A Four-Stage Decomposition of Word-Problem Solving and Mechanistic Fragility in LLM Math Reasoning

Zhongdi Qu
Cornell University
zq84@cornell.edu

Carla P. Gomes
Cornell University
gomes@cs.cornell.edu

Abstract

Large language models solve grade-school math word problems with high accuracy, yet a single irrelevant clause inserted into the problem can collapse it. We reconcile these observations with a mechanistic account. We show that the model's internal computation decomposes into a four-stage sequential pipeline, Schema Abstraction, Operation Planning, Operand Binding, and Computation, each stage producing a distinct intermediate representation in an identifiable band of layers. Using the same scaffold to diagnose distractor-induced failure, we localize the corruption to a single stage, Operation Planning, implemented by a set of attention heads whose causal role we validate bidirectionally. In short, we provide a mechanistic interpretation of math word problem reasoning in LLMs, and their failure when distracted.

1 Introduction

Two recent lines of work on LLM math word problem solving point in opposite directions. Mirzadeh et al. (2025) show that inserting an irrelevant "NoOp" clause into a grade-school math word problem collapses accuracy, and conclude that LLM math reasoning is closer to surface pattern-matching than to genuine inference. Cheng et al. (2025) report a two-stage *abstract-then-compute* pipeline in which the model first encodes the problem as an abstract schema and then computes the result numerically, evidence that LLMs do not rely on shallow pattern-matching shortcuts. We unify both findings into a single mechanistic account: LLMs do solve math word problems via an orderly pipeline, and NoOp fragility arises from the failure of a single stage in that pipeline, implemented by a specific set of attention heads. Pipeline structure and selective fragility coexist.

We present a four-stage pipeline: Schema Abstraction, Operation Planning, Operand Binding, and Computation. Each stage produces a distinct intermediate representation in the residual stream. We identify the stage boundaries with three independent methods: residual cosine similarity exposes them from layer-to-layer dynamics alone, per-category linear probes pinpoint what changes at each boundary, and cross-prompt activation patching tests which residual-stream changes causally determine the final answer. The three methods converge on the same four-stage decomposition.

We then use this scaffold to diagnose NoOp fragility. Activation patching localizes NoOp-induced failure to the Operation Planning stage. An engagement-anchored per-head Direct Logit Attribution identifies a compact set of attention heads that route through the distractor clause, splitting into subsets that write *toward* and *against* the wrong operation plan. Bidirectional attention scaling validates the heads causally: amplifying the anti-wrong-plan subset on NoOp prompts recovers a substantial fraction of the lost accuracy, while ablating either subset on data the model normally solves causes a catastrophic accuracy collapse. Together, these results identify the heads as the mechanistic instrument of Operation Planning, and NoOp fragility as what happens when this instrument is overwhelmed by distractor information.

Our Contributions. (1) A mechanistically-grounded four-stage decomposition of LLM word-problem solving, and initial evidence that such decomposition generalizes to other reasoning domains. (2) A circuit-level diagnosis of NoOp fragility that localizes the failure to Operation Planning, implemented by a compact set of attention heads. (3) A training-free lever on the failure, i.e., a single global scalar that recovers 0.51 of NoOp failures, locating where any future repair must act.[1]

[1] Code and datasets are available at https://github.com/deliaqu/llm-reasoning-decomposed.

Findings of the Association for Computational Linguistics: EMNLP 2026, pages 17383–17402
October 24-29, 2026 ©2026 Association for Computational Linguistics

2 Background and Related Work

2.1 Fragility of LLM math reasoning

Mirzadeh et al. (2025) introduce GSM-SYMBOLIC, which re-instantiates GSM8K (Cobbe et al., 2021), the foundational dataset of grade-school math word problems, from *templates*: problem outlines whose entity names and numeric values are slots, so that the instances of one template share an outline and differ only in the slot values. They report that language model accuracy drops by up to 65% when an irrelevant "NoOp" clause, a sentence that adds no information that a competent reasoner should use, is inserted into the prompt. They argue that this degradation reveals that LLMs solve math word problems by pattern matching rather than genuine reasoning. Mirzadeh et al. (2025) is part of a broader line of work that probes math-reasoning robustness by injecting computationally irrelevant text into GSM8K problems, including Shi et al. (2023), Li et al. (2024), and Yang et al. (2025). These critiques agree at the behavioral level but stop short of localizing the failure inside the model's computation. Concurrently with our work, Han et al. (2026) give a mechanistic analysis of sensitivity to meaning-preserving perturbations and propose an architecture-level taxonomy of failure modes. They characterize how fragility differs across architectures, while we localize such failures internally. Concretely, we answer the question: when an LLM fails to solve a math word problem that contains an irrelevant "NoOp" clause, which modules inside the model are responsible?

2.2 Mechanistic accounts of LLM math

Earlier work provides accounts of how LLMs perform various skills involved in solving math word problems, including property representation (Gurnee and Tegmark, 2024), entity binding (Wang et al., 2022), and arithmetic computation (Stolfo et al., 2023). Each of these examines a particular skill required for solving math word problems, without analyzing how those skills compose into a complete solution. By contrast, we demonstrate that a four-stage reasoning pipeline organizes the full process in LLMs solving math word problems.

From the pipeline perspective, the work most closely related to ours is Cheng et al. (2025), who identify a two-stage *abstract-then-compute* mechanism for LLMs solving math word problems. They identify specific layers where abstraction occurs, followed by layers where arithmetic computation takes place. However, this two-stage pipeline is identified using synthetic one-step problems with restricted vocabulary, and is confined to the direct prompting regime, disabling Chain-of-Thought (Wei et al., 2022), a standard component of LLM reasoning. Their analysis also focuses on the mechanisms underlying successful problem solving, rather than on model fragility. We close these gaps: we demonstrate that a staged reasoning pipeline operates on real, multi-step problems such as those in GSM8K, across direct and CoT regimes, and we identify the modules in this pipeline that are causally responsible for the failure mode.

2.3 Mechanistic methods

We use three standard tools from the mechanistic interpretability literature. Linear probing (Alain and Bengio, 2016; Conneau et al., 2018) trains a linear regression on residual activations to test whether a target feature is linearly decodable. Since attention and MLP input projections and the unembedding all read the residual stream linearly (Elhage et al., 2021, 2022), a probe drop shows that a feature has left the linearly accessible representation at the probed position. Activation patching (Vig et al., 2020; Meng et al., 2022) replaces a target residual with a donor activation and measures the resulting change in output, identifying where causally relevant information lives. It, on the other hand, does not make any linearity assumption. Direct Logit Attribution (Wang et al., 2022) projects each model component's output onto the unembedding direction of a target token, isolating which components write toward or away from that token.

3 The Four-Stage Hypothesis

Refining on the two-stage abstract-then-compute view of Cheng et al. (2025), we hypothesize that math word problem reasoning is composed of four stages. Each stage produces a distinct intermediate representation, and each can fail independently. Errors from any stage propagate and result in wrong final answers. Our decomposition is informed by cognitive science accounts of human math word-problem solving, which typically distinguish three or four cognitive operations: parsing the narrative into a propositional "situation model", selecting an appropriate solution plan, and executing the resulting computation (Kintsch and Greeno, 1985; Hegarty et al., 1995).

Consider the GSM8K question: *"Janet's ducks*

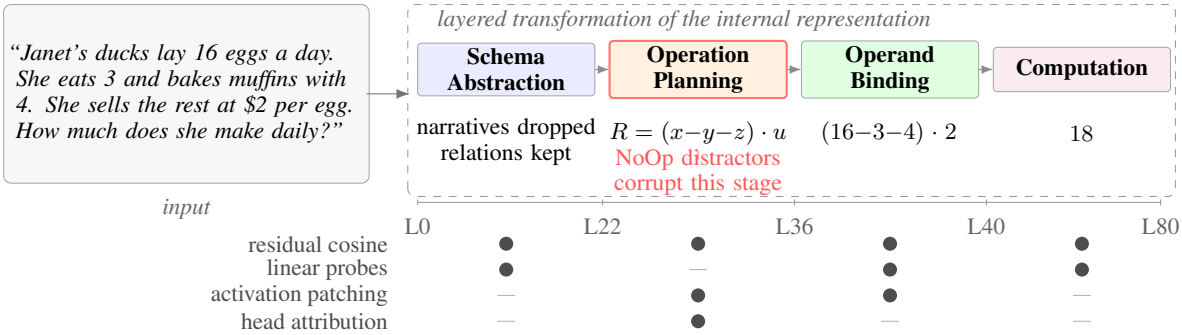

Figure 1: Overview of the four-stage pipeline, illustrated on one GSM8K problem. The problem is the model's input. The four stages are internal, successive transformations of the residual that represents the problem. Each stage leaves a distinct intermediate representation: surface narratives are stripped (Schema Abstraction), an operation plan is formed over abstracted variables (Operation Planning), prompt values are bound into it (Operand Binding), and the result is computed (Computation). Layer boundaries for Llama-3.3-70B are marked below, and the lower grid shows which method establishes each stage (filled dot) or does not speak to it (dash). Boundaries are placed by an automatic criterion on the dynamics of residual similarity between problems, and linear probes and activation patching are used to interpret the content of the representations at those boundaries (§5). Direct logit attribution additionally localizes NoOp fragility to Operation Planning (§6).

