ nats, respectively (n=1000 per condition). The matched runs use the same renderer; where the core and follow-up runs repeat the remote-duplicate cell, its signed effect is 0.042 and 0.073, respectively.

A bound but wrong remote duplicate has nearly the same mean signed effect as a correct remote duplicate (0.075 versus 0.073), while retaining a mean absolute effect of 0.53 nats. Both correct and wrong remote duplicates therefore retain sensitivity that their small signed means alone would obscure. Neither reproduces the near-zero absolute effect of a correct local copy. In the tested delay condition, a bound duplicate placed two steps later gives a signed effect of 0.089 when it still precedes the consumer (n=543), compared with 0.427 when it follows the consumer (n=457). This contrast concerns whether the duplicate is available at prediction time; it does not establish general insensitivity to distance. The five core configurations replicate across models (Table A11).

Cell	Gemma-2-2B	Qwen3-8B	OLMo-2-7B
no copy	.362 / 1.33	1.89 / 2.33	3.40 / 6.87
correct local	.006 / .009	.011 / .011	.001 / .001
value-free filler	1.28 / 1.50	1.40 / 2.39	5.51 / 7.01
wrong local	.380 / .430	.234 / .415	1.62 / 2.07
remote duplicate	.073 / .797	.290 / .809	.930 / 3.64

Table A11: Mean signed / mean absolute source effects (nats) for the five core configurations. The reduction with a correct local copy replicates across all three models (n=800 per cell for Qwen3 and OLMo-2). Correct local copies reduce both effects to near zero; remote duplicates reduce the signed mean but retain substantial absolute effects; and wrong local copies retain both. The signed reduction from a remote duplicate is strongest in Gemma-2 and Qwen3 and partial in OLMo-2 (0.93 versus a no-copy effect of 3.40). The correct-local absolute effects have trace-clustered 95% CIs $[.004, .016]$, $[.007, .015]$, and $[.0002, .003]$; this run stores no no-path pairs, and distance-matched no-path effects in model-generated restated traces are at most 0.005 nats.

A2.3 Instruction-by-Copy Randomized Factorial

We cross the prompt-level style instruction I with the assigned trace format R on identical problems (Gemma-2-2B, n=800 direct edges per cell).

A2.4 Surface-Form Comparisons

We render the same arithmetic DAGs with eight consumer formats: no copy, exact numerals, number words, decimals, currency amounts, zero-padded numbers, spaced digits, and arithmetic expressions. For each model and style, the saved run contains 1200 direct edges. The comparison in Sec. 4 uses 720 edges with leaf sources, drawn from 120 traces. This filter keeps the source-line text identical

	R = restated	R = variable-form
I = restatement instruction	0.0025	0.388
I = variable-form instruction	0.0032	0.474

Table A12: Mean signed source-patch effects (nats) under independently assigned instructions and trace formats. The main effect of copy availability is 0.43 nats. The instruction's controlled direct effect is 0.0007 on restated traces and 0.086 on variable-form traces. Thus, the instruction has no detectable direct effect when a copy is present and modestly increases source sensitivity when no copy is present.

across styles; every condition uses the same trace–source–consumer triples. We report the mean absolute source-patch effect, not the signed mean.

The source and consumer variable identities, problem graph, and underlying values are held fixed. Source counterfactuals preserve token count within a style. The inserted strings differ across styles, so their token counts and downstream positions also differ. This experiment compares the tested renderings; it does not isolate an independent effect of normalization difficulty, lexical familiarity, or token length. The separate filler factorial controls length for its own copy-content comparison, not for these eight forms.

Local form	Gemma-2-2B	Qwen3-8B	OLMo-2-7B
No copy	1.292 [1.197, 1.391]	2.191 [2.054, 2.345]	6.110 [5.793, 6.441]
Exact numeral	0.003 [0.002, 0.003]	0.008 [0.006, 0.010]	0.0001 [0.0001, 0.0002]
Number words	0.026 [0.012, 0.043]	0.071 [0.051, 0.093]	0.055 [0.016, 0.101]
Decimal	0.002 [0.002, 0.002]	0.013 [0.009, 0.021]	0.0002 [0.0001, 0.0002]
Currency	0.005 [0.005, 0.006]	0.022 [0.017, 0.029]	0.004 [0.003, 0.005]
Zero-padded	0.345 [0.312, 0.381]	0.222 [0.187, 0.260]	0.905 [0.797, 1.017]
Spaced digits	0.508 [0.439, 0.581]	0.198 [0.162, 0.235]	1.775 [1.630, 1.922]
Arithmetic expression	0.434 [0.402, 0.469]	0.978 [0.884, 1.076]	1.536 [1.418, 1.661]

Table A13: **Source sensitivity across numeric surface forms.** Mean absolute effect in nats with trace-cluster bootstrap 95% intervals (2000 resamples). Every cell contains the same 720 leaf-source edges from 120 traces per model. Near-zero entries are rounded; lengths are matched within each clean/counterfactual pair, not across rendering styles. The full precision and selection checks are reproduced by `01-code/summarize_main.py`.

For person-name values, additional conditions uppercase the name, prepend the honorific *Dr.*, or substitute an alias supplied in the prompt. These name-domain checks supplement the numeric surface-form comparison.

A2.5 Generation Under a Wrong Local Copy

In experiment J2, we write a wrong value b' into the local copy and either leave the remote source clean or patch its activations to a counterfactual a'. The original source, patched source, and wrong copy imply three distinct candidate consumer results. Table A14 reports their frequencies among generated consumer values for Gemma-2-2B (n=904 edges). Distinct candidates separate behavioral outcomes; the frequencies still do not directly measure internal route use.

Condition	original	wrong-copy	counterfactual	other
clean copy, clean source	.997	.000	.000	.003
wrong copy, clean source	.002	.903	.000	.095
wrong copy, patched source	.002	.904	.001	.093

Table A14: Frequencies of original, wrong-copy-implied, counterfactual-source-implied, and other results in Gemma-2-2B (n=904 edges). The wrong-copy result dominates with either source setting. The likelihood-level source effect in Table A39 therefore does not produce a corresponding shift in decoded results in this setting.

A2.6 SOURCE–COPY GENERATION CROSS: FULL CELLS

This section reports the full source×copy generation cross for Gemma-2-2B. The template experiment uses greedy value decoding with the consumer prefix supplied ($n = 976$ edges per cell); it does not use the separate problem block of the self-generation experiment below. Let $S, C \in \{0, 1\}$ denote original or counterfactual source and copy settings. The original result is y_0, and the counterfactual result is y_1. We define

$$\mathrm{CDE}_0 = P(Y = y_0 \mid S = 0, C = 0) - P(Y = y_0 \mid S = 1, C = 0), \tag{6}$$

$$\Delta_S = P(Y = y_1 \mid S = 1, C = 1) - P(Y = y_1 \mid S = 0, C = 1). \tag{7}$$

Thus CDE_0 measures loss of the original result with a correct copy fixed, whereas Δ_S measures gain of the counterfactual result with a counterfactual copy fixed. In the $S = C = 1$ cell, both occurrences imply y_1, so a matching output cannot distinguish their use. Other outcomes include unparsed generations and values matching neither candidate; all remain in the denominator.

Source	Copy	original result	cf. result	other	95% CI
clean	clean	0.998	0.000	0.002	[.993, .999]
clean	cf	0.002	0.891	0.107	[.870, .909]
patched	clean	0.998	0.000	0.002	[.993, .999]
patched	cf	0.003	0.891	0.106	[.870, .909]

Table A15: Generated consumer values under assigned source and copy settings. The columns report original, counterfactual (cf.), and other result frequencies. The 95% Wilson interval is for the original-result frequency in correct-copy rows and the counterfactual-result frequency in counterfactual-copy rows. Counterfactual-result frequency is 0.891 in both source states, and both $|\mathrm{CDE}_0|$ and $|\Delta_S|$ are at most 0.003. An independent replication with trace identifiers (n=1166 per cell) gives counterfactual-result frequencies of 0.901 [.881, .919] with a clean source and 0.903 [.883, .922] with a patched source (trace-clustered bootstrap 95% CIs). With a correct copy, CDE_0 has CI $[-.003, +.003]$ and passes the two one-sided tests (TOST) procedure at the pre-specified 0.05 equivalence margin; with a counterfactual copy, Δ_S has CI $[-.001, +.007]$.

In a Qwen3 native-format rerun that saves trace identifiers (n=1166 per cell), the corresponding intervals are [.000, .003] for CDE_0 and [.014, .028] for Δ_S. For Qwen3-8B, the following pairs give clean-source / patched-source frequencies. The template replication gives original-result frequencies of $0.997/0.998$ with a correct copy and counterfactual-result frequencies of $0.903/0.905$ with a counterfactual copy; $|\mathrm{CDE}_0|$ is at most 0.004. Inside Qwen3's native `<think>` format, these values are $1.000/0.999$, $0.886/0.907$, and at most 0.009, respectively. Across the six native-format models, Δ_S is positively associated with size (Spearman $\rho = .83$), but the series mixes Qwen and Llama base architectures and is not monotonic below 10B (Table A23). GSM8K uses a different corruption and candidate construction, described in Sec. A3.2; its results appear in Table A22.

A2.7 SOURCE–COPY CONFLICTS IN MODEL-WRITTEN SOLUTIONS

This exploratory experiment (NC), which was not among the pre-specified extensions, applies the source–copy cross to solutions that each model writes itself and follows every edit to the final answer. Unlike the G3 experiment of Sec. A2.8, the edited copy is supplied text: the model generates only from the consumer's result onward.

Solutions. Each model first writes a greedy solution. GSM8K prompts use the eight-shot format of G6 (Sec. A3.2) or, for `<think>` runs, the chat template with the question and a request for a boxed answer. Arithmetic problems are the variable-DAG problems of Sec. 3. Their few-shot prompts request restated operands, and their `<think>` prompts request one `xN = expression = value` line per step, so these solutions are written under a format request. Solutions are limited to 320 (GSM8K few-shot), 400 (arithmetic few-shot), or 2048 (`<think>`) new tokens. Solutions without an extractable final answer, including `<think>` solutions that do not close the reasoning block, are excluded. Each run processes problems in a fixed order until a preset time budget or problem cap, so problem counts differ across settings.

Edges and edits. We parse every binary calculation whose written result is correct for its written operands, including prose forms such as "16 eggs minus 7 eggs equals 9"; in <think> runs we parse only the thinking text. An edge links a source calculation with result a to the first calculation on a later line that uses a as an operand, and we skip it if another calculation between them also produces a. We also require a to be an integer with $a \geq 2$ that differs from both operands of its own calculation, and nonnegative integer results for every edited value. Each problem contributes at most three edges. The source occurrences are a and its restatements on the source line. The copies are the occurrences of a in the consumer line's arithmetic, including both the prose operands and the annotation in GSM8K's << >> format. We choose counterfactual values a' and a'' with the digit count of a, absent from the solution, such that the consumer's operation yields distinct integer results y_0, y', and y'' from a, a', and a''. Every edited text keeps the original token length. Writing a' into the source line leaves its displayed operands unchanged, so the edited source line is arithmetically inconsistent, whereas an edited copy shows no error because the consumer's result is cut. The activation edit keeps the displayed source and replaces the span from the source calculation to the end of its line, at the embedding output and every layer, with activations from the text-edited source line.

Cells. We cut each text just before the consumer's result and greedily continue for at most 256, 320, or 1536 new tokens (GSM8K few-shot, arithmetic few-shot, <think>). Writing settings as (source, copy), the cells are (a, a), (a, a'), (a', a), (a', a'), and (a', a'') with the source edited in the text; (a', a) and (a', a') are repeated with the source edited in its activations. The first four correspond to $(S, C) = (0, 0)$, $(0, 1)$, $(1, 0)$, and $(1, 1)$ in Sec. A2.6, where y' plays the role of y_1; a'' plays the role of the wrong copy b' in J2 (Sec. A2.5). In the (a', a'') cell, y'' follows the copy, y' follows the source, and y_0 is implied by neither edited occurrence. An edge is *isolated* if no other occurrence of a appears in the model-written text before the cut.

					(a', a)	(a, a')		(a', a')	(a', a'')			isolated	
Model	Task	Format	prob.	edges	y_0	y'	y_0	y_0	y''	y' [95% CI]	y_0	edges	y'
Gemma-2-2B	GSM8K	few-shot	716	1120	1.000	.973	.002	.000	.982	.000 [.000, .000]	.000	981	.000
Qwen3-8B	GSM8K	few-shot	497	860	.993	.924	.027	.001	.966	.008 [.002, .015]	.002	748	.009
Gemma-2-2B	arithmetic	few-shot	214	642	.998	.997	.000	.000	.997	.000 [.000, .000]	.000	538	.000
Qwen3-8B	arithmetic	few-shot	162	486	.973	.932	.016	.000	.959	.002 [.000, .006]	.000	404	.002
Qwen3-8B	GSM8K	<think>	47	117	1.000	.444	.479	.239	.615	.017 [.000, .044]	.316	9	.000
R1-7B	GSM8K	<think>	73	147	.986	.524	.340	.048	.721	.020 [.000, .046]	.163	19	.105
R1-32B	GSM8K	<think>	73	154	.987	.253	.688	.195	.500	.058 [.019, .106]	.357	16	.312
R1-70B	GSM8K	<think>	91	121	.942	.636	.281	.008	.868	.050 [.016, .092]	.025	97	.052
Qwen3-8B	arithmetic	<think>	38	114	1.000	.281	.474	.053	.623	.009 [.000, .026]	.281	3	.000
R1-7B	arithmetic	<think>	32	79	1.000	.899	.063	.000	.987	.000 [.000, .000]	.000	57	.000
R1-32B	arithmetic	<think>	47	91	1.000	.956	.033	.000	1.000	.000 [.000, .000]	.000	62	.000
R1-70B	arithmetic	<think>	79	227	1.000	.260	.674	.079	.793	.000 [.000, .000]	.176	14	.000

Table A16: **Consumer results under source–copy edits in model-written solutions.** Entries are fractions of edges whose regenerated consumer result equals y_0, y', or y'' in the indicated (source, copy) cell, with the source edited in the text. The unedited (a, a) cell yields y_0 on every edge in every setting, and editing the source in its activations changes the (a', a) and (a', a') columns by at most 0.009, as expected from the text-swap equivalence; this variant is an implementation check, not independent evidence. The prob. column counts problems that contribute edges, and the interval is a problem-clustered bootstrap. The isolated columns count isolated edges and give the (a', a'') source-implied rate among them.

Consumer results. A correct copy keeps the original result in at least 94% of edges in every setting (Table A16). In the (a, a') cell, few-shot runs follow the copy in 92–100% of edges, whereas <think> runs more often produce y_0; there y_0 is also implied by the source, so this cell cannot identify the occurrence used. The (a', a'') cell separates the occurrences. The source-implied result y' appears in at most 0.9% of its edges on arithmetic problems and few-shot GSM8K and in 1.7–5.8% of GSM8K <think> edges. On isolated GSM8K <think> edges, the rate is 0–31%, with the maximum from 5 of 16 R1-32B edges (problem-clustered 95% CI 11–57%); these subsets are small and not randomized. When <think> runs depart from the copy in the (a', a'') cell, they usually produce y_0; R1-70B on GSM8K is the exception, with departures split among y', y_0, and other values. Producing y_0 is not specific to conflict: with source and copy agreeing on a', Qwen3-8B and R1-32B

still produce y_0 on 24% and 19% of GSM8K <think> edges. In five of the eight <think> settings, at most 15% of edges are isolated, and continuations recompute the source calculation more often after the source edit (58–67% with only the source edited and 68–74% in the (a', a'') cell, versus 18–54% in the unedited cell). Thus y_0 can come from other occurrences of a or from re-derivation. Because the edited source is visibly wrong, source-implied rates may understate the source-following that a consistent edit would produce. Departures also depend on model and task: R1-32B follows the copy on every arithmetic edge but departs on half of its GSM8K edges.

Final answers. We score final answers on answer-separating edges, whose copy-implied and source-implied replayed answers differ. In few-shot formats, 84–90% of these edges end with the copy-implied answer, so the edited copy propagates through the rest of the solution (Table A17). In <think> formats, at most 4.6% do (6.2% under the alternative replay), 81–99% end with the original answer, and up to 18% reach no final answer within the continuation limit. The original answer also dominates when the regenerated consumer result followed the copy (77–99% of such edges). We do not test where the continuation abandons the edited value, so these results show only that <think> final answers need not follow an edited intermediate result.

Model	Task	Format	edges	copy-implied	original	none
Gemma-2-2B	GSM8K	few-shot	995	.859	.005	.001
Qwen3-8B	GSM8K	few-shot	773	.836	.038	.005
Gemma-2-2B	arithmetic	few-shot	623	.896	.000	.000
Qwen3-8B	arithmetic	few-shot	474	.857	.008	.000
Qwen3-8B	GSM8K	<think>	45	.000	.889	.089
R1-7B	GSM8K	<think>	62	.000	.919	.081
R1-32B	GSM8K	<think>	77	.000	.922	.065
R1-70B	GSM8K	<think>	90	.000	.989	.000
Qwen3-8B	arithmetic	<think>	66	.000	.864	.106
R1-7B	arithmetic	<think>	65	.046	.831	.015
R1-32B	arithmetic	<think>	39	.000	.974	.000
R1-70B	arithmetic	<think>	152	.000	.809	.184

Table A17: **Final answers in the (a', a'') cell.** Edges are those whose copy-implied and source-implied replayed answers differ. Copy-implied answers replay the original solution's later calculations with the consumer's result set to y''; later calculations that reuse a directly take a', and replaying them with a instead lowers the few-shot rates by at most 0.03. Original is the model's own unedited final answer, and none marks continuations without a final answer; the remaining edges end with other answers.

Scope. Edges are selected: both calculations must parse, be correct as written, and share an explicit copied operand. The source edit is visible while the copy edit is not, other occurrences of a remain unedited, and the arithmetic solutions follow a format request. The results concern local arithmetic dependencies in these solutions, not unrestricted reasoning.

A2.8 Regenerating the Local Copy

The G3 experiment supplies a problem block containing all step definitions and regenerates the consumer line after the source intervention. The prompt contains an instruction, a one-shot example, the problem block with each step's expression but not its value (for example, `x3 = 9 * 4` and `x5 = x3 - x1`), and the solution up to the consumer line. The problem block preserves the target computation when the original consumer line is removed. Without it, six attempted configurations across 2B–27B models reach only 0.02–0.24 baseline accuracy. G3 therefore tests a constrained form of self-generation, not unrestricted CoT.

Table A18 keeps all 741 saved samples in each arm. The target-copy parser extracts the operand corresponding to the intervened source. Failure to extract it can reflect an alternative line format rather than a malformed generation. The single-edge agreement metric substitutes the extracted operand into the original operation while retaining the *ground-truth value of the other operand*. It compares the resulting value with the generated result. A missing copy or result counts as disagreement. Because

the other generated operand is not used by this metric, agreement does not certify the arithmetic of the full generated line.

Renamed sources and other copies. G3 uses the digit-resampling generator, so the patched source line usually defines a different variable (675 of 741 patched samples); in the example above, `x3 = 9 * 4 = 36` becomes `x1 = 3 * 2 = 48`. The consumer's operand is then undefined in the patched solution, although the problem block still names it. Among the 358 patched samples with another extracted target copy, 264 copy the first operand of the source's own definition (191 from a leaf definition in the problem block and 73 from an earlier line), 40 write a line for another variable, 22 copy a donor-line operand, and the remaining 32 copy other values from the trace. Among the 196 samples without an extracted target copy, 48 skip to the answer and 148 write lines with values but no variable references. The 66 patched samples whose line keeps its name yield 72.7% [62, 84] counterfactual copies and 98.5% counterfactual results. The name-preserving replication in Sec. A1.3 confirms that the low propagation reflects the lost binding: with the binding intact, the counterfactual copy appears in 81–94% of patched samples across three models.

Outcome	Source clean	Source patched
Original target copy	0.997 [0.993, 1.000]	0.018 [0.009, 0.027]
Counterfactual target copy	0.000 [0.000, 0.000]	0.235 [0.207, 0.262]
Other extracted target copy	0.001 [0.000, 0.004]	0.483 [0.448, 0.519]
No extracted target copy	0.001 [0.000, 0.004]	0.265 [0.235, 0.292]
Original result	0.993 [0.987, 0.999]	0.059 [0.042, 0.079]
Counterfactual result	0.000 [0.000, 0.000]	0.271 [0.242, 0.300]
Other result	0.007 [0.001, 0.013]	0.669 [0.636, 0.703]
Single-edge copy-implied agreement	0.993 [0.987, 0.999]	0.537 [0.503, 0.572]

Table A18: **All-sample outcomes when Gemma-2 writes the consumer line.** Proportions and trace-cluster bootstrap 95% intervals. Each arm contains 741 samples; the four copy categories and three result categories separately sum to one before rounding. Patched samples include 174 intended counterfactual copies, 196 without an extracted target copy, and 201 counterfactual results. Single-edge copy-implied agreement holds for 398 samples and uses the ground-truth value of the other operand, as defined above. These rates show partial propagation and should not be equated with the supplied-copy results in Table 1.

A2.9 A Fixed Restatement Rule Across Domains

We test whether a single restatement rule predicts large source effects without per-domain tuning. Table A19 reports the balanced accuracy of that one fixed rule in each transfer domain; no per-domain threshold is fitted. For GSM8K, the feature is later reappearance anywhere in the trace, and the target is the final parsed result (Table A41). Performance varies across conditions. Under Qwen3's variable-form prompt, the rule detects all 1298 positive edges but only 12 of 42 negative edges, giving balanced accuracy 0.64. The GSM8K subsets contain only 8, 5, and 6 positive cases for Gemma-2, OLMo-2, and Qwen3, respectively, limiting the precision of their comparisons.

	Gemma-2		OLMo-2		Qwen3	
Domain	BA	n_+/n_-	BA	n_+/n_-	BA	n_+/n_-
Arithmetic self-generated	0.97	127/1801	0.99	320/1471	0.86	1213/127
Arithmetic variable-form prompt	0.99	276/1578	0.99	108/1533	0.64	1298/42
k-hop	0.79	145/519	0.75	70/371	0.82	121/320
GSM8K	0.67	8/107	0.87	5/80	0.79	6/173

Table A19: **Performance of a fixed restatement rule.** Balanced accuracy (BA) of the pre-specified rule (non-restated $\Rightarrow$ signed source-patch effect ≥ 0.25 nats), applied without per-domain tuning. n_+/n_- gives the positive and negative class counts; their sum is the total sample size. The arithmetic prompt condition uses the variable-form instruction and the model's realized restatement. For GSM8K, non-restated means that the source value does not reappear later in the trace; the measured effect is on the final parsed result. The same samples and label definitions are used for the baseline comparisons in Table A9.

A3 Generality Across Formats, Value Types, Scale, and Architecture

A3.1 Local-Copy Substitution in Native Thinking Formats

Our controlled templates impose a trace format, whereas reasoning models provide their own native format. We therefore render the same traces inside Qwen3-8B's native chat template, placing solution lines in its `<think>` block. The teacher-forced, all-layer source patch gives a mean absolute effect of 0.005 nats on restated edges (median 0.0007) and 2.73 nats on non-restated edges (n=800 per condition). Qwen3's observational restated-edge effect of 0.86 nats under one-shot generation does not recur in this controlled native rendering. The result also holds in larger reasoning models. DeepSeek-R1-Distill-Qwen-32B gives 0.004 nats with restatement and 1.97 without it (n=600 per condition); DeepSeek-R1-Distill-Llama-70B gives 0.010 and 4.99 nats, respectively (n=400). Thus source sensitivity without a local copy remains large at scale, while the relative suppression from a correct copy remains strong.