lay 16 eggs per day. She eats three for breakfast every morning and bakes muffins for her friends every day with four. She sells the remainder at the farmers' market daily for $2 per fresh duck egg. How much in dollars does she make every day at the farmers' market?" Solving this problem requires four distinct cognitive steps, each producing a result that the next step operates on. **Stage 1: Schema Abstraction.** The solver abstracts away surface narrative details such as entity names and story-specific wording ("Janet", "ducks", etc.), while preserving the underlying relational structure of the problem: a quantity x is produced daily, quantities y and z are consumed, the remainder is sold at price u per unit, and the goal is revenue R. The resulting representation captures the problem schema, but not yet the operations. **Stage 2: Operation Planning.** Identify the sequence of operations needed to solve the abstracted problem schema: $R = (x - y - z) \cdot u$. The output encodes the operation plan, but not yet the concrete operand values. **Stage 3: Operand Binding.** Substitute concrete prompt values into the operation plan: $x \leftarrow 16$ (eggs laid per day), $y \leftarrow 3$ (eaten for breakfast), $z \leftarrow 4$ (used for muffins), $u \leftarrow 2$ (dollars per egg). The output is the fully grounded numeric expression $(16-3-4)\cdot 2$. **Stage 4: Computation.** The solver computes $(16 - 3 - 4) \cdot 2 = 18$ and reads out the answer 18.

We ask whether LLMs develop an analogous sequential decomposition that appears as distinguishable transformations in the residual stream, and if so, which stage fails under distraction. §5 shows that the model's internal representations undergo a four-stage transformation during math word-problem solving. §6 then identifies which stage breaks under an irrelevant "NoOp" distractor.

4 Experimental Setup

We use Llama-3.3-70B-Instruct (Grattafiori et al., 2024) (80 layers) as the main target, with Qwen-2.5-14B-Instruct (Yang et al., 2024) and Gemma-2-9B-it (Gemma Team, 2024) replications to show generalization in App. F. All prompts follow the GSM8K answer format: the model emits the marker #### immediately before the final numeric answer (full template in App. A).

We use four primary datasets, two from Mirzadeh et al. (2025): **GSM-Symbolic**, templated variants of GSM8K math word problems, and **GSM-P1**, which extends each GSM-Symbolic problem with one extra clause that requires one additional computation step. Mirzadeh et al. (2025)'s NoOp dataset is not released, so we re-create **GSM-NoOp** via a human-in-the-loop process described in App. A. We similarly construct a control **GSM-Filler**, matched in token length to the NoOp. Filler questions are of the same structure and contain the same numeric tokens as NoOp questions, but the Filler clauses are neutral descriptions that are less distracting than the NoOp. We also created **padded-Symbolic**, a length-controlled substitute for GSM-Symbolic used in the Stage 2 patching experiments (§5.1). Each GSM-Symbolic question

is padded with a computation-free clause matched in length to the extra clause of its GSM-P1 counterpart, so that a (padded-Symbolic, P1) pair differs in the operations the problem requires while matching narrative, operand values, and token length. Without this control, any patching effect between Symbolic and P1 could be attributed to their length difference rather than to the extra operation. App. A gives details on all datasets and their construction. Table 4 reports Llama-3.3-70B's performance on every dataset.

All metrics are aggregated at the template level. For each template we compute the mean over its instances, then report the mean across templates. Confidence intervals are 95% template-bootstrap CIs (cluster bootstrap with templates as the resampling unit), reflecting variability across problem outlines rather than within-template instance noise.

5 Identifying the Four-Stage Pipeline

We extract the answer-prefix residual (the residual stream at the token position immediately before the model emits the final answer) at every layer of the model on the GSM-Symbolic subset the model solves correctly ($n = 1{,}200$ across 75 templates under direct prompting). Two independent analyses on this layer sequence converge on the same boundary layers (Fig. 2). A pairwise-similarity analysis (panel a) tracks *when* the residual representation changes from layer to layer, and per-category linear probes (panel b) identify *what* content enters or leaves at each transition.

Residual similarity (Fig. 2 a). At each layer we compute the cosine similarity between every pair of answer-prefix residuals and split pairs by template membership. *Within* pairs share a problem outline, and *cross* pairs do not. Four boundary drops emerge automatically (layer-to-layer drop ≥ 0.025 at $z \geq 3$). At L22, cross-template similarity drops while within-template similarity stays flat. The model has begun separating different problem outlines but has not yet differentiated instances within an outline. At L36, cross-template similarity drops again with the within-template curve still flat, indicating further consolidation of the outline-level structure. At L40, within-template similarity finally drops, signaling that instance-specific content such as operand numbers has entered the residual. Both curves decline together at L80 as the computation concludes and the answer is read out.

Residual content probes (Fig. 2b). At the same answer-prefix position, we train per-layer binary linear probes for four target categories: surface *entity* names from the question narrative ("Janet", "ducks", etc.), *operators* (+, −, ×, ÷), *number* tokens used as operands, and the gold *final-answer* value. Probes use template-disjoint cross-validation so no template appears in both train and test, forcing the probe to generalize across problem outlines rather than memorize template-specific encodings. The probe trajectories align with the boundaries from panel (a). Entity decodability collapses at L22 from 0.78 to chance (0.53) and stays at chance thereafter, while operator decodability rises and remains elevated. By L22 the residual has stopped representing surface words and now carries an abstract operator schema. Number-token accuracy jumps from chance to 0.77 at L40, where within-template similarity also first drops, indicating that template-specific operand numbers have entered the answer-position residual. Final-answer accuracy climbs from chance at L40 to 0.93 at L80, the signature of computation.

The similarity analysis tells us *when* the residual changes, and the probes tell us *what* changes. The two readouts agree on three of the four layer indices ($L22/L40/L80$) despite using no shared parameters. On that basis we identify the three boundaries, each boundary identified automatically from the similarity dynamics, with three stages, each stage's interpretation supported by the probes readout. L22 marks Schema Abstraction (surface entities no longer represented), L40 marks Operand Binding (operands enter the residual at the answer prefix position), and L80 marks Computation (the final answer appears). The staged structure is also observed in other datasets and domains. The same experiments reproduce all four signatures on SVAMP (Patel et al., 2021), and recovers a staged structure on PhantomWiki (Gong et al., 2025) multi-hop question answering (App. G).

Besides these three aforementioned boundaries, the similarity analysis also indicates that cross-template residuals experience a drop in similarities at L36. However, direct probes show no significant results there so we cannot directly interpret what has entered the residuals at L36. Unlike the other stages, Operation Planning does not show up in direct probes since full operation plan has no objective label to probe against: the annotated formula is one of many equivalent forms and the model may plan in another. The causal patching experiments that follows in §5.1 fill in this gap,

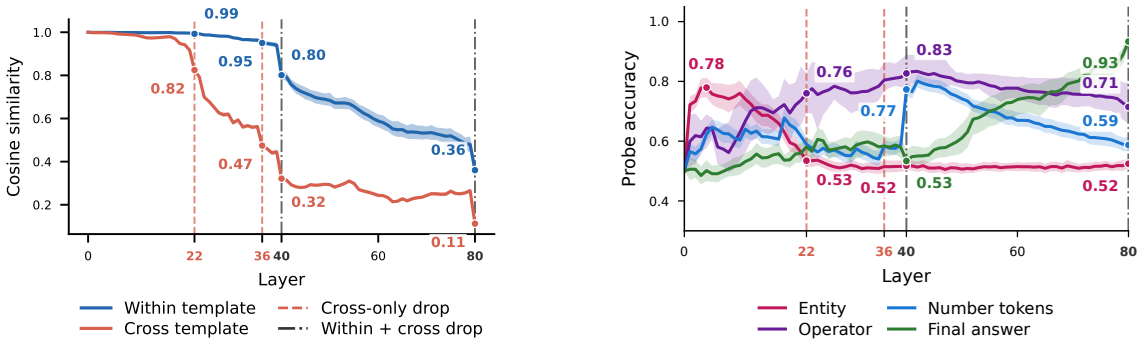

Figure 2: Four-stage signature in the answer-prefix position residual. **(a) Left: residual cosine similarity.** Mean pairwise cosine between within-template (blue) and cross-template (orange) prompts at each layer. Four drops emerge at L22, L36, L40, and L80. **(b) Right: token-presence probes.** Per-layer linear-probe accuracy for entity tokens, operators, operand numbers, and the gold final-answer value. Entity decodability collapses at L22 ($0.78 \rightarrow 0.53$), operand numbers jump to 0.77 at L40, and final-answer rises from chance to 0.93 between L40 and L80. Robustness checks in App. B and App F.

identifying L36 as the Operation Planning boundary. Because Schema Abstraction, Operand Binding, and Computation are directly visible in the probe signatures, we focus the targeted patching analyses below on Operation Planning. §5.2 adds a mechanistic account of the Operand Binding stage.