A3.2 Value-Type Generality: Words, Names, and PrOntoQA Deduction

We repeat the copy-configuration comparison of Fig. 2 in three non-numeric domains. All experiments use the same source-line activation patch and score direct edges; sample sizes are reported with each configuration.

Words. Each step selects which of two operands belongs to a named semantic category. We use twelve categories and words that are single tokens in every model's tokenizer. Because the selected word equals one operand, changing the source leaves the consumer's correct value unchanged on half of the edges. The main comparison uses the value-changing half; Table A20 includes the unchanged half as a control. Source patching can change a value's likelihood even when the correct consumer value is unchanged. We retain these edges as a control for confidence changes, rather than treating every residual effect as propagation of a different value.

Names. The prompt supplies a closed-world fact list, and traces compose two interleaved chains of unary relations such as boss, teacher, mentor, coach, landlord, and doctor. For each chain, a parallel counterfactual chain specifies the results expected after a source change. We also track the premise span that produced each value, allowing premise and derived-source patches to be compared.

PrOntoQA. We use the benchmark of Saparov & He (2023). Its official fictional-ontology generator supplies 3–5-hop problems, with 300 problems per condition. The gold chain alternates derived facts with rule restatements. The *rule-restated* condition preserves this format, *conclusion-only* drops the rules, and *rule-consequent-masked* keeps each rule but replaces its consequent with length-matched

masking. Because the first derived fact appears verbatim in the context, we report only internal edges whose sources are derived rather than given.

Counterfactual words, names, and concepts occur nowhere else in the trace and preserve token count; we redraw invalid candidates up to eight times. Table A20 reports every configuration with trace-clustered CIs, Table A21 reports PrOntoQA internal-edge effects, and Table A22 reports the generation-time source×copy cross, including expanded GSM8K runs.

Two additional results qualify the main pattern. For names, Qwen3 retains $\mathrm{CDE}_0 = 0.22$ with a correct copy fixed and $\Delta_S = 0.24$ with a counterfactual copy fixed, showing that both conditions can retain source effects in this domain. For PrOntoQA, a rewritten rule whose subject conflicts with the derived concept is followed infrequently when the source is clean but more often when the source is also patched: 4% versus 72% in Qwen3, and 30% versus 51% in Gemma-2. The result implied by the rewritten rule is therefore more likely when the source agrees with that rule.

GSM8K generation protocol. The G6 experiment first generates solutions, then selects parsed binary arithmetic steps whose original result agrees with their displayed operation and whose operands explicitly repeat a source value. It supplies the consumer prefix and greedily regenerates at most eight tokens from the first result position. The designated counterfactual candidate is computed by replacing the target operand in the original parsed operation; the other operand stays fixed when this candidate is computed. The actual prefix edit replaces every matching occurrence of the source value. If both operands share that value, the single-operand candidate can differ from the result of the fully edited expression. Saved outcome records do not retain the operand pairs, so we cannot quantify how often this occurs from these records. The reported frequencies measure agreement with the designated candidate, not necessarily correctness of the entire edited expression. Separately, source donors resample the numerals in the source line without enforcing arithmetic consistency or agreement with the edited copy. Thus, unlike the template and native-format generation crosses, $S = C = 1$ need not align the source and copy. The original-result frequency of 1.000 in the clean GSM8K cells concerns these selected local steps, not the accuracy of full solutions.

A3.3 Scale Series: Six Reasoning Models

Table A23 reports native-format source-patch effects and the generation cross for six R1-Distill models. At sizes up to 14B, we use n=1200 edges for the patch analysis and 1166 samples per generation cell. The 32B model uses 600 edges for the patch analysis and 583 samples per generation cell; the 70B model uses 400 and 391, respectively. Δ_S remains small from 1.5B through 8B, increases at 14B and 32B, and reaches 0.146 at 70B (Spearman $\rho = 0.83$ across six sizes). The magnitude of CDE_0, with the correct copy fixed, remains at most 0.018 throughout. This series combines Qwen- and Llama-based models, so size and base architecture are not separately controlled. The 1.5B model assigns a mean clean gold log-probability of -1.1 to values on non-restated edges and has the smallest correct-copy contrast in the series (0.21 versus 1.46).

A3.4 Non-Transformer Architectures

Because the instrument intervenes on each block's residual output, we can also apply it to a state-space model (Mamba2-2.7B, `AntonV/mamba2-2.7b-hf`) and a recurrent neural network (RNN; RWKV-4-3B, Pile-trained). In both architectures, a correct local copy reduces the mean signed source-patch effect by a factor of 13 (Table A24). Signed means can hide cancellation, and this run saves only summary quantiles. For Mamba2, the layer-0 patch of the depth sweep in Table A25 applies the same intervention to 385 edges from 40 traces and saves every edge: the mean absolute effect is 1.05 nats $[0.91, 1.24]$ without a copy and 0.043 $[0.034, 0.055]$ with a correct local copy, with 41% versus 5% of edges negative. For RWKV, no per-edge values were saved; the local-copy interquartile range $[-0.030, 0.127]$ shows two-sided effects, and a name-preserving rerun that saves every edge shows that its signed reduction overstates suppression (Sec. A1.3). The filler control is unavailable for RWKV in this run because its tokenizer splits the filler characters into variable-length spans. In Mamba2, source patches beginning between layers 0 and 19 produce signed effects of 0.55–0.61 nats, whereas patches beginning at layer 29 or later produce effects between -0.08 and 0.01 nats (Table A25). This brackets the relevant depth transition between layers 19 and 29 of 64.

Domain	Model	Configuration	n	mean Δ [95% CI]	mean $\lvert\Delta\rvert$	tf. acc.
words	Gemma-2-2B	no copy	750	5.510 [5.327, 5.685]	5.517	0.66
		correct local copy	750	-0.102 [-0.142, -0.062]	0.316	0.68
		matched-length filler	750	4.703 [4.549, 4.872]	4.741	0.63
		wrong local copy	750	3.278 [3.124, 3.420]	3.300	0.38
		remote duplicate	750	0.997 [0.900, 1.104]	1.374	0.62
		bare word	750	1.644 [1.535, 1.750]	1.833	0.65
		after all consumers	750	5.500 [5.323, 5.684]	5.508	0.66
		value-unchanged edges (no copy)	750	0.213 [0.170, 0.262]	0.367	
		value-unchanged edges (correct local copy)	750	0.305 [0.255, 0.356]	0.436	
words	Qwen3-8B	no copy	741	11.466 [11.133, 11.789]	11.469	0.84
		correct local copy	737	0.150 [0.105, 0.193]	0.221	0.89
		matched-length filler	741	10.562 [10.199, 10.901]	10.563	0.85
		wrong local copy	734	7.331 [7.024, 7.619]	7.334	0.50
		remote duplicate	741	1.213 [1.073, 1.350]	1.438	0.80
		bare word	741	3.412 [3.177, 3.641]	3.459	0.82
		after all consumers	741	11.456 [11.121, 11.807]	11.459	0.83
		value-unchanged edges (no copy)	737	0.020 [-0.031, 0.074]	0.250	
		value-unchanged edges (correct local copy)	735	0.031 [0.001, 0.058]	0.096	
words	OLMo-2-7B	no copy	741	9.883 [9.558, 10.194]	9.934	0.77
		correct local copy	737	-0.354 [-0.424, -0.288]	0.514	0.77
		matched-length filler	741	9.000 [8.645, 9.352]	9.031	0.79
		wrong local copy	734	4.699 [4.378, 5.033]	4.942	0.45
		remote duplicate	741	1.468 [1.285, 1.648]	2.144	0.77
		bare word	741	2.274 [2.082, 2.489]	2.684	0.76
		after all consumers	741	9.891 [9.591, 10.216]	9.939	0.77
		value-unchanged edges (no copy)	737	0.627 [0.498, 0.757]	0.879	
		value-unchanged edges (correct local copy)	735	0.350 [0.269, 0.431]	0.682	
names	Gemma-2-2B	no copy	750	4.003 [3.777, 4.236]	4.037	0.57
		correct local copy	750	0.075 [-0.010, 0.156]	0.593	0.60
		matched-length filler	750	2.514 [2.345, 2.675]	2.560	0.49
		wrong local copy	750	1.790 [1.661, 1.921]	1.885	0.43
		remote duplicate	750	1.771 [1.634, 1.906]	1.836	0.56
		bare word	750	1.939 [1.805, 2.085]	1.995	0.54
		after all consumers	750	4.002 [3.772, 4.237]	4.035	0.57
		premise patch (no copy): self / consumer		9.36 / 0.253 [0.090, 0.420]		
		premise patch (correct local copy): self / consumer		9.78 / 0.230 [0.067, 0.390]		
names	Qwen3-8B	no copy	750	14.374 [13.937, 14.804]	14.434	0.69
		correct local copy	750	0.937 [0.807, 1.083]	1.006	0.71
		matched-length filler	750	9.778 [9.453, 10.100]	9.834	0.66
		wrong local copy	750	8.653 [8.287, 9.009]	8.676	0.65
		remote duplicate	750	3.622 [3.337, 3.927]	3.715	0.70
		bare word	750	7.272 [6.886, 7.655]	7.328	0.70
		after all consumers	750	14.390 [13.941, 14.807]	14.444	0.70
		premise patch (no copy): self / consumer		22.99 / 0.129 [-0.100, 0.339]		
		premise patch (correct local copy): self / consumer		25.34 / 0.161 [0.042, 0.271]		
names	OLMo-2-7B	no copy	750	9.404 [8.869, 9.959]	9.585	0.57
		correct local copy	750	0.395 [0.199, 0.580]	1.107	0.64
		matched-length filler	750	7.331 [6.860, 7.800]	7.478	0.54
		wrong local copy	750	4.451 [4.161, 4.775]	4.742	0.45
		remote duplicate	750	3.578 [3.251, 3.918]	3.868	0.59
		bare word	750	3.903 [3.557, 4.241]	4.155	0.57
		after all consumers	750	9.418 [8.897, 9.961]	9.597	0.57
		premise patch (no copy): self / consumer		17.44 / -0.480 [-0.780, -0.204]		
		premise patch (correct local copy): self / consumer		19.40 / -0.335 [-0.582, -0.092]		
PrOntoQA	Gemma-2-2B	no copy	708	0.533 [0.441, 0.622]	1.157	0.67
		correct local copy	706	0.583 [0.518, 0.647]	0.642	0.86
		rule consequent masked	707	0.134 [0.080, 0.189]	0.637	0.68
PrOntoQA	Qwen3-8B	no copy	705	1.720 [1.597, 1.842]	1.798	0.79
		correct local copy	696	0.084 [0.051, 0.121]	0.107	0.96
		rule consequent masked	699	1.517 [1.371, 1.661]	1.561	0.85
PrOntoQA	OLMo-2-7B	no copy	705	1.215 [1.050, 1.378]	1.750	0.75
		correct local copy	696	0.226 [0.166, 0.290]	0.283	0.94
		rule consequent masked	699	0.509 [0.381, 0.635]	1.252	0.77

Table A20: **Copy configurations for words, names, and PrOntoQA.** Mean signed source effect with trace-clustered 95% CI, mean absolute effect, and teacher-forced value accuracy (tf. acc.) per configuration; effects are in nats. Premise-patch rows for names report effects on the patched step's own value / on its consumers. PrOntoQA rows: internal edges only.

Model	conclusion-only	rule restated	rule consequent masked	tf. acc.
Gemma-2-2B	0.53	0.58	0.13	0.67 / 0.86
Qwen3-8B	1.72	0.08	1.52	0.79 / 0.96
OLMo-2-7B	1.22	0.23	0.51	0.75 / 0.94

Table A21: **PrOntoQA: mean signed source effect on internal edges (nats).** Rule restatement reduces the effect in Qwen3 and OLMo-2; masking the rule's consequent weakens this reduction in both models. Gemma-2 does not show the same reduction and has lower value accuracy in the conclusion-only format. The last column reports teacher-forced value accuracy (tf. acc.) for conclusion-only / rule-restated traces.

		counterfactual result					
Domain	Model	S clean	S patched	Δ_S	original result	CDE_0	n/cell
words	Gemma-2-2B	0.873	0.797	-0.077	0.830	-0.030	600
words	Qwen3-8B	0.880	0.967	0.087	0.978	0.063	583
names	Gemma-2-2B	0.720	0.777	0.057	0.787	0.068	600
names	Qwen3-8B	0.727	0.970	0.243	0.982	0.218	600
PrOntoQA	Gemma-2-2B	0.296	0.506	0.210	0.751	0.397	257
PrOntoQA	Qwen3-8B	0.041	0.722	0.681	0.989	0.426	270
GSM8K steps	Gemma-2-2B	0.877	0.878	0.001	1.000	0.003	896
GSM8K steps	Qwen3-8B	0.756	0.814	0.059	1.000	0.003	700

Table A22: **Generation-time source×copy interventions across domains.** The first two outcome columns report counterfactual-result frequency with a counterfactual copy fixed ($C = 1$) and the source clean or patched; Δ_S is their difference. Original-result frequency is measured with both inputs clean ($S = C = 0$), and CDE_0 is its decrease after source patching with a correct copy fixed (Eq. 6). For GSM8K, the counterfactual result is a designated single-operand candidate, and source corruption is independent of the copy edit (Sec. A3.2). GSM8K rows pool the test split and training-split extensions: $6{,}469$ problems for Gemma-2 and $2{,}569$ for Qwen3, yielding 0.14 and 0.27 usable edges per problem.

	mean $\lvert\Delta\rvert$ (nats)		counterfactual result			
Model	restated	non-restated	S clean	S patched	Δ_S [95% CI]	CDE_0
R1-Distill-Qwen-1.5B	0.206	1.46	0.754	0.791	0.037 [0.020, 0.054]	0.018
R1-Distill-Qwen-7B	0.030	2.14	0.875	0.896	0.021 [0.012, 0.032]	0.009
R1-Distill-Llama-8B	0.014	5.91	0.896	0.921	0.025 [0.014, 0.037]	0.000
R1-Distill-Qwen-14B	0.010	2.28	0.852	0.906	0.054 [0.043, 0.066]	0.003
R1-Distill-Qwen-32B	0.004	1.97	0.863	0.919	0.057 [0.021, 0.092]	0.000
R1-Distill-Llama-70B	0.010	4.99	0.785	0.931	0.146 [0.098, 0.194]	0.000

Table A23: **Six-model size comparison.** Columns report mean absolute source-patch effects (nats) on restated and non-restated edges in native `<think>` traces, counterfactual-result frequencies with $C = 1$, their source-induced difference Δ_S with a nominal 95% CI, and CDE_0 with $C = 0$. Generation intervals are trace-clustered through 14B; the 32B and 70B summary-only generation runs use normal approximations that do not account for within-trace clustering. The source-effect runs retain trace records at every model size and support the clustered intervals in Fig. 3(b).

Model	no copy	local copy	matched filler	n/style
Mamba2-2.7B	0.449	0.034	0.436	964
RWKV-4-3B	0.146	0.011	—	962

Table A24: **Local-copy substitution in a state-space model and an RNN.** Mean signed source-patch effect on direct edges (nats).

first layer / total	0/64	10/64	19/64	29/64	35/64	40/64	45/64	51/64	59/64
no copy (signed)	0.55	0.61	0.58	-0.07	-0.08	-0.07	-0.07	-0.03	0.01
correct local copy (signed)	0.03	0.05	0.08	0.02	0.02	0.02	0.02	0.00	-0.00

Table A25: **Mamba2-2.7B source-effect depth profile.** Mean signed source-patch effect (nats) when the source step is patched from the specified layer through the final layer.

A3.5 RLVR AND SCALE EXTENSIONS

We also test an RLVR-distilled reasoning model and compare Gemma-2-2B with Gemma-2-27B. Table A26 reports both.

Model / condition	restated	non-rest.	AUROC
R1-Distill-7B, template	0.021	1.944	—
R1-Distill-7B, self-gen (rate .69)	0.028	2.524	0.765
Gemma-2-27B, template	0.003	2.728	—
Gemma-2-27B, self-gen (rate .00)	—	3.002	0.989

Table A26: Mean signed source effects (nats) in an RLVR-distilled reasoning model and across a $13\times$ within-family size increase. The self-generated rows give the fraction of direct edges with restated values and the distance-stratified AUROC for direct versus no-path pairs. Template experiments also include a value-free filler control (R1-Distill 1.75; Gemma-2-27B 3.11 nats). Gemma-2-27B follows the one-shot example's variable form exactly, so its restated cell comes from the template contrast.

A4 WHERE THE EFFECT LOCALIZES

Depth and head interventions provide partial localization of the effect. Varying the first patched layer places the source-effect transition at 56–69% of network depth in the three transformers and between layers 19 and 29 of 64 in Mamba2 (Sec. A4.1, Sec. A3.4). In OLMo-2, patching counterfactual outputs of eight selected heads reproduces 99% of the all-layer source effect, versus 4% for matched controls (Sec. A4.4). When a correct copy is present, a separate ablation of candidate switching heads recovers only 8–9% of the no-copy source effect. Residual-stream steering also changes source sensitivity, but random-direction controls produce substantial changes in Qwen3 (Sec. A4.3, Sec. A4.5). These interventions identify contributing components without establishing a complete circuit.

A4.1 COARSE DEPTH PROFILES OF THE SOURCE EFFECT

We vary the first patched layer a and patch the source span from that layer through the final layer. Comparing the resulting effects shows where source sensitivity changes most sharply as early layers are excluded from the intervention.

Corruption start a:	0	mid_1	mid_2	late
Gemma-2 (0/8/14/20)	1.00	0.945	0.885	-0.02
OLMo-2 (0/8/16/24)	1.00	0.99	0.92	-0.03
Qwen3 (0/9/18/27)	1.00	0.982	0.952	-0.01

Table A27: Remaining share of the all-layer source-patch effect when patching source activations from layer a through the final layer. Excluding early layers changes the effect little, whereas beginning the patch at the late settings removes most of it. The denser sweeps in Fig. A4 locate the transition more closely. AtP* attribution mass over the top 256 components (Kramár et al., 2024) peaks earlier than the source-sensitivity collapse in Gemma-2 (L10–13 versus L16–18) and OLMo-2 (L9–13 versus L18–21). The ranges overlap in Qwen3 (mass L16–23; collapse L21–24), so the separation is common but not universal.

A4.2 Depth Profiles Across Copy Conditions

Layer-resolved profiles locate where source sensitivity changes across four copy conditions. We patch the source span's activations from layer k through the final layer and vary k in Gemma-2-2B (n=400 direct edges per condition and depth). With a correct local copy, the mean absolute source effect remains at most 0.02 nats at every depth. Without a copy, the profile retains 86% of its all-layer effect through 54% relative depth and falls sharply between 62% and 69%.

The remote-duplicate and wrong-duplicate profiles decline in the same region. The mean absolute effect with a remote duplicate falls from 0.83 to 0.46 nats at 62% depth and to 0.24 at 69%; with a wrong duplicate, it falls from 0.55 to 0.38 and then 0.23 nats. Their mean signed effects approach zero slightly earlier (62%) than the no-copy effect (69%). Thus, the remaining source sensitivity under these copy conditions passes through the same depth range rather than moving to a separate one.

The profiles decline in a similar depth range across models, with a more gradual tail in OLMo-2. In Qwen3-8B, the no-copy, remote-duplicate, and wrong-duplicate effects fall between 61% and 69% relative depth ($2.92 \to 0.25$, $1.07 \to 0.20$, and $1.12 \to 0.30$), while the correct-copy effect stays at most 0.03. In OLMo-2-7B, the corresponding effects decline between 56% and 62% ($5.20 \to 1.72$, $2.42 \to 1.09$, and $2.39 \to 1.42$) but retain substantial tails, while the correct-copy effect remains near zero.

A4.3 Partial Recovery of Source Sensitivity by Head Ablation

We ablate the candidate heads identified in Sec. A4.4 to test whether they contribute to the loss of source sensitivity when a local copy is present. For each edge, we compare three head conditions: intact heads, the three switching heads zeroed, and matched same-layer random triples zeroed (n=600 edges per cell). The clean and patched forward passes use the same head condition.

Zeroing the switching heads raises the mean absolute source effect on restated edges from 0.005 to 0.121 nats in Gemma-2 ($24\times$) and from 0.001 to 0.507 in OLMo-2 ($500\times$). Both control triples remain near floor (≤ 0.011). A rerun that saves trace identifiers gives non-overlapping trace-clustered bootstrap 95% CIs: 0.121 $[.089, .157]$ for the Gemma-2 ablation versus $[.004, .017]$ for its controls, and 0.507 $[.380, .648]$ for the OLMo-2 ablation versus $[.0001, .001]$ for its controls.

The recovery is nevertheless only 8–9% of the no-copy effect (9.0% in Gemma-2 and 7.5% in OLMo-2). The intervention therefore identifies contributing heads, not a complete implementation of the route change. Redundancy is also consistent with the modest reduction in the variable-form effect under the same ablation (17% in Gemma-2 and 32% in OLMo-2). The ablation does not restore source sensitivity under remote or wrong duplicates, suggesting that other components mediate those effects.

After full variable-form SFT, the ablation experiment still shows partial recovery. In this evaluation, the no-copy mean absolute source effect rises from 1.35 to 2.67 nats. The separate checkpoint evaluation reports a source effect of 4.64 nats (Table A33). With the heads intact, the restated-edge mean absolute effect remains 0.009; zeroing the same heads raises it to 0.089, compared with 0.012 and 0.047 for the controls. The remote-duplicate effect again shows no corresponding recovery.

A4.4 Retrieval Heads

Table A28 ranks heads by attention to the source value in variable-form traces. It also reports their attention to that value and to the local copy in restated traces. The comparison distinguishes Gemma-2, whose leading heads redirect attention from the source to the local copy, from Qwen3, whose top-ranked sets for the two occurrences are disjoint.

Causal follow-up. We intervene directly on the top heads using traces disjoint from the discovery set (Gemma-2: 300 traces; Qwen3: 150 traces). We define ablation cost as clean minus ablated gold log-probability, so a positive value means that ablation hurts prediction.

In Gemma-2, patching counterfactual outputs of only the three heads downstream of the source line reproduces a median 16% of the all-layer source-patch effect on strong edges (n=2111, $|\Delta| > 0.5$).

Matched same-layer control triples reproduce 3%. Yet zero-ablation does not degrade non-restated consumer prediction: the mean ablation cost is -0.12 nats, compared with $+0.08$ for controls. This combination is consistent with redundant routing and attention self-repair (McGrath et al., 2023).

In Qwen3, the candidate remote-reading heads L23H4, L23H5, and L23H31 are more important for prediction. Their zero-ablation cost is 0.63 nats on variable-form traces, versus 0.01 for controls; the trace-clustered CI for this difference is $[0.51, 0.73]$. On restated traces, the median cost is zero but the mean is 0.26, reflecting a heavy-tailed subset. This residual engagement is consistent with Qwen3's nonzero observational effect on restated edges in Table A7. Patching counterfactual head outputs reproduces 36% of the total source-patch effect in variable-form traces, concentrated in a subset of edges.