5.1 Stage 2: Operation Planning

We localize Operation Planning using activation patching between GSM-Symbolic and GSM-P1. For every Symbolic question, its P1 counterpart adds one extra clause that introduces one extra operation, so the two prompts differ in their required operation set. If the model plans operations during a stage, its representations of the two prompts must encode different plans by the end of that stage, and that difference must causally affect the final answer. We use padded-Symbolic as a length-controlled substitute for Symbolic (App. A for details). We run two cross-prompt activation patching experiments on (padded-Symbolic, P1) pairs, restricted to instances where the model answers correctly on both sides ($n = 3{,}763$ across 87 templates). Both setups are illustrated in Fig. 3.

(i) Question-span patching (Fig. 4a). At each layer, P1's residual at every problem-statement token position is overwritten by padded-Symbolic's residual at the same positions. The forward pass then resumes and the model freely generates its own CoT and final answer. We report the normalized patching effect ρ on $\log P(\text{final answer})$ at the answer prefix. For each pair, $\rho = (\ell^*_{\text{tgt}} - \ell_{\text{tgt}})/(\ell_{\text{src}} - \ell_{\text{tgt}})$, where ℓ_{src} and ℓ_{tgt} are the unpatched $\log P(\text{P1's gold answer})$ values on padded-Symbolic and P1 respectively, with $\rho = 0$ meaning no transfer and $\rho = 1$ meaning full transfer to padded-Symbolic's baseline log-probability.

(ii) Single-token patching at the end of an injected CoT (Fig. 4b). Both prompts are extended with the same CoT trace, taken from the gold CoT for the corresponding GSM-Symbolic instance. The trace solves padded-Symbolic correctly by construction but *omits* the extra step P1 requires. At each layer we overwrite only P1's residual at the last token of the injected CoT (the position immediately before where the model would emit the answer prefix ####) and let the model continue. By giving P1 a CoT that is valid only for padded-Symbolic, we force the model to decide whether the provided trace matches the question's operation plan. If the plan has crystallized at layer L, patching padded-Symbolic's residual there should make the model commit immediately, raising $\log P(\#\#\#\#)$ at that position. Otherwise the model continues to plan additional steps for P1's actual answer, and $\log P(\#\#\#\#)$ remains low. We report the normalized patching effect ρ on $\log P(\#\#\#\#)$. The commit-readiness metric isolates the Operation Planning stage by testing whether the model has formed a plan consistent with the provided CoT, without conflating that signal with later-stage answer production.

Both panels converge on the same transition boundaries. In Panel (a), padded-Symbolic's question-span residual fully restores P1's gold-answer log-probability at $\rho = 1.00$ before L22, then collapses across L22–L36 to $\rho = 0.21$ at L36. By the end of Operation Planning, the model has ex-

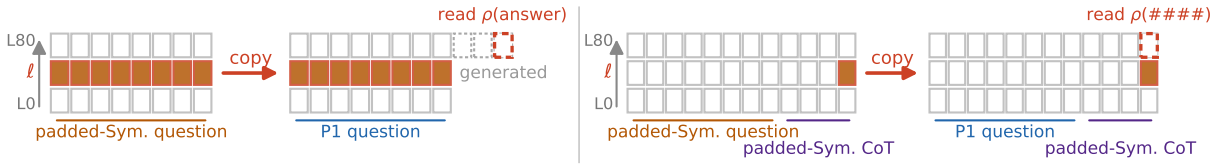

Figure 3: Operation Planning patching experiments. (padded-Symbolic, P1) pair differs in exactly one operation, so any causal effect is attributable to that operation alone. **Left:** Question-span patching. Every question-span state is replaced, the model then generates its own CoT, and ρ is read at the answer marker it emits, on the next-token $\log P(\text{P1's gold answer})$. An effect here means the operation plan has been formed in the patched activations. **Right:** End-of-CoT patching. Only the final state of an injected CoT is replaced. That CoT solves padded-Symbolic but omits the step P1 requires, and ρ is read at that last layer on the next-token $\log P(\texttt{####})$. An effect here means the patched activations carry padded-Symbolic's plan, so the model treats the injected CoT as sufficient and commits.

tracted its operation plan from the input P1 question and no longer hands its answer over when patched. Panel (b) reports the complementary single-token result on commit-readiness. It rises from 0.08 at L30 to 0.88 at L36, saturating over the remaining layers. Decomposing the patching effects into the attention and MLP modules gives a first read on the underlying mechanism. The attention-output patch spikes at the same time as the full-residual rise, indicating that attention is the active component during Operation Planning. §6 returns to this attention pathway with a finer head-level decomposition under NoOp-induced Operation Planning failure.

5.2 Stage 3: Operand Binding

GSM-Symbolic provides a natural testbed for localizing Operand Binding. Problems within a template share structure but differ in their instantiated operand values, so the representational gap between two same-template instances isolates exactly the binding of operands. We run two within-template patching experiments on length-matched pairs that the model solves correctly on both sides.

(i) Question-span patching (Fig. 5a). At each layer, the target's residual at every question-token position is overwritten by the source's residual at the same positions. Two instances of one template differ only in their operand values, so replacing the question span replaces exactly those values and nothing else. The forward pass then resumes and the model freely generates its CoT. We report the normalized patching effect ρ on $\log P(\text{target's gold answer})$ at the answer prefix, where $\rho = 0$ is no transfer and $\rho = 1$ is full transfer ($n = 1{,}387$ pairs across 99 templates). **(ii) Single-token patching at the end of direct prompt (Fig. 5b).** In direct mode we patch the target's residual at only the answer-prefix position immediately before the model's emitted answer ($n = 333$ pairs across 50 templates).

Both panels converge on L40 as the Operand Binding boundary. Panel (a) shows a sharp cliff at L40: question-span transfer holds at $\rho = 0.62$ at L39 and falls to 0.26 at L40, a single-layer collapse. By L40 the operand binding has already been decided on the target side, so patching the question span after L40 can barely shift the answer. Panel (b) patches the answer-prefix position alone, decomposing the transport mechanism into attention and MLP modules. Attention spikes at L40, while the MLP trace stays relatively flat with a late rise during Computation. This resembles the IOI-style mover-head signature (Wang et al., 2022). At L40, attention heads move operand information from the question's value-token positions into the answer-prefix residual, where it binds into the operation plan established upstream.

Sections 5.1 and 5.2 separate the Operation Planning and the Operand Binding stages using the same experiments run on different data pairs, which separates operation structure from operand values. The Operation Planning pair varies the required operations while matching operand values, and its effect concentrates in the band ending at L36. On the other hand, the Operand Binding pair varies operand values while holding the operation plan fixed, and its effect appears only at L40.

6 Diagnosing NoOp Fragility

We localize NoOp fragility by mirroring the Stage 2 question-span patching from §5.1, this time asking whether NoOp's wrong answer can be rescued by Filler's question representation. On (Filler-correct, NoOp-wrong) pairs ($n = 346$ across 25 templates), at each layer we overwrite NoOp's residual at every problem-statement position with Filler's, let the model freely generate its CoT, and measure the

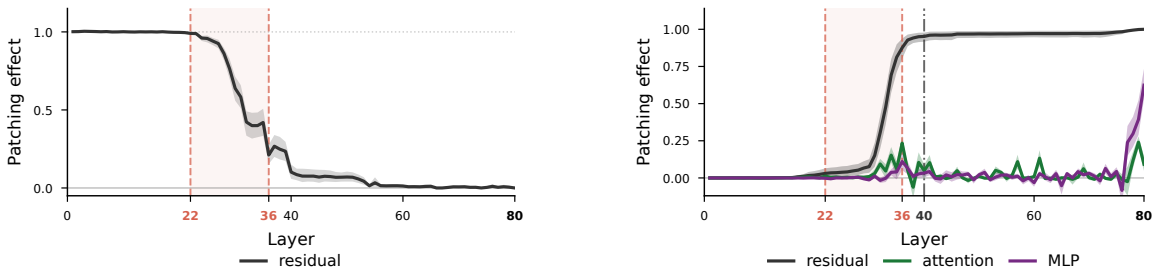

Figure 4: Stage 2: Operation Planning at L36. Both setups are defined in Fig. 3. **(a) Left: CoT-mode question-span patching.** Patching effect stays near 1 below L22 and decays to 0.21 by L36. By L36, the operation plan has been extracted and patching the question representations no longer changes the answer. **(b) Right: end-of-CoT patching**, decomposed by component. The full-residual trace rises from 0.08 to 0.88 across L30–L36. By L36 the patched state at the end of CoT carries padded-Symbolic's plan, so the model treats the injected CoT as sufficient and commits. Attention spikes simultaneously with the rise for the full residual. §6 further dissects the attention mechanism.