OLMo-2 resembles Gemma-2: its top head redirects attention to the local copy (0.90), its median ablation cost is 0.003 nats, and the share of the source effect reproduced by head patching exceeds the matched-control level by a median of 18 percentage points. Among the top 12 heads ranked separately by attention to the remote source and to the local copied-operand segment, the sets share six heads in Gemma-2, two in OLMo-2, and none in Qwen3.

Reproducing effects with selected head outputs. A separate experiment ranks heads by attention to the source or local copy and patches their outputs from counterfactual runs into clean runs. For each of Gemma-2 and OLMo-2, we use 40 traces for ranking and 60 held-out traces for evaluation. The source intervention is scored as the decrease in the original value's log-probability, relative to the full source-span patch. In OLMo-2, eight heads reproduce 99% of the full 2.92-nat effect, compared with 4% for matched controls. Gemma-2's smaller 0.26-nat baseline makes the corresponding ranking less stable.

For the local-copy intervention, we instead score the gain in the copy-implied counterfactual result's first-token log-probability, relative to editing the copy text itself. Two heads reproduce 83% of this gain in OLMo-2 and four reproduce 93%; Gemma-2 reproduces 86% with eight heads and 98% with twelve. The evidence therefore supports model-dependent concentration and redundancy, not a universal head circuit. Full path-level necessity and sufficiency tests in the style of Wang et al. (2023) remain open.

Model	head	source (no copy)	source (copy)	local copy
Gemma-2	L18H6	0.803	0.011	0.890
	L17H3	0.707	0.035	0.454
	L16H6	0.686	0.004	0.841
Qwen3	L23H4	0.912	0.003	0.030
	L23H5	0.888	0.001	0.036
	L23H31	0.777	0.006	0.045
OLMo-2	L19H22	0.728	0.031	0.902
	L16H0	0.664	0.101	0.372
	L18H19	0.564	0.047	0.256

Table A28: Top retrieval heads ranked by source-value attention in variable-form traces. Columns give mean attention to the remote source in variable-form and restated traces, then to the local copy in restated traces; L and H identify the layer and head. The Gemma-2 heads redirect attention to the local copy; Qwen3's top-ranked source and local-copy head sets are disjoint.

A4.5 A Residual Direction Associated with Copy Presence

For each model, we render 60 estimation problems in both formats and collect residual states at the value-predicting position in the bottleneck layers (Gemma-2 L16–18, OLMo-2 L18–20, and Qwen3 L21–23). The value and its two preceding tokens are identical across each format pair. Averaging the restated-minus-non-restated difference over 300 pairs defines a direction d_ℓ at each layer.

On 100 held-out problems, we add or subtract αd_ℓ at the same position in both clean and patched passes, then measure the mean absolute source effect. Controls use a norm-matched random direction at each layer or estimate and apply d at early layers (L3–5). Table A29 reports subtraction from restated traces.

At $\alpha = 1$, subtraction raises this effect from 0.004 to 0.150 nats in Gemma-2, from 0.000 to 0.477 in OLMo-2, and from 0.022 to 0.411 in Qwen3. The matched controls are smaller, although Qwen3's random direction reaches 0.127. At larger strengths, random directions also perturb Qwen3 and OLMo-2, and accuracy declines. Adding d to non-restated traces does not suppress source sensitivity and instead lowers the clean log-probability (from -2.8 to -5.9 in Gemma-2 at $\alpha = 1$). The intervention therefore identifies a copy-associated residual correlate that is neither a sufficient copy representation nor a uniquely selective control direction.

A5 Controlled Training Experiments

A5.1 Controlled Training: Validity Criteria and Controls

This appendix defines when we report the source-suppression index G for a from-scratch run and describes controls that distinguish a learned effect from an intervention too weak to measure. The models have six decoder layers, hidden width $d = 256$, and 5.4M parameters. They are pre-LayerNorm transformers with eight attention heads, GELU MLPs of width $4d$, and learned absolute position embeddings for 512 positions. A word-level vocabulary contains variable names, the integers 0–999, operators, and formatting tokens. Training uses AdamW with learning rate 10^{-3}, weight decay 0.01 on non-embedding weight matrices, a one-cycle schedule with 5% warm-up, gradient clipping at 1.0, and batches of 32 traces of depth 3 or 4 drawn from the arithmetic generator. Evaluation uses 60 held-out traces rendered at $p_{\text{eval}} = 0.5$.

Reporting criteria. Screening only on the restated source effect would presuppose suppression. We instead apply the following criteria to each saved run:

1. Leaf value-token accuracy is at least 0.85 and leaf negative log-likelihood is at most 0.50.
2. Each evaluation format contains at least 100 direct edges. The variable-form mean absolute source effect is at least 1 nat, its AUROC against no-path pairs is at least 0.80, and its absolute signed mean is at least half its mean absolute effect.
3. At least one probe in the restated condition—remote source, local copy segment, or local reference segment—has a mean absolute effect at least $\max(3f_r, 0.5)$ nats, where f_r is the restated condition's *remote-source* no-path floor. This is a shared floor for the reporting check; the individual probe floors are also reported as controls.

The criteria establish learned leaf predictions and a measurable intervention signal. They do not require suppression or high consumer accuracy in both formats. An unstable run at learning rate 3×10^{-3} fails them at step $20{,}000$ despite a noise-corrected index of 0.65; its leaf accuracy is 0.84 and its variable-form AUROC 0.73. Of the 43 saved from-scratch runs, only the two runs at that learning rate fail the criteria, and they appear only in the optimization probe below. Passing runs not shown in a table are a third seed at $p_{\text{wrong}} = 1.0$, whose $G = -1.33$ is close to the reported seeds, and duplicate evaluations of the filler runs. We report consumer-format accuracy separately to avoid treating sensitivity as competence.

Controls. Every source effect is reported against a no-path floor drawn from the same population as the direct edges. We use only internal consumer lines because leaf consumers are insensitive by construction and would bias the floor downward. The local copy probe replaces the line's copied-operand segment; the local reference probe replaces its variable-reference segment. The copy probe is accompanied by its own no-path floor and by the matched reference-segment intervention. We do not divide the copy effect by the remote effect: the copy patch sits one token from its target, whereas the remote patch sits roughly thirty tokens away. We report the two effects separately with their distances from the prediction position stated. The implied-result readout avoids this asymmetry by comparing two hypotheses about the same token in one forward pass. Results agree when computed over all corrupted copies or only the digit-count-matched subset.

Length-matched training control. Trace length and token predictability both change with p, so an apparent restatement effect could instead follow from longer or more predictable lines. We separate these factors with a corpus in which the restated segment is present but value-free. Each operand

Model	condition	mean $\lvert\Delta\rvert$ [95% CI]	clean log-prob	acc.	n
Gemma-2-2B (L16–18)	base	0.004 [0.003, 0.006]	-0.00	1.00	1000
	$-\alpha d$, α=0.25	0.012 [0.010, 0.015]	-0.00	1.00	600
	random dir., α=0.25	0.004 [0.003, 0.006]	-0.00	1.00	600
	early layers, α=0.25	0.004 [0.003, 0.006]	-0.00	1.00	600
	$-\alpha d$, α=0.5	0.044 [0.035, 0.056]	-0.01	1.00	600
	random dir., α=0.5	0.005 [0.003, 0.006]	-0.00	1.00	600
	early layers, α=0.5	0.004 [0.003, 0.005]	-0.00	1.00	600
	$-\alpha d$, α=1	0.150 [0.139, 0.161]	-0.10	0.98	1000
	random dir., α=1	0.010 [0.006, 0.014]	-0.00	1.00	1000
	early layers, α=1	0.007 [0.005, 0.010]	-0.00	1.00	1000
	$-\alpha d$, α=2	0.218 [0.206, 0.230]	-0.43	0.83	1000
	random dir., α=2	0.083 [0.070, 0.099]	-0.11	0.97	1000
	early layers, α=2	0.062 [0.049, 0.077]	-0.02	0.99	1000
	$-\alpha d$, α=4	0.156 [0.145, 0.169]	-1.11	0.61	1000
	random dir., α=4	0.416 [0.381, 0.452]	-2.26	0.56	1000
	early layers, α=4	0.088 [0.074, 0.104]	-0.09	0.97	1000
OLMo-2-7B (L18–20)	base	0.000 [0.000, 0.001]	-0.00	1.00	1000
	$-\alpha d$, α=0.25	0.001 [0.000, 0.003]	-0.00	1.00	600
	random dir., α=0.25	0.000 [0.000, 0.001]	-0.00	1.00	600
	early layers, α=0.25	0.000 [0.000, 0.000]	-0.00	1.00	600
	$-\alpha d$, α=0.5	0.017 [0.008, 0.026]	-0.01	0.99	600
	random dir., α=0.5	0.002 [0.000, 0.003]	-0.00	1.00	600
	early layers, α=0.5	0.000 [0.000, 0.000]	-0.00	1.00	600
	$-\alpha d$, α=1	0.477 [0.427, 0.531]	-0.76	0.76	1000
	random dir., α=1	0.008 [0.002, 0.018]	-0.01	1.00	1000
	early layers, α=1	0.001 [0.000, 0.002]	-0.00	1.00	1000
	$-\alpha d$, α=2	1.096 [1.033, 1.161]	-5.86	0.07	1000
	random dir., α=2	0.386 [0.331, 0.442]	-1.12	0.73	1000
	early layers, α=2	0.001 [0.000, 0.003]	-0.00	1.00	1000
	$-\alpha d$, α=4	0.471 [0.439, 0.503]	-6.44	0.02	1000
	random dir., α=4	0.529 [0.500, 0.560]	-6.82	0.09	1000
	early layers, α=4	0.066 [0.028, 0.113]	-0.00	1.00	1000
Qwen3-8B (L21–23)	base	0.022 [0.010, 0.045]	-0.01	1.00	1000
	$-\alpha d$, α=0.25	0.020 [0.007, 0.048]	-0.01	1.00	600
	random dir., α=0.25	0.030 [0.010, 0.071]	-0.02	1.00	600
	early layers, α=0.25	0.030 [0.008, 0.067]	-0.02	1.00	600
	$-\alpha d$, α=0.5	0.032 [0.020, 0.045]	-0.01	1.00	600
	random dir., α=0.5	0.037 [0.015, 0.077]	-0.02	1.00	600
	early layers, α=0.5	0.030 [0.009, 0.068]	-0.02	1.00	600
	$-\alpha d$, α=1	0.411 [0.342, 0.482]	-0.45	0.89	1000
	random dir., α=1	0.127 [0.111, 0.146]	-0.05	0.99	1000
	early layers, α=1	0.024 [0.011, 0.047]	-0.01	1.00	1000
	$-\alpha d$, α=2	1.466 [1.314, 1.611]	-2.72	0.67	1000
	random dir., α=2	0.376 [0.342, 0.411]	-1.47	0.56	1000
	early layers, α=2	0.013 [0.008, 0.020]	-0.00	1.00	1000
	$-\alpha d$, α=4	0.810 [0.731, 0.894]	-2.56	0.65	1000
	random dir., α=4	0.438 [0.404, 0.474]	-2.48	0.48	1000
	early layers, α=4	0.192 [0.156, 0.231]	-0.04	1.00	1000

Table A29: **Residual-direction interventions on restated traces.** Mean absolute source effect (nats; trace-clustered 95% CI), clean value log-probability, and value accuracy after subtracting the copy-associated direction, a norm-matched random direction, or a copy-associated direction estimated and applied at early layers.

is replaced by a placeholder of the same digit count ("$x_5 = x_3 - x_1 = \texttt{vv} - \texttt{vv} = 12$"). The two corpora have identical sequence lengths, training token budgets, and problem sequences. In its matched rendering, a filler-trained model shows no source suppression (Table A30). Its restated lines have source effects of 19.6–20.7 nats, compared with 7.2–15.2 for variable-form lines, so G is negative. At the same p, the value-carrying control reaches $G = 0.997$–0.999, with correct-copy effects at most 0.03 nats. Length-matched, value-free restatement therefore does not reproduce the substitution effect.

The same runs show why the readout must match the corpus. Evaluating a filler-trained model with digits in the copy slot introduces an unseen rendering. Restated-line accuracy then falls to 0.21–0.23, compared with 0.85–0.98 in the matched rendering, and yields an apparent $G \approx 0.6$–0.7 from off-distribution suppression. The same weights give $G < 0$ in their matched rendering. Two caveats apply. The placeholder preserves digit count, which the model demonstrably uses, so it is value-free only conditional on that feature. Also, G becomes unstable when its denominator is small. We therefore report medians over three seeds alongside raw effects; one $p = 0.95$ seed reaches -5.8.

Training corpus	Evaluation	seeds	restated acc.	no-copy $\lvert\Delta\rvert$	restated $\lvert\Delta\rvert$	G
$p = 0.50$						
values, real	matched	2	0.975	12.83	0.03	$+0.997$
value-free filler	matched	3	0.977	15.15	20.71	-0.38
value-free filler	mismatched	3	0.231	12.87	3.87	$+0.66$
$p = 0.95$						
values, real	matched	2	0.994	5.62	0.00	$+0.999$
value-free filler	matched	3	0.847	7.18	19.62	-0.97
value-free filler	mismatched	3	0.212	6.45	2.04	$+0.60$

Table A30: Training-side length control. Corpora differ only in whether the restated segment carries values; length, token budget, and the problem stream are identical. "Matched" evaluates a model in its training rendering, whereas "mismatched" evaluates a filler-trained model with digits in the copy slot. G is a median over seeds, the effects are means; all runs pass the reporting criteria of this appendix. The mismatched rows quantify the effect of evaluating an unfamiliar rendering; they use the same weights as the matched rows above them. The values-real rows are separate executions of the $p = 0.5$ and $p = 0.95$ configurations with the same seeds, token stream, and evaluation edges as Tables 2 and A32; they start from a different initialization (per-edge effect correlation 0.82–0.95), so their effects do not match those tables exactly.

Consumer competence across the sweep. Both evaluation formats have support in every training corpus, but their accuracies differ as their frequencies change. At $p = 0.05$, where suppression is weak, restated and variable-form value-token accuracies are 0.627 and 0.651. At $p = 0.25$, both are high (0.989 and 0.982), providing a strong-suppression condition with competent prediction in either format. The larger-p models retain large source effects on no-copy lines even as their accuracy deteriorates, reaching 0.028 at $p = 0.95$ (Table A31).

Accuracy loss alone does not determine the bias in G. These endpoints establish retained sensitivity, not reliable computation from the source. Likewise, the moderate low-p accuracy means that the frequency sweep does not isolate corpus frequency from learning progress. The variable-form AUROC remains 0.947–0.994, so the instrument detects source-dependent changes despite these competence limits. Restated-source AUROC approaches chance when the measured source effect becomes small.

Reliability manipulation. At fixed restatement probability $p = 0.5$, a fraction p_{wrong} of restated training lines contains one wrong operand copy. The corrupted operand is chosen at random; the other copied operand, source definitions, and true result remain unchanged. Thus p_{wrong} is a fraction of *lines*, not of all individual operand occurrences. Evaluation uses the same clean mixed rendering for every model. Local interventions are applied at evaluation time to measure dependence, separately from the training corruption.

Optimization. Sequence composition changes with p, so optimizer instability could masquerade as an effect of p. At learning rate 3×10^{-3}, the $d = 256$, six-layer model is unstable at every tested

		restated format			no-copy format				
p	seeds	acc.	NLL	AUROC	acc.	NLL	AUROC	no-copy $\lvert\Delta\rvert$	G
0.05	4	0.627	1.94	0.936	0.651	1.89	0.947	8.69	0.11
0.10	2	0.720	1.40	0.963	0.702	1.44	0.971	9.39	0.08
0.15	2	0.890	0.60	0.956	0.865	0.69	0.990	11.06	0.30
0.20	2	0.969	0.21	0.948	0.954	0.41	0.992	12.10	0.57
0.25	4	0.989	0.06	0.874	0.982	0.33	0.994	12.87	0.91
0.50	4	0.989	0.03	0.825	0.479	0.80	0.994	13.23	0.99
0.75	2	0.972	0.12	0.693	0.039	1.78	0.993	11.73	1.00
0.95	2	0.991	0.02	0.656	0.028	4.62	0.979	7.16	1.00

Table A31: Readout diagnostics across the restatement sweep, averaged over seeds. Every run satisfies the reporting criteria in this appendix; these do not impose a consumer-accuracy threshold. Accuracy and negative log-likelihood (NLL) measure value-token prediction on consumer lines in each evaluation format; AUROC separates direct edges from no-path controls within that format. The expanded diagnostic sweep pools the number of seeds shown; Table 2 and Fig. 4 use the primary two-seed series. High source-effect AUROC is compatible with low value-token accuracy, particularly on the rare no-copy format at large p.

		source effect					value-token accuracy	
p_{wrong}	seed	no copy	with copy	local copy	local ref.	G	no copy	with copy
0.0	0	12.885	0.076	19.661	0.024	0.994	0.504	0.956
0.0	1	13.388	0.066	20.713	0.020	0.995	0.447	1.000
0.5	0	6.970	14.697	0.000	15.668	-1.109	0.830	1.000
0.5	1	6.982	14.203	0.002	14.322	-1.034	0.851	1.000
1.0	0	6.790	15.496	0.004	15.251	-1.282	0.887	0.994
1.0	1	6.455	14.718	0.007	14.729	-1.280	0.851	0.950

Table A32: **Fixed-frequency reliability control, both seeds.** All models train for 40000 steps at $p = 0.5$ and are evaluated at $p_{\text{eval}} = 0.5$. Effects are mean absolute changes in log-probability (nats). Each run has 282 no-copy and 318 restated direct edges from 60 evaluation traces, and 135 local-probe edges. The copy and reference probes replace their respective local segments. The wrong-copy training conditions retain strong source/reference sensitivity while their local-copy effects approach zero. Value-token accuracies distinguish that sensitivity from prediction competence. After corrupting a local copy at evaluation, the mean log-probability advantage of the true result over the copy-implied result is 18.41 and 17.76 nats for $p_{\text{wrong}} = 0.5$ in seeds 0 and 1, respectively ($n = 126$ valid contrasts per seed). In the corrupted-copy conditions, local-copy effects of 0.0004–0.0069 nats lie near their no-path floors of 0.0014–0.0019.

value of p. A 2×2 probe over learning rate and seed at $p = 0.05$ attributes the instability to learning rate rather than an initialization-by-corpus interaction. All reported runs use 1×10^{-3} for $40{,}000$ steps and reach leaf accuracy of at least 0.997. Training loss is not comparable across p: higher-p corpora spread the same problems over more deterministic tokens and have a lower per-token entropy floor (0.508 at $p = 0.05$ versus 0.422 at $p = 0.95$). We use leaf accuracy to check optimization and report consumer accuracy separately.

A5.2 Training Checkpoints

Table A33 tabulates the checkpoint measurements underlying the main text's format-controlled training result.

	base	25%	50%	100%
variable-form: source effect (nats)	1.008	4.045	4.260	4.638
variable-form: restatement rate	0.921	0.000	0.000	0.000
variable-form: accuracy	0.867	0.883	0.983	0.950
mixed: source effect (nats)	1.008	3.494	3.603	3.960
mixed: restatement rate	0.921	0.350	0.580	0.487
mixed: accuracy	0.867	0.900	0.983	0.950
restated: source effect (nats)	1.008	0.559	0.727	0.595
restated: restatement rate	0.921	0.877	0.928	0.993
restated: accuracy	0.867	0.750	0.917	0.967

Table A33: LoRA (Hu et al., 2022) SFT checkpoint measurements for the Gemma-2-2B trajectories in Fig. A6. These are individual runs; the three-seed endpoint summary appears in Table A34. Columns give the fraction of the SFT schedule completed. Source-patch effects are measured on the same held-out variable-form traces at every checkpoint and in every condition. The comparison is therefore not confounded by training-induced changes in generated format.

A5.3 Training Seeds and the Compression Objective

We make two further checks of the training result. The first repeats format-controlled training across seeds. The second tests whether selecting for brevity produces the same change in source sensitivity.

Training format	endpoints (3 seeds)	mean±sd	restatement
variable-form	4.64 / 5.08 / 4.83	4.85 ± 0.22	0.00 (all)
restated	0.59 / 0.77 / 0.79	0.72 ± 0.11	0.99–1.00

Table A34: Source effects (nats) across three Gemma-2-2B SFT seeds; sd denotes standard deviation (base 1.01; mixed arm 4.27 ± 0.27). As the fraction of non-restated training lines increases, endpoint source effects are 0.72, 3.87, 4.27, 4.46, and 4.85 nats at 0%, 25%, 50%, 75%, and 100%. Source sensitivity increases monotonically, with the largest increase between 0% and 25% non-restated training lines. Restatement in generated traces remains 0.44–0.64 through the 50% mixture and falls to 0.01 at 75%. Endpoint accuracy is 0.92–0.97 in every seed. In post-SFT joint interventions (n=600 per cell), the source effect with a clean local copy is 0.004, 0.008, and 0.009 at the base, 50%, and 100% variable-form SFT checkpoints; the corresponding wrong-copy effects are 1.08, 0.97, and 1.10. Training therefore changes source sensitivity primarily in the no-copy regime.

A6 Monitoring and Selection Pressure

A6.1 Coverage, Provenance, and Sensitivity

For each parsed source–consumer edge, V_{value} records whether the consumer writes the source value, V_{prov} whether it names the source, and V_{causal} whether a pre-specified source intervention detectably changes the consumer prediction. A line is locally covered only if every required operand is present. In the Qwen3 loop analysis, C_{line} averages over traces the fraction of *variable-referencing internal*

	source effect (nats)	restate rate	accuracy
base	1.008	0.921	0.867
brevity SFT, 50%	0.734	0.957	0.983
brevity SFT, 100%	1.017	0.849	0.883

Table A35: LoRA SFT on 549 brevity-selected, correct, self-generated traces (34% selection yield), without explicit format control. The endpoint source effect is unchanged, and restatement declines only modestly. A brevity instruction likewise shortens traces by 37% while retaining a restatement rate of 1.0. Generic brevity pressure therefore does not reproduce the effect of selection that explicitly disfavors restatement. The transient 50%-checkpoint dip to 0.73 is consistent with the corpus's 0.96 restatement rate and disappears by the endpoint.

lines that state both operand values. Each of the 100 evaluation traces per round contains at least one eligible line. This macro average differs from a pooled edge-level restatement rate and does not assess lines excluded by the parser.

The quantities need not agree. In the one-shot Qwen3 traces, every parsed dependency names its source, but only 11% restate its value (Table A7). A deterministic line-local checker detects 99.8–100% of the tested result inconsistencies when operands are visible, but none in variable-form traces. Locally consistent copy corruptions instead require a cross-line provenance check. LLM monitors additionally face extraction and reasoning errors (Anwar et al., 2026). These are access and capability distinctions for specific checkers, not certificates of global faithfulness.

A6.2 Tie-Breaking Among Passing Traces

Two Qwen3-8B expert-iteration loops retain only traces passing the same line-local recomputation rule. When required operands are missing, the line passes without an arithmetic comparison. The loops differ in how they select among passing traces: uniformly at random or by the smallest restatement statistic. Each loop uses one training seed, two update rounds, and 100 evaluation traces per round.