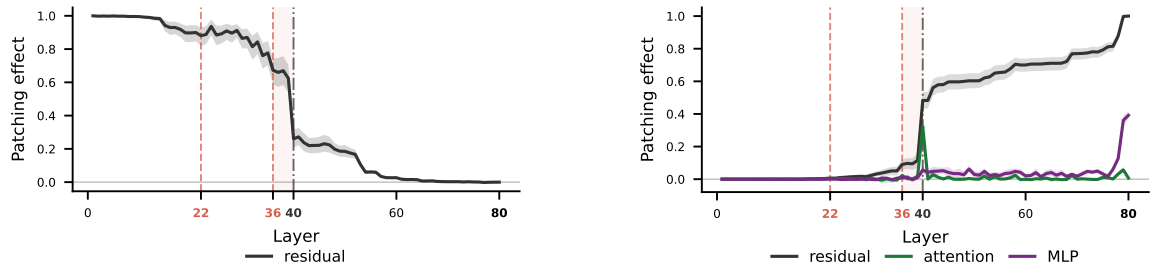

Figure 5: Stage 3: Operand Binding at L40. Both panels patch between two instances of one template, which differ only in their operand values. **(a) Left: CoT-mode question-span patching.** The patching effect holds at 0.62 at L39 and falls to 0.26 at L40, a single-layer collapse. By L40 the operands have been bound on the target side, so supplying the source's values no longer changes the answer. **(b) Right: direct-mode patching at the answer prefix**, decomposed by component. The full residual jumps from 0.12 at L39 to 0.48 at L40, indicating that the operand values have entered the answer-prefix position. Patching the attention output alone reaches 0.32 at L40 and returns to zero immediately after, while the MLP stays near zero until Computation. Attention is therefore what carries operand values into the answer-prefix residual, an IOI-style mover-head signature.

patching effect on $\log P(\text{final answer})$, normalized so 1 is full transfer and 0 is none.

Figure 6a shows the patching effect staying near 1 below L22, collapsing afterwards, and reaching to near 0 by L36. The cliff sits in the same band as the clean Stage 2 transition (Fig. 4), placing NoOp-induced failure inside Operation Planning: not a Schema-Abstraction error (which would corrupt before or at L22), not an Operand-Binding confusion (which would land at L40), but a corrupted operation plan. Once that plan crystallizes at L36, the answer it commits to is locked in the residual stream and no longer responds to question-span patching. Figure 6b decomposes the same effect in direct mode at the answer-prefix position. The full-residual trace rises across Stage 2 with the attention curve spiking in sync on and before L36, a first hint that attention carries the corruption.

The cliff locates the corruption but does not name the heads that write it. To trace it, we use an *engagement-anchored* per-head Direct Logit Attribution (DLA, illustrated in Fig. 7). The anchor is the moment NoOp's own CoT first commits to the wrong plan: we find the first occurrence in the trace of a *distractor-only* number (a number present in NoOp but absent from the matched Symbolic question), identify the sentence that contains it, and define the *target token* $t^\star$ as the first token of that sentence, i.e. the opening of the sentence that first engages the distractor. The *readout position* r is the slot immediately before $t^\star$, where the model would emit it. The DLA then attributes each head's contribution to $\log P(t^\star)$ at position r that flows through attention to the NoOp clause. For head h at layer L and NoOp clause-token positions $\mathcal{C}$:

$$\mathrm{DLA}_h^{\mathcal{C}} = \sum_{p \in \mathcal{C}} \alpha_h[r,p] \cdot W_U[t^\star]^\top W_O^h V_p^h, \quad (1)$$

where $\alpha_h[r,p]$ is the post-softmax attention weight

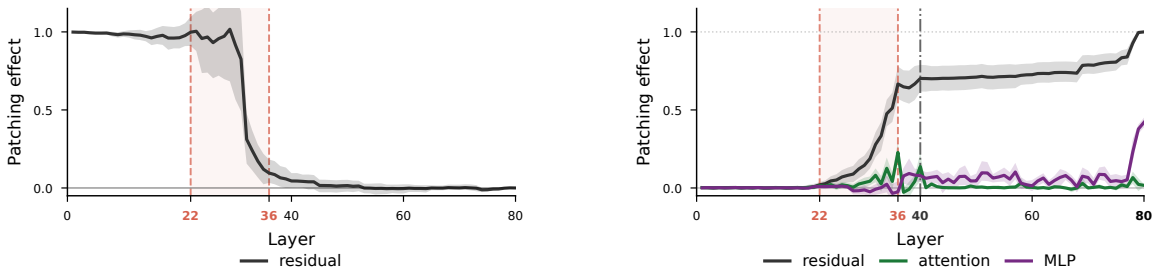

Figure 6: NoOp fragility localizes to Operation Planning. Both panels patch from Filler-correct into NoOp-wrong, a pair differing only in the distractor clause. **(a) Left: CoT-mode question-span patching.** The patching effect holds at 1.00 through L22 and collapses to 0.10 by L36. The cliff falls in the same band as the clean Operation Planning transition (Fig. 4), so what NoOp corrupts is the operation plan rather than an earlier or later stage. **(b) Right: direct-mode decomposition at the answer prefix**, decomposed by component. The full residual rises from near zero at L22 to 0.52 at L36, localizing NoOp fragility to the Operation Planning band. Attention spikes simultaneously with the rise for the full residual, which we dissect further with the DLA.

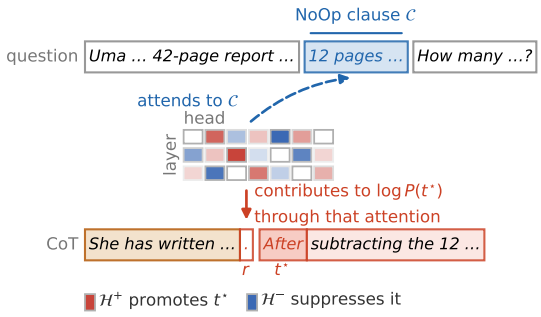

Figure 7: Engagement-anchored DLA. The target token $t^\star$ is the first token of the model-produced CoT that engages the distractor clause $\mathcal{C}$, shown in red. The readout position r is the slot immediately before it. Eq. 1 scores each head at each layer by the amount its write at r adds to $\log P(t^\star)$ through that head's own attention to $\mathcal{C}$. The score asks of each head, reading specifically the NoOp clause, whether it pushes the model toward or away from getting distracted by the clause.

that head h attributes to clause token at position p from readout position r, V_p^h is head h's value at p, W_O^h is the head's slice of the output projection, and $W_U[t^\star]$ is the unembedding row for the per-example target token. Each summand factors into a *routing* term (the attention weight $\alpha_h[r,p]$, i.e. how much r attends to clause token p) and a *writing* term (the OV product $W_U[t^\star]^\top W_O^h V_p^h$, the signed per-unit-attention contribution to $\log P(t^\star)$). Summed over $\mathcal{C}$, the DLA isolates the head's contribution to $t^\star$ that flows specifically through attention to the NoOp clause.

We compute DLA on the NoOp-wrong subset whose CoT actively engages the distractor number ($n = 720$ across 37 templates), and define two head sets within the Operation Planning band $L \in [23, 36]$: the *anti-engagement* set $\mathcal{H}^- = \{(L,h) : \mathrm{DLA}_h^{\mathcal{C},\mathrm{noop}} < -10^{-4}\}$, whose clause attention writes *against* $t^\star$ (suppressing the wrong-plan token), and the *pro-engagement* set $\mathcal{H}^+ = \{(L,h) : \mathrm{DLA}_h^{\mathcal{C},\mathrm{noop}} > +10^{-4}\}$, which writes *for* it. The highest single-layer concentrations of both sets are found at L36, the Operation Planning boundary (App. D, Fig. 11).

We test causal responsibility by scaling each head's output by a constant α throughout the forward pass and the generation rollout, for every head in sets $\mathcal{H}^-$, $\mathcal{H}^+$, and $\mathcal{H} = \mathcal{H}^- \cup \mathcal{H}^+$ respectively, and size-matched random heads drawn from layers *outside* L23–L36. $\alpha = 1$ recovers the unperturbed model, $\alpha = 0$ ablates the head, and $\alpha = 2$ doubles its contribution. Two complementary experiments probe the heads' role in Operation Planning: **(i) Ablation damage on correct examples.** If $\mathcal{H}$ is the operation-planning machinery, ablating it on prompts the model normally solves should break the model. We ablate each head set at $\alpha = 0$ on the correctly-solved Symbolic subset ($n = 4{,}752$, 100 templates). **(ii) Amplification recovery on NoOp-wrong.** If $\mathcal{H}^-$ specifically suppresses the wrong-plan token $t^\star$ but is too weak to flip the answer on its own, amplifying it should rescue some of the lost accuracy. We amplify each head set at $\alpha = 2$ on the NoOp problems where the model answered wrongly and measure the fraction that recover the correct final answer.