Random selection raises C_{line} from 0.083 to 0.120; least-restating selection lowers it to 0.007 (Fig. A1). Edge-level value visibility starts at 0.116 and ends at 0.158 under random selection and 0.010 under least-restating selection. Every retained trace passes, and explicit provenance remains 1.000. Task accuracy starts at 0.910 and ends at 0.980 and 0.960, respectively. The least-restating endpoint still has a 2.4-nat source-patch effect. A pass rate alone therefore cannot identify how much of the trace was checked.

The declining trajectory is induced by the least-restating tie-break, not by monitor optimization alone. A length-minimizing tie-break raises coverage to 0.137 and value visibility to 0.192.

Selection controls in Gemma-2. The separate Gemma-2 loops described in Sec. A1.5 compare shortest-correct selection with least-restating correct selection. Under shortest-correct selection, mean generation length falls from 305 to 254 characters and then plateaus; the first-round selection differential is only 1.6%. Accuracy rises from 0.867 to 0.967, while restatement increases from 0.921 to 0.930, 0.993, and 0.997. The source-patch effect remains at or below baseline ($1.008 \to 0.946 \to 0.763 \to 0.845$ nats). Thus, shortest-correct selection favors compact solutions that still restate their operands. Generic optimization and length pressure therefore need not reduce visibility, consistent with mixed results on length penalties (Little, 2026; MacDermott et al., 2025).

Under targeted selection, the reference-dependent restatement statistic falls to 0.506 ± 0.298 across three seeds, reaching 0.169 in the seed with the largest decline. The source-patch effect rises to 1.206 ± 0.148 nats at accuracy at least 0.98; $\pm$ denotes the standard deviation across seeds. The share of internal steps whose lines expose both operand values does not decline: it reaches 0.967 ± 0.030, compared with a baseline of 0.879. In the seed with line-level measurements, every two-literal line remains arithmetically consistent at every checkpoint. The selected traces increasingly omit source names while preserving locally checkable arithmetic (Sec. A6.4).

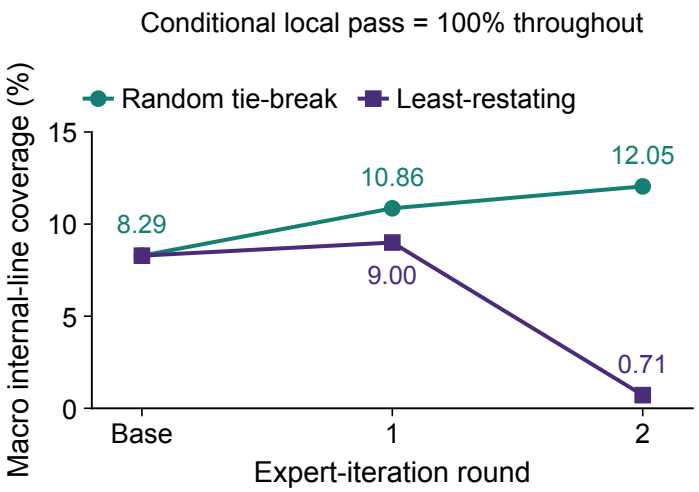

Figure A1: **A pass-rate ceiling does not identify line-local coverage.** Two one-seed Qwen3-8B loops evaluate 100 traces per round. Coverage is the macro mean of the per-trace fraction of variable-referencing internal lines that expose both literal operands. The figure reports conditional arithmetic consistency among locally checkable lines, which remains 100% in both loops. The two selection rules yield different coverage; the example is not an estimate of training variance or general monitor robustness.

A separate audit of targeted Gemma-2 selection illustrates the resulting measurement issue. The reference-dependent restatement statistic falls from 0.899 to 0.584, but a dependency-graph analysis raises value visibility from 0.909 to 0.962. By round three, 38% of traces have no variable reference, versus 1% at baseline. The loss is explicit provenance rather than local operand availability (Sec. A6.4; Fig. A2).

A6.3 Monitor Evaluation

We measure detection of value corruptions in template traces by monitor and access scope. Parser monitors use n=500 samples per cell, and LLM monitors use n=120; parentheses report false-alarm rates on clean lines. We test three corruptions: replacing the result, replacing a local copy while retaining the original result, and replacing a local copy and recomputing the result from it. The last corruption is locally consistent but conflicts with the source.

Monitor	corruption (regime)	line-local	multi-line/full
parser	result (restate)	1.00	1.00
parser	result (variable-form)	0.00	1.00
parser	inconsist. copy (restate)	0.998	0.00*
parser	consist. copy (restate)	0.00	0.996
Qwen3-8B	result (restate)	.69 (.00)	.33 (.00)
Qwen3-8B	result (variable-form)	.01 (.00)	.29 (.01)
Qwen3-8B	consist. copy (restate)	.00 (.00)	.77 (.00)
Gemma-2-2B	result (restate)	.23 (.00)	.37 (.03)
Gemma-2-2B	result (variable-form)	.14 (.21)	.28 (.08)

Table A36: Detection rates for result and copy corruptions under different access conditions; parentheses give false-alarm rates. The deterministic line-local checker detects result inconsistencies when both operand values are visible and detects none in variable-form traces. Locally consistent provenance corruptions evade every line-local monitor in this experiment. The LLM monitors underperform the deterministic resolver even with full access, and Qwen3-8B detects fewer simple restated corruptions with full context than line-local context (0.33 versus 0.69). *The resolver recomputes from resolved definitions and therefore misses copy-only inconsistencies. Combining the two parser checks covers all three corruption types, with detection rates of 0.996–1.00 in these samples.

A6.4 What Targeted Pressure Erodes: Variable References, Not Values

We re-run the targeted expert-iteration loop and retain 150 evaluation traces per round. The loop selects the least-restating correct sample from $k = 4$ candidates for each of 400 problems over three rounds. The reference-dependent restatement statistic in Sec. A6.2 is computed per trace. It measures the share of variable-referencing direct edges for which the source value is written on the consumer line, and assigns zero to traces with no variable references.

Re-parsing the stored traces shows that the share with no variable reference rises from 1% at baseline to 38% after round 3. Among lines that retain variable references, source values remain restated on 94–97% of edges in every round. Among literal-only lines, the share that combines two earlier results without naming them rises from 5% to 26%. The lines continue to state both operand values and remain arithmetically consistent. Thus, the measured decline reflects loss of *source names*, not loss of operand values. Line-local recomputation can still succeed even as the trace stops identifying where those values originated.

A6.5 An External LLM-Judge Monitor Under Targeted Pressure

We evaluate an external judge on the stored traces from Sec. A6.4. Each arithmetically consistent line k is presented twice with only its preceding context: once as written and once with its result replaced by a wrong number of the same digit count. Half of the corruption proposals add or subtract an integer between 1 and 10. The judge answers whether the last result is correct. This protocol tests result corruptions only; locally consistent copy corruptions, which a line-local judge cannot detect without the source, are evaluated with the parser monitors in Sec. A6.3. We disable thinking in both Qwen3-32B and Qwen3-8B. The score $P(\text{No})$ is the first-token probability mass assigned to variants of No, normalized over the combined Yes and No variants. We evaluate up to 150 items per line kind and round, with only 15–29 reference-only items.

We classify lines as *restated* when they contain a variable reference and both operand values, *reference-only* when they contain a variable reference without the values, and *literal-only* when they contain no variable reference. The last category includes leaves and steps whose source names were removed. Table A37 reports AUROC and recall / false-alarm rate at $P(\text{No})>0.5$. The 32B judge discriminates all line kinds well (AUROC ≥ 0.88), with no systematic decline across rounds. At the 0.5 threshold, it is conservative on restated and literal-only lines (recall about 0.5, false-alarm rate ≤ 0.03). It flags the rarer reference-only lines more readily (recall 0.62–0.88, false-alarm rate 0.06–0.18). The 8B judge is weaker throughout (AUROC 0.65–0.82; Table A38), illustrating a monitor-capability gap rather than a change in the traces. The 8B judge's discrimination declines slightly on restated lines, but there is no common decline across line kinds as source names disappear. The small reference-only subsets also limit comparisons across rounds.

		restated		reference-only		literal-only		all
Round	restate stat.	AUROC	recall/FA	AUROC	recall/FA	AUROC	recall/FA	AUROC
base	0.90	0.93	0.57/0.02	0.93	0.79/0.10	0.96	0.53/0.03	0.93
1	0.91	0.95	0.51/0.01	0.94	0.88/0.18	0.98	0.58/0.00	0.95
2	0.87	0.93	0.51/0.00	0.94	0.73/0.07	0.97	0.53/0.00	0.95
3	0.58	0.95	0.51/0.00	0.88	0.62/0.06	0.96	0.48/0.00	0.95

Table A37: **Qwen3-32B judge across targeted-selection rounds.** We report the AUROC of $P(\text{No})$ for corrupted versus clean lines and recall / false-alarm rate (FA) at threshold 0.5, split by line kind. Reference-only lines contain 15–29 items; "restate stat." is the reference-dependent restatement statistic defined in Sec. A6.4.

A7 Pre-Specified Extensions and Supporting Results

A7.1 Outcomes of Pre-Specified Extensions

Before running the extensions in Secs. 4–5, we recorded eight expectations and their decision thresholds. We report all outcomes, including misses.

		restated		reference-only		literal-only		all
Round	restate stat.	AUROC	recall/FA	AUROC	recall/FA	AUROC	recall/FA	AUROC
base	0.90	0.68	0.25/0.01	0.74	0.41/0.14	0.67	0.27/0.03	0.68
1	0.91	0.67	0.18/0.01	0.82	0.65/0.18	0.70	0.27/0.00	0.69
2	0.87	0.66	0.22/0.01	0.76	0.40/0.13	0.68	0.23/0.01	0.67
3	0.58	0.65	0.17/0.01	0.75	0.44/0.12	0.68	0.21/0.01	0.67

Table A38: **Qwen3-8B judge with thinking disabled.** AUROC and recall / false-alarm rate (FA) at $P(\text{No}) > 0.5$, using the protocol and line categories of Table A37. "Restate stat." is the reference-dependent statistic defined in Sec. A6.4.

1. A correct local copy would remove at least 90% of the no-copy effect for words and names in every model: *met* (signed ratios ≤ 0.07).
2. Length-matched filler would remove at most 20%: *met* for words (0.86–0.92 remains), but *not met* for names (0.63–0.78 remains).
3. A wrong local copy would leave 0.2–0.8 of the effect: *met*.
4. Remote duplicates and bare words would leave at least 60%, as for numbers: *not met*. Only 13–51% remains for words and names; the main text reports this value-type dependence.
5. Counterfactual-result frequency would be at least 0.70 for words and names, with CDE_0 at most 0.05: the outcome-frequency threshold was met. The CDE_0 threshold was met for words in Gemma-2 (-0.03), but not for names in Gemma-2 (0.07), words in Qwen3 (0.06), or names in Qwen3 (0.22).
6. Δ_S would increase with size: *met* as an overall descriptive association (Spearman $\rho = 0.83$), though not monotonically below 10B and with base architecture varying across the series.
7. Subtracting the copy-associated direction would raise the mean absolute source effect on restated edges at least $3\times$, while controls would remain below $1.5\times$: the intervention threshold was met at $\alpha = 1$, but the random-direction controls exceeded the $1.5\times$ limit in all three models.
8. Steering at $\alpha \leq 2$ would keep accuracy within five percentage points: *not met* at $\alpha = 2$ in any model, or at $\alpha = 1$ in OLMo-2 and Qwen3. Only $\alpha \leq 0.5$ satisfies the criterion across all three models (Table A29).

A7.2 Additional Intervention Results

This section collects supporting quantities cited in the abstract or main text.