Tables 1 and 2 report the outcomes. Ablation collapses previously-correct accuracy by orders of magnitude on Symbolic-correct: $\mathcal{H}^+$ alone drops it to 0.06, $\mathcal{H}^-$ alone to 0.47, and their union to 0.02, all against random-outside controls that retain ≥ 0.98, with damage gaps reaching $\sim$10–20σ. Amplifying $\mathcal{H}^-$ on NoOp-wrong lifts accuracy from 0

Head set	$\lvert\mathcal{H}\rvert$	Symbolic
$\mathcal{H}^-$	134	0.47 ± 0.04
$\mathcal{H}^+$	163	0.06 ± 0.01
$\mathcal{H}$	297	0.02 ± 0.01
Random-135	135	0.98 ± 0.01
Random-297	297	0.98 ± 0.01

Table 1: Answer accuracy on Symbolic-correct after ablating each head set ($\alpha = 0$). Accuracy is 1 before ablation by construction. $\mathcal{H}$ heads collapse accuracy by orders of magnitude relative to baseline.

Head set	$\lvert\mathcal{H}\rvert$	NoOp
$\mathcal{H}^-$	134	$\mathbf{0.51 \pm 0.06}$
$\mathcal{H}^+$	163	0.33 ± 0.06
$\mathcal{H}$	297	0.45 ± 0.07
Random-135	135	0.19 ± 0.05

Table 2: Answer accuracy on NoOp-wrong after amplification ($\alpha = 2$). Accuracy is 0 before amplification by construction. $\mathcal{H}^-$ gives a $+0.32$ gap over random-outside ($\sim 4\sigma$). $\mathcal{H}^+$ gives $+0.14$ ($\sim 2\sigma$).

to 0.51, $\sim 4\sigma$ over the 0.19 random-outside baseline. Amplifying $\mathcal{H}^+$ lifts it to 0.33 ($\sim 2\sigma$), consistent with these heads' Operation Planning role.

Together, the two interventions identify $\mathcal{H}$ as the causal instrument of Operation Planning. $\mathcal{H}^+$ is the dominant subset. Its ablation alone damages clean reasoning 7–8$\times$ more than $\mathcal{H}^-$'s. When faced with NoOp distractors, the anti-engagement signal from $\mathcal{H}^-$ is overwhelmed by $\mathcal{H}^+$'s larger pro-engagement force. However, the DLA should not be read as a verdict on which heads cause the failure because Eq. 1 sums only a head's direct write to $t^\star$ through its attention to the NoOp clause. α-scaling, on the other hand, rescales the whole head over the full generation. Consequently, amplifying $\mathcal{H}^+$ helps as well. Moreover, the blunt 0.51 recovery is not a repair ceiling. A single global scalar on a fixed head set recovers half of the failures, which places the failure at these heads and shows that a training-free lever exists there.

6.1 Classifying the Induced Failures

The scaling experiments show that the heads are necessary for solving the problems. They do not by themselves show which stage breaks when they are absent. We therefore classify the failures induced when the heads are ablated.

On GSM-Symbolic problems the model normally solves correctly, we ablate a head set and read the resulting reasoning traces. We sample 60 failure traces per head set and classify which stage the failures belong to. An Operation Planning failure reads the problem correctly but applies wrong or missing operations. Each trace is labeled independently by two annotators who see only the problem and the model's trace, and not which heads were turned off, so that the labels cannot be steered toward our hypothesis. A trace counts as a Planning failure only when both annotators agree.

Head set	Operation Planning failures
$\mathcal{H}$	**0.91**
Random-297	0.10

Table 3: Fraction of induced failures that are Operation Planning failures. Ablating the planning heads makes the model apply the wrong operations. Ablating size-matched random heads outside the band produces misreadings and arithmetic errors instead.

Table 3 reports the result. Ablating $\mathcal{H}$ produces Operation Planning failures in 0.91 of cases, against 0.10 for size-matched random heads outside the band, whose failures are misreadings and arithmetic slips instead. App. E demonstrates the contrast through an example problem. The stage label is thus validated by what breaks: if L36 implemented some other function, ablating its heads would produce that other failure type.

7 Conclusion

We presented a four-stage pipeline, Schema Abstraction, Operation Planning, Operand Binding, and Computation, that decomposes how LLMs process GSM-style math word problems. Residual cosine dynamics, presence probes, and cross-prompt activation patching converge independently on the same stage boundaries. The framework refines Cheng et al. (2025)'s two-stage abstract-then-compute account into four stages and extends it from synthetic one-step direct-mode prompts to multi-step CoT reasoning. Within this scaffold, NoOp fragility localizes to Operation Planning, associated with a compact set of attention heads whose bidirectional causal role we validated by scaling: ablation collapses correct solutions by orders of magnitude against size-matched random controls, while amplifying the anti-engagement subset alone recovers a substantial fraction of NoOp failures. Our results suggest that distractor-induced failures arise from localized corruption of an intermediate Operation Planning stage rather than a global breakdown of reasoning.

Limitations

Our analysis focuses on math word-problem reasoning and on distractor-induced failures in GSM-style tasks. The transfer results of §5 and App. G are first steps beyond that scope, but they are probe-level replications rather than the full battery of methods applied in the main text, and mapping other tasks' pipelines at that depth, with probes, patching, and causal interventions, remains open. A second direction is repair. The amplification result in §6 identifies the location for repair, and turning it into a targeted, deployable intervention is future work that the localization makes concrete.

Acknowledgments

This project is partially supported by an AI2050 Senior Fellowship, a Schmidt Sciences program; the National Science Foundation (NSF); the National Institute of Food and Agriculture (USDA/NIFA) (2023-67021-39829); the Air Force Office of Scientific Research (AFOSR) (FA9550-23-1-0322)

References

Guillaume Alain and Yoshua Bengio. 2016. Understanding intermediate layers using linear classifier probes. *arXiv preprint arXiv:1610.01644*.

Ziling Cheng, Meng Cao, Leila Pishdad, Yanshuai Cao, and Jackie CK Cheung. 2025. Can llms reason abstractly over math word problems without cot? disentangling abstract formulation from arithmetic computation. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 14306–14333.

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

Alexis Conneau, German Kruszewski, Guillaume Lample, Loïc Barrault, and Marco Baroni. 2018. What you can cram into a single $&!#* vector: Probing sentence embeddings for linguistic properties. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2126–2136, Melbourne, Australia. Association for Computational Linguistics.

Nelson Elhage, Tristan Hume, Catherine Olsson, Nicholas Schiefer, Tom Henighan, Shauna Kravec, Zac Hatfield-Dodds, Robert Lasenby, Dawn Drain, Carol Chen, et al. 2022. Toy models of superposition. *arXiv preprint arXiv:2209.10652*.

Nelson Elhage, Neel Nanda, Catherine Olsson, Tom Henighan, Nicholas Joseph, Ben Mann, et al. 2021. A mathematical framework for transformer circuits. *Transformer Circuits Thread*.

Gemma Team. 2024. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*.

Albert Gong, Kamilė Stankevičiūtė, Chao Wan, Anmol Kabra, Raphael Thesmar, Johann Lee, Julius Klenke, Carla P Gomes, and Kilian Q Weinberger. 2025. Phantomwiki: On-demand datasets for reasoning and retrieval evaluation. *arXiv preprint arXiv:2502.20377*.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Wes Gurnee and Max Tegmark. 2024. Language models represent space and time. In *International Conference on Learning Representations*, volume 2024, pages 2483–2503.

Shou-Tzu Han, Rodrigue Rizk, and KC Santosh. 2026. Fragile reasoning: A mechanistic analysis of llm sensitivity to meaning-preserving perturbations. *arXiv preprint arXiv:2604.01639*.

Mary Hegarty, Richard E Mayer, and Christopher A Monk. 1995. Comprehension of arithmetic word problems: A comparison of successful and unsuccessful problem solvers. *Journal of educational psychology*, 87(1):18–32.

Walter Kintsch and James G Greeno. 1985. Understanding and solving word arithmetic problems. *Psychological review*, 92(1):109–129.

Qintong Li, Leyang Cui, Xueliang Zhao, Lingpeng Kong, and Wei Bi. 2024. Gsm-plus: A comprehensive benchmark for evaluating the robustness of llms as mathematical problem solvers. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2961–2984.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. 2022. Locating and editing factual associations in gpt. *Advances in neural information processing systems*, 35:17359–17372.

Iman Mirzadeh, Keivan Alizadeh-Vahid, Hooman Shahrokhi, Oncel Tuzel, Samy Bengio, and Mehrdad Farajtabar. 2025. Gsm-symbolic: Understanding the limitations of mathematical reasoning in large language models. In *International Conference on Learning Representations*, volume 2025, pages 94743–94765.

Arkil Patel, Satwik Bhattamishra, and Navin Goyal. 2021. Are NLP models really able to solve simple math word problems? In *Proceedings of the 2021*

Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 2080–2094.

Freda Shi, Xinyun Chen, Kanishka Misra, Nathan Scales, David Dohan, Ed Chi, Nathanael Schärli, and Denny Zhou. 2023. Large language models can be easily distracted by irrelevant context. In *International Conference on Machine Learning*, pages 31210–31227. PMLR.

Alessandro Stolfo, Yonatan Belinkov, and Mrinmaya Sachan. 2023. A mechanistic interpretation of arithmetic reasoning in language models using causal mediation analysis. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 7035–7052.

Jesse Vig, Sebastian Gehrmann, Yonatan Belinkov, Sharon Qian, Daniel Nevo, Yaron Singer, and Stuart Shieber. 2020. Investigating gender bias in language models using causal mediation analysis. *Advances in neural information processing systems*, 33:12388–12401.

Kevin Wang, Alexandre Variengien, Arthur Conmy, Buck Shlegeris, and Jacob Steinhardt. 2022. Interpretability in the wild: a circuit for indirect object identification in gpt-2 small. *arXiv preprint arXiv:2211.00593*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc V Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 35, pages 24824–24837. Curran Associates, Inc.

An Yang et al. 2024. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*.

Minglai Yang, Ethan Huang, Liang Zhang, Mihai Surdeanu, William Yang Wang, and Liangming Pan. 2025. How is llm reasoning distracted by irrelevant context? an analysis using a controlled benchmark. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 13329–13347.

A Implementation Details

Prompt template. All experiments use zero-shot prompting with the model's native chat template. The system prompt is fixed across conditions:

```
You are a chatbot who is capable of
performing the arithmetic problems.
```

The user message concatenates the question with a mode-specific instruction.
For direct prompting:

```
{question}
Please answer the question directly
WITHOUT showing the reasoning process.
At the end, you MUST write the answer
as an integer after '####', without the
equation or units.
```

For chain-of-thought (CoT) prompting:

```
{question}
Let's think step by step. At the end,
you MUST write the answer as an integer
after '####', without the equation or
units.
```

In CoT experiments that inject a cached reasoning trace (e.g., the Stage 2 single-token patching setup in §5.1), the cached reasoning text, anything the model generates before the answer prefix ####, is inserted after the assistant header.

GSM-NoOp construction. For each GSM-Symbolic template, Claude Opus 4.7 is prompted to propose three candidate NoOp clauses that read as natural continuations of the problem narrative. One is manually selected under two criteria: the clause must be inconsequential to the gold computation, and it must be plausible enough to function as a distractor. The selected clause is then programmatically instantiated across the template's entities and numbers (e.g. substituting matching characters, units, and quantities), producing one NoOp instance per Symbolic instance with one-to-one alignment. We sample a random subset of the resulting NoOp questions and manually verify that each reads naturally end-to-end, the inserted clause does not change the gold answer, and the clause is not trivially ignorable.

GSM-Filler construction. GSM-Filler is built in parallel to NoOp using the same template-by-template pipeline. For each NoOp clause we author a filler clause that reuses the same numeric token(s) that appear in the matched NoOp clause but embeds them in a context with no computational implication (e.g., a descriptive or stylistic mention). Holding the surface number constant isolates the contribution of the surrounding semantics: any accuracy gap between GSM-Filler and GSM-NoOp on length-matched, digit-matched prompts attributes to how the model interprets the clause, not to which numeric tokens are present. Filler is then length-matched to NoOp at the prompt level by adding filler words like "indeed", "as noted" to the shorter side.

GSM-Filler-DF construction. GSM-Filler-DF is a stricter digit-free baseline built with the same pipeline as GSM-Filler but with all number tokens stripped from the inserted clause and replaced by words like "some", i.e., words when substituting for a number token still make the question reads naturally. It rules out residual "soft-NoOp" effects that the digit-matched GSM-Filler may inherit from the surface number itself.

padded-Symbolic construction. padded-Symbolic uses the same authoring pipeline as GSM-Filler-DF, with the filler clause length-matched to the corresponding GSM-P1 clause rather than to the NoOp clause. It serves as a length-controlled substitute for GSM-Symbolic in the Stage 2 patching experiments (§5.1).

Dataset accuracy gradient. Table 4 reports accuracy on all six datasets, including padded-Symbolic and the digit-free GSM-Filler-DF. Padded-Symbolic accuracy matches GSM-Symbolic to within -0.004 direct / -0.016 CoT, supporting its use as a length-controlled substitute in §5.1. The digit-bearing GSM-Filler used in the main text shows a small soft-NoOp drop (-0.067 direct, -0.052 CoT) that is $\sim 2\times$ smaller than GSM-NoOp's in CoT and $\sim 1.5\times$ smaller in direct; the digit-free GSM-Filler-DF is statistically indistinguishable from baseline.

B Prompting Mode Robustness

§5 uses the answer-position residual under direct prompting as the canonical readout for Figure 2: direct prompting forces all internal computation through a single position with no externalized trace, cleanly exposing the staged structure. We verify here that the same signature appears under Chain-of-Thought (CoT) prompting and at a pre-reasoning readout where the model has not yet emitted any trace.

Dataset	n	Direct Acc.	Direct Δ	CoT Acc.	CoT Δ
GSM-Symbolic	5,000	0.241	—	0.950	—
padded-Symbolic	4,852	0.237	-0.004	0.934	-0.016
GSM-P1	5,000	0.068	-0.173	0.912	-0.038
GSM-Filler-DF	5,000	0.237	-0.004	0.941	-0.010
GSM-Filler	5,000	0.174	-0.067	0.898	-0.052
GSM-NoOp	5,000	0.140	-0.101	0.780	-0.171

Table 4: Llama-3.3-70B accuracy across all datasets (50 templates $\times$ 100 instantiations each), including padded-Symbolic (the length-matched substitute used in the Stage 2 patching experiments) and the digit-free GSM-Filler-DF row. Δ is relative to GSM-Symbolic. Padded-Symbolic accuracy matches GSM-Symbolic to within -0.004 direct and -0.016 CoT, supporting its use as a length-controlled substitute in §5.1. The digit-bearing GSM-Filler used in the main text yields a small but consistent "soft-NoOp" drop, strictly larger than the digit-free GSM-Filler-DF's null effect and strictly smaller than GSM-NoOp's drop.

CoT-mode replication. Figure 8 repeats Fig. 2 at the answer-prefix token *following* the cached CoT trace. The L22 signature replicates in both panels: cross-template cosine drops, entity decodability collapses to chance, and operator features rise. The Stage-3 and Stage-4 readouts shift as expected: the L40 number-token jump is smaller (operands are already attended to via the CoT trace) and the final-answer probe saturates early (the CoT trace text contains the answer already). These shifts are downstream consequences of attention into CoT trace tokens, not changes to the underlying stage structure.

Pre-reasoning matched-position control. A complementary control reads at the assistant-header position of a CoT-framed prompt *before any reasoning has been generated*: same prompt template as the CoT readout, but with no CoT trace tokens given as input. Figure 9 shows the L22 entity-drop / operator-rise dissociation persists at this readout, attenuated but directionally identical (entity $0.78 \rightarrow 0.68$ here vs. $0.78 \rightarrow 0.53$ in direct mode). After dropping at L22, entity probe rises again at L40 since the entity names are relevant for writing the CoT trace. Number token and final answer accuracies stay close to chance throughout as the operand binding and computation happens throughout the CoT trace, instead of at the pre-CoT position. The three positions thus exhibit a coherent ordering (direct $\approx$ CoT $>$ pre-reasoning), consistent with L22 circuitry engaging most strongly at decision-point tokens where the model is committing to an output.

C Filler-DF Replications

The main text presents the Stage 2 break test on the digit-bearing filler control (GSM-Filler). The length-matched digit-free variant (GSM-Filler-DF) has near-zero behavioral effect when compared with GSM-Symbolic (Table 4), and serves as a different control that checks if the extra numeric token in the NoOp has any effect on model fragility. Figure 10 replicates both panels of the main-text Stage 2 break test (Fig. 6) on the GSM-Filler-DF control ($n = 305$ across 19 templates).

Question-span cliff (panel a). The L22–L36 cliff in the patching effect on $\log P(\text{final answer})$ replicates the digit-bearing variant in the main text, confirming the cliff is not specific to the digit-bearing filler.

Sublayer decomposition (panel b). The answer-prefix sublayer decomposition (full residual / attention / MLP) for direct mode also replicates: the full-residual rise is present in the L22–L36 band.

D Engagement-Anchored DLA: Full-Layer Scan

The main-text engagement-anchored DLA analysis is restricted to L22–L36, the operation-planning band identified independently by the patching results in §5.1. To confirm that this band-restriction is not cherry-picking, we report results on the same DLA computation across all 80 layers of Llama-3.3-70B. Figure 11 shows the per-layer count of heads with $|\text{DLA}| > 10^{-4}$, split by anti- and pro-engagement direction. Three observations support the main-text scope:

Pre-L22 negative control. Layers L0–L22 are essentially silent: only 5.5% of heads exceed $|\text{DLA}| > 10^{-4}$, with maximum $|\text{DLA}|$ below 0.0026 at any head. The engagement-anchored measurement does not produce spurious signal in pre-planning layers.

Density peaks inside the Operation Planning band. Inside the band, 27.2% of heads exceed the active-head threshold, with mixed-sign structure (anti- and pro-####) consistent with the main-text $\mathcal{H}^-/\mathcal{H}^+$ decomposition. This is the highest active-head density in the layer sweep.