Joint cell (remote, local)	n	mean	median
patched, clean	491	0.0037	0.0015
clean, corrupted	491	3.823	3.752
patched, corrupted	491	4.410	4.313

Table A39: Joint source–copy interventions (experiment J1) in Gemma-2-2B, reported as the log-probability drop in nats from the all-clean baseline. The source effect is 0.0037 with a clean local copy and 0.587 with a corrupted local copy, for an interaction of $+0.583$. In cross-model replication (n=491 each), the corresponding effects rise from 0.010 to 0.227 in Qwen3, 0.0002 to 1.548 in OLMo-2, and 0.004 to 0.946 in Gemma-2-27B. The conditional increase appears in all four models, with architecture-dependent magnitude.

A7.3 Additional Figures and Tables

This section collects the enlarged plots and supporting tables referenced from the main text.

Generation-time cell	measure	value
restated, baseline	original-result frequency	.997
restated, patched	original / counterfactual frequency	.992 / .001
variable-form, patched	median shift (fraction > 0)	$+1.4$ (.77)
variable-form, baseline-correct	mean shift (fraction > 0)	$+4.3$ (1.00)

Table A40: Generation-time interventions (G1/G1b; n=776 behavioral samples, 761 logit-level samples, and 138 baseline-correct samples). Shift is the change after source patching in the counterfactual-versus-original log-probability margin at the first result token, measured in nats. Cases with identical first tokens are excluded; parentheses give the fraction of positive shifts. The no-copy logit evaluation is distinct from the G3 experiment that regenerates the consumer line. G3's outcomes and single-edge agreement metric are reported over all saved samples in Table A18.

Transfer domain	restated	non-rest.
k-hop (G/Q/O)	.037 / .0004 / .018	.52 / .92 / 1.17
GSM8K (G/Q/O)	.043 / .020 / .009	1.55 / .74 / 3.33

Table A41: Mean signed source effects (nats) in the transfer tasks; G/Q/O denotes Gemma-2/Qwen3/OLMo-2. Relational traces measure direct-edge effects under assigned formats. The GSM8K row measures effects on the final parsed result, grouped by whether the source value appears again later in the trace; it does not isolate a copy on one consumer line. Non-restated effects are 14–2300$\times$ the restated effects for relational traces and 36–378$\times$ for the pooled GSM8K runs.

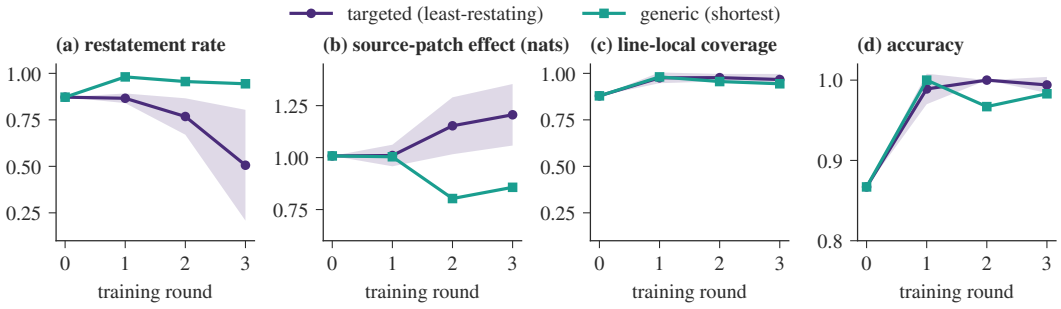

Figure A2: **Gemma-2 selection-loop diagnostics.** Restatement rate, source-patch effect, full-line coverage, and accuracy across the least-restating and shortest-correct loops described in Sec. A1.5. Purple shows the targeted-selection mean with one-standard-deviation bands over three seeds; teal shows one shortest-correct run. The restatement statistic depends on explicit variable references, so it can fall while full-line coverage remains high (Sec. A6.4).

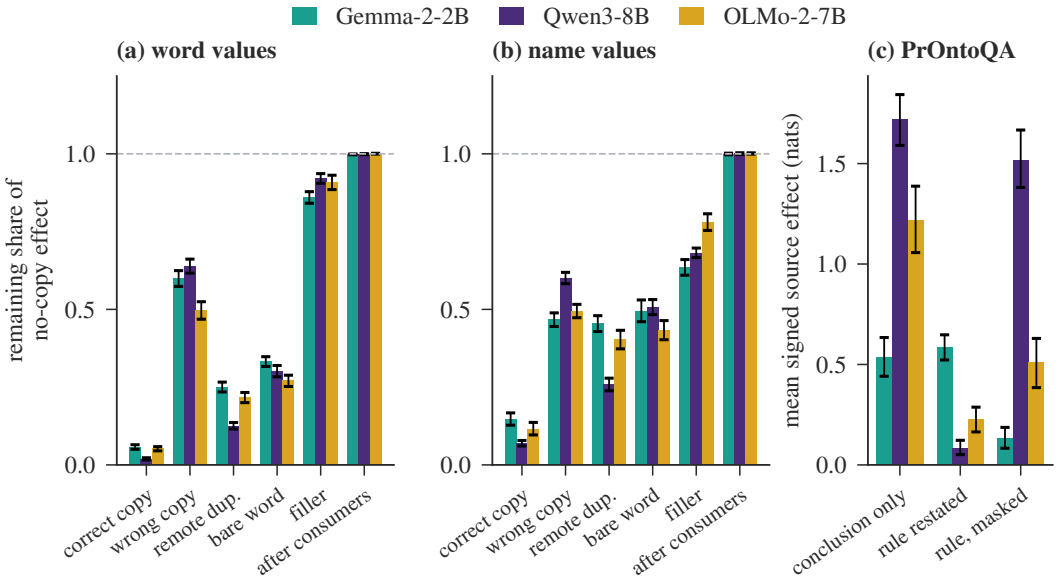

Figure A3: **Local-copy substitution beyond arithmetic.** (a,b) Share of the no-copy source-patch effect remaining in each copy configuration for word and name values across three models. We report $|\Delta|$ ratios on value-changing word edges with trace-clustered 95% CIs; Table A20 gives signed values. A correct line-local copy removes 85–98% in every cell; wrong copies attenuate partially; for these value types remote duplicates and bare words also remove half to seven-eighths, whereas a bare number removes almost nothing. (c) On PrOntoQA internal edges, rule restatement reduces the mean signed source effect in Qwen3-8B and OLMo-2-7B but not in Gemma-2-2B, whose conclusion-only value accuracy is lower. In Qwen3 and OLMo-2, masking the rule's consequent weakens substitution.

Model	forced arm	rest. share	source effect (nats) rest. / non-rest.	acc.
Gemma-2	restate	99.8%	0.009 / —	0.957
	variable-form	84.4%	0.010 / 2.37	0.803
Qwen3	restate	100%	0.010 / —	1.000
	variable-form	0.9%	0.002 / 2.68	0.821
OLMo-2	restate	99.9%	0.0003 / —	0.988
	variable-form	92.9%	0.0007 / 7.20	0.963

Table A42: **Prompt-forced format interventions.** Realized restatement share, mean signed source-patch effect (nats) split by realized restatement, and answer accuracy in six prompt conditions. Source effects on restated edges are near zero in every condition, whereas non-restated effects are about 2.4 nats or more. Restated edges have small source effects under either prompt. Non-restated edges are rare under restatement prompts, limiting comparisons between prompt arms in the no-copy condition. Each arm evaluates 1340–1854 direct edges. Trace-clustered 95% CIs for non-restated edges under variable-form prompts are $[2.20, 2.54]$, $[2.60, 2.78]$, and $[6.18, 9.23]$; Gemma-2 conditions replicate across seeds.

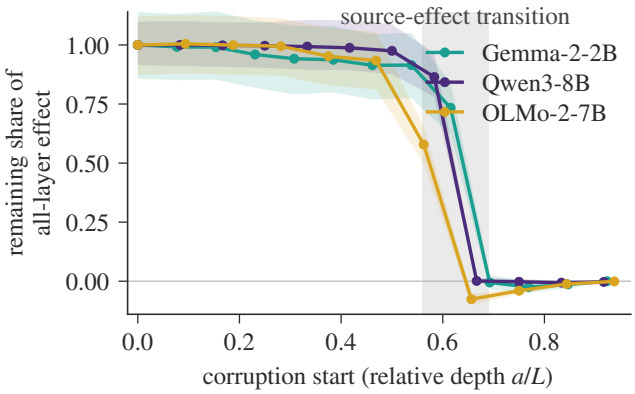

Figure A4: **Depth localization of the source-patch effect.** Remaining share of the all-layer source-patch effect when source activations are patched from relative depth a/L through the final layer. Dense 2–3-layer sweeps use trace-clustered bootstrap 95% bands. The transition occurs at L16–18 of 26 in Gemma-2-2B, L21–24 of 36 in Qwen3-8B, and L18–21 of 32 in OLMo-2-7B, corresponding to 56–69% relative depth.

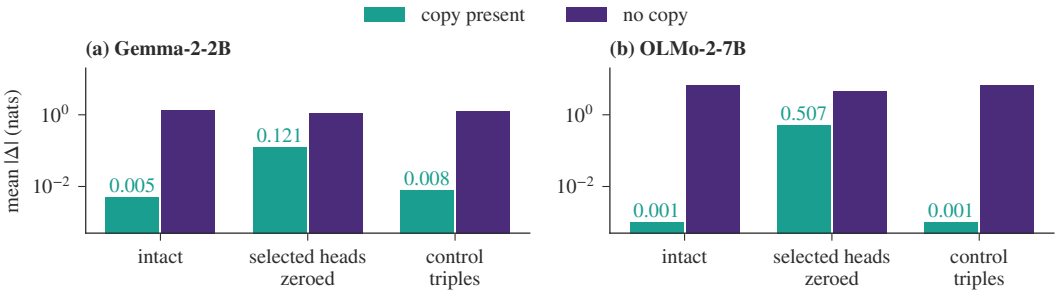

Figure A5: **Partial recovery of source sensitivity under head ablation.** Mean absolute source effect with a line-local copy present or absent under three head conditions (n=600 edges per cell; log scale). Zeroing three source-switching heads raises the correct-copy effect 24× in Gemma-2 and 500× in OLMo-2, but recovers only 8–9% of the no-copy effect. Matched same-layer control triples remain near floor, and the no-copy effect declines only modestly.

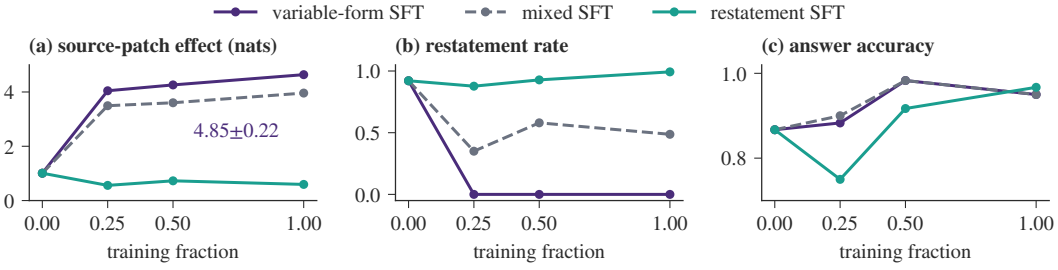

Figure A6: **Format-controlled training shifts source sensitivity.** Source-patch effect (left), restatement rate (center), and accuracy (right) across LoRA SFT checkpoints. Curves show individual runs; the annotation reports the three-seed endpoint mean ± standard deviation. Variable-form training increases the effect 4.8× to 4.85±0.22 over three seeds; restatement training lowers it below baseline to 0.72±0.11; and the 50/50 mixture (dashed) approaches the variable-form endpoint. Across seeds, endpoint accuracy remains 0.92–0.97.