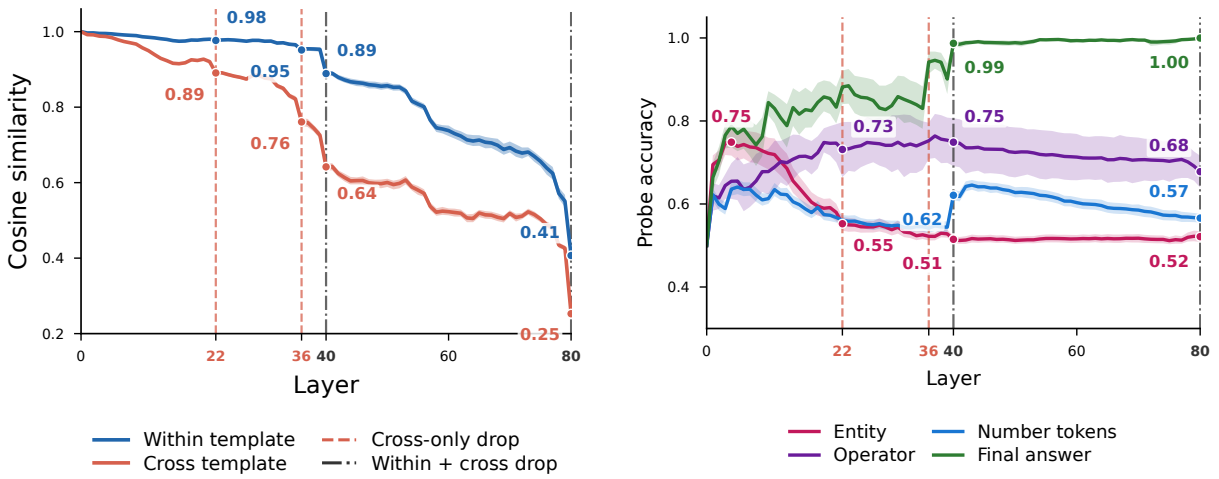

Figure 8: CoT-mode replication of Fig. 2. **(a) Left: residual cosine similarity. (b) Right: token-presence probes.** The readout sits at the answer-prefix token *following* the cached chain-of-thought trace, so attention at the readout position can attend over the trace tokens. The L22 signature replicates in both panels: cross-template cosine drops, entity decodability collapses to chance, operator features rise. The L40 number-token jump is smaller than in direct mode (operands are attended-to via the trace) and the final-answer probe saturates early (the trace contains the answer string). Both are expected consequences of attention into the externalized reasoning, not contradictions of the staged internal structure exposed in Fig. 2.

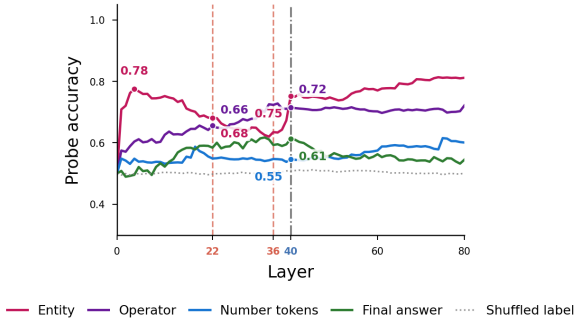

Figure 9: Linear presence probing at the assistant-header position of a CoT-framed prompt before any reasoning has been generated. The entity-drop / operator-rise dissociation at L22 that defines Stage 1 in Fig. 2 reappears here. The drop is smaller than at the decision-point readouts in Fig. 2 and Fig. 8, consistent with stage circuitry engaging most strongly at output-commitment positions. Stage 3 (number-token) and Stage 4 (final-answer) signatures are absent at this position, consistent with binding and computation being done during CoT generation.

Post-L36 activity is real but downstream. After L36, 12.9% of heads remain above threshold, and several layers (L57, L65, L74, L78) carry isolated large-magnitude heads. We interpret these as the operand-binding attention mover (§5.2) and computation machinery downstream of, and dependent on, operation planning.

E Failure Classification: A Worked Example

The following problem was solved correctly by the unablated model and appears in both ablation conditions of §6.1, so the two failure types can be compared on a single item. *A fog bank takes* 398 *minutes to cover every* 3 *miles of a* 42*-mile city. How long will it take to cover the whole city?* (gold answer 5,572).

With size-matched random heads ablated, the model forms the correct plan and slips only in the arithmetic: it computes the per-mile rate $398 \div 3$, multiplies by 42, and answers 5570 instead of 5572. Comprehension and operation selection are intact, only the final value is wrong, which is not an Operation Planning failure.

With the $\mathcal{H}$ heads ablated, the model reads the same quantities and still computes the per-mile rate correctly, then fails to form the multiplication. It asserts that covering the whole city takes the same 398 minutes as covering 3 miles, and answers 398, a quantity copied from the question. Comprehension is intact and no arithmetic is botched. The operation plan is never completed.

The blind annotation sheets and the adjudicated per-trace labels for all sampled failures are included in the code release.

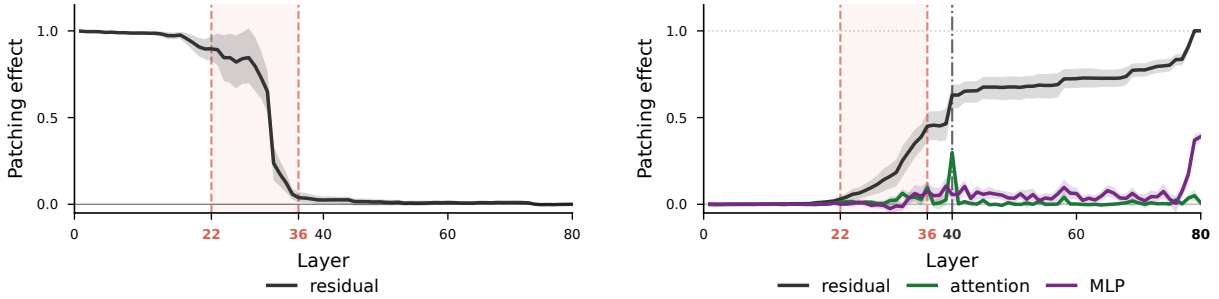

Figure 10: NoOp diagnosis of Llama-70B on the digit-free Filler-DF control, companion to Fig. 6. **(a) Left: question-span CoT-mode patching.** From Filler-DF-correct to NoOp-wrong. The L22–L36 cliff replicates. **(b) Right: direct-mode prompt-end decomposition.** Full residual (blue), attention output only (green), MLP output only (red). The full-residual rise replicates.

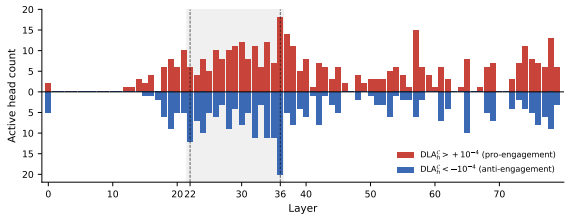

Figure 11: Per-layer count of heads with $|\mathrm{DLA}_h^{\mathcal{C},\mathrm{noop}}| > 10^{-4}$ across all 80 layers of Llama-3.3-70B. Bars above the zero-line: heads writing toward $t^\star$ (pro-engagement, $\mathcal{H}^+$). Bars below: heads writing against $t^\star$ (anti-engagement, $\mathcal{H}^-$). Gray shading marks the Operation Planning band. Activity ramps from near-zero pre-L22, reaches its highest density inside the band (27.2% of heads active vs. 5.5% below L22 and 12.9% above L36), and peaks at L36 with 20 anti- + 18 pro-engagement heads. Isolated post-L36 layers (L57, L65, L74, L78) carry secondary activity, attributable to operand-binding and computation/readout machinery downstream of, and dependent on, operation planning.

F Generalization to Other Models

We replicate the Stage 1 signatures from Section 5 on two additional open-source models: Gemma-2-9B-it (42 layers) and Qwen-2.5-14B-Instruct (48 layers). The four-stage residual-stream structure reproduces in both.

		Direct		CoT	
Model	Dataset	Acc.	Δ	Acc.	Δ
Llama-3.3-70B	Symbolic	0.241	—	0.950	—
	P1	0.068	−0.173	0.912	−0.038
	Filler-DF	0.237	−0.004	0.941	−0.010
	Filler	0.174	−0.067	0.898	−0.052
	NoOp	0.140	−0.101	0.780	−0.171
Gemma-2-9B	Symbolic	0.125	—	0.859	—
	P1	0.036	−0.089	0.761	−0.098
	Filler-DF	0.120	−0.005	0.856	−0.003
	Filler	0.079	−0.046	0.783	−0.076
	NoOp	0.062	−0.063	0.556	−0.303
Qwen-2.5-14B	Symbolic	0.162	—	0.918	—
	P1	0.059	−0.103	0.878	−0.040
	Filler-DF	0.159	−0.003	0.913	−0.005
	Filler	0.127	−0.035	0.897	−0.021
	NoOp	0.083	−0.079	0.762	−0.156

Table 5: Four-dataset accuracies for the three models (companion to Table 4). Δ is relative to GSM-Symbolic for each model. The Filler-DF-null / Filler-small-drop / P1-drop / NoOp-largest-drop ordering replicates across all three model families, with digit-bearing Filler producing a small "soft-NoOp" drop strictly between digit-free Filler-DF and full NoOp.

Dataset accuracies. Table 5 reports the four-dataset accuracies for Gemma-2-9B and Qwen-2.5-14B alongside Llama-3.3-70B (companion to Table 4). The qualitative pattern from Llama replicates in both: Filler-DF is nearly indistinguishable from Symbolic ($|\Delta| \leq 0.005$), P1 incurs a moderate drop, and NoOp is the largest behavioral hit.

	Llama 3.3-70B (80L)	Gemma 2-9B (42L)	Qwen 2.5-14B (48L)
L22	22 (0.28)	11 (0.26)	23 (0.48)
L36	36 (0.45)	27 (0.64)	31 (0.65)
L40	40 (0.50)	32 (0.76)	37 (0.77)
L80	80 (1.00)	42 (1.00)	48 (1.00)

Table 6: Stage-boundary layers and their depth ratios (layer / total layers) for each model.

Template similarity. Figure 12 shows residual cosine similarity dynamics for each model in direct mode with model-specific stage boundaries marked. Both Gemma-9B and Qwen-14B reproduce the canonical four-event sequence from Fig. 2a: a first cross-only drop (Schema Abstraction), a second cross-only drop (Operation Planning), a joint within+cross drop (Operand Binding), and a terminal-layer readout (Computation).

Stage-boundary depth across models. Table 6 reports each model's stage-boundary layers as fractions of total depth. Stage 1 (Schema Abstraction) completes at roughly the same proportional depth on Llama and Gemma (0.26–0.28) but considerably deeper on Qwen (0.48), so Qwen spends nearly half of its layers stripping surface narrative before the operation plan begins to form. Stage 2 (Operation Planning) and Stage 3 (Operand Binding) both shift later as a fraction of depth on Gemma and Qwen (~ 0.65 and ~ 0.77) than on Llama (0.45 and 0.50), leaving Llama with the longest tail of Stage 4 (Computation) layers in proportional terms (50% of the model vs. 23–24% on the smaller models).

Presence probes. Figure 13 replicates Fig. 2b setup on Gemma-9B and Qwen-14B with each model's own stage boundaries marked. All three probed stage signatures reproduce: **(Stage 1)** entity decodability falls from a pre-cliff peak ($\sim$ 0.80–0.91) to chance ($\sim$ 0.51–0.55) by the cross-only boundary identified independently in Fig. 12; **(Stage 3)** number-token decodability jumps from chance to $\sim$ 0.78–0.81 at the within-cross boundary; **(Stage 4)** final-answer decodability rises from chance to $\sim$ 0.78–0.86 approaching the terminal layer.

Stage 2 patching. Figures 14 and 15 replicate the main-text Stage 2 panels (Fig. 4) on Gemma-2-9B and Qwen-2.5-14B, with each model's own stage boundaries marked. The question-span patching effect on $\log P(\text{final answer})$ rides near 1 before each model's Schema-Abstraction boundary (Gemma L11, Qwen L23) and collapses by the Operation-Planning boundary (Gemma L27, Qwen L31). For the single-token patch at the end of the injected CoT, the full-residual commit-readiness trace rises across the same band, with attention as the leading sublayer component, matching the Llama Stage 2 signature. Gemma has $n = 864$ examples across 48 templates. Qwen has $n = 1{,}044$ across 36 templates.

Stage 3 patching. Figures 16 and 17 replicate the main-text Stage 3 panels (Fig. 5) on the two additional models, with each model's own stage boundaries marked. On these two models the CoT-mode patch is restricted to the numeric-value-token positions rather than applied to the whole question span as in Fig. 5a. Within a template the two are equivalent, since the source and target questions differ only in their operand values, and the value-token form additionally holds every operand at the same absolute position in both prompts. The CoT-mode within-template value-token cliff sits at the Operation-Planning $\rightarrow$ Operand-Binding transition (Gemma L27$\rightarrow$L32, Qwen L31$\rightarrow$L37). Gemma sample sizes are $n = 926$ pairs across 87 templates (CoT-mode) and $n = 478$ across 47 (direct-mode). Qwen has $n = 711$ across 81 templates (CoT-mode) and $n = 617$ across 49 (direct-mode).

NoOp fragility. Figure 18 replicates the main-text NoOp diagnosis (Fig. 6a) on the Filler control for both models. The CoT-mode question-span cliff sits inside each model's Schema-Abstraction to Operation-Planning band (Gemma L11–L27, Qwen L23–L31). The per-layer sweep retains 738 pairs across 44 templates on Gemma and 140 pairs across 43 templates on Qwen.

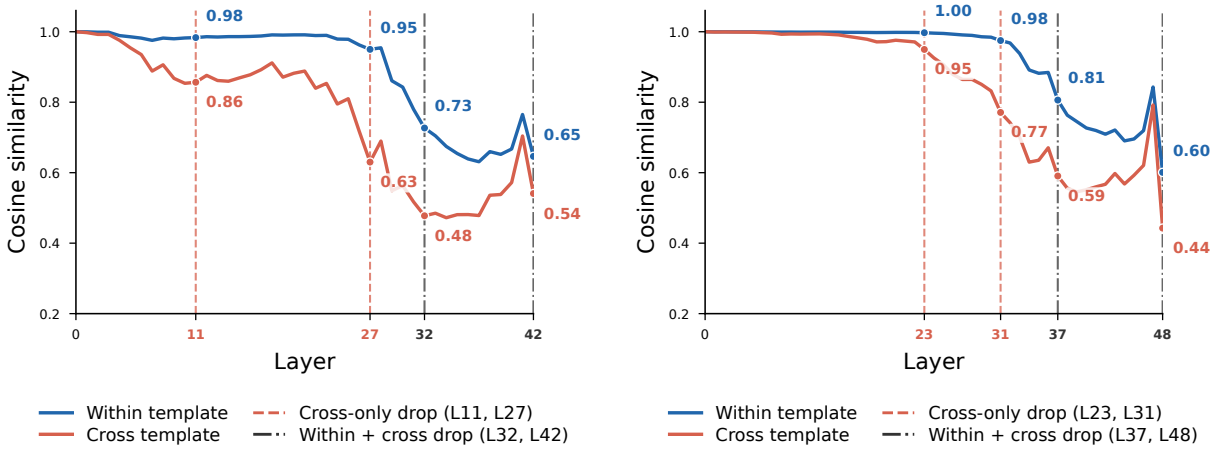

Figure 12: Direct-mode residual cosine similarity dynamics for two additional models (companion to Fig. 2a). **(a) Left: Gemma-2-9B (42 layers). (b) Right: Qwen-2.5-14B (48 layers).** Each panel marks the model's own stage boundaries derived from the same automatic criterion used for Llama. Both models reproduce the four-event signature.

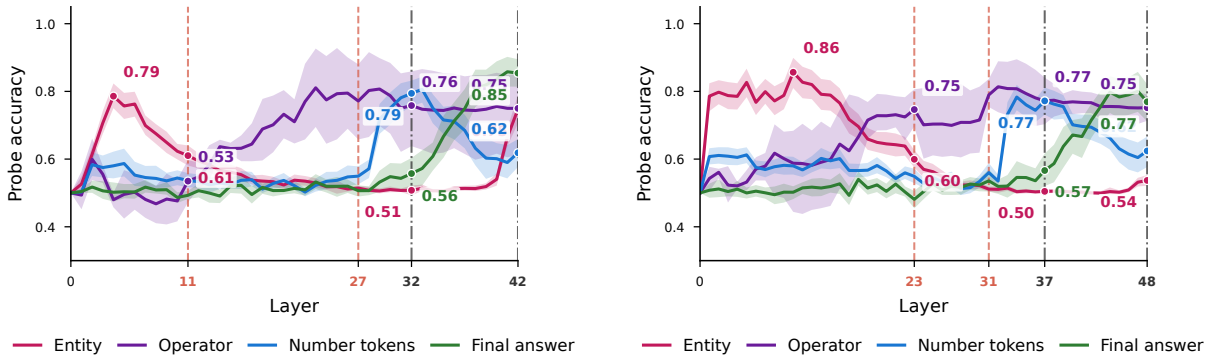

Figure 13: Linear presence-probe accuracies for the two additional models, companion to Fig. 2b. **(a) Left: Gemma-2-9B (42 layers). (b) Right: Qwen-2.5-14B (48 layers).** Each panel marks the model's own stage boundaries. Entity decodability collapses at Stage 1 boundary, number-token decodability jumps at Stage 3 boundary, and final-answer accuracy rises from Stage 3 boundary through the end, matching the Llama signature and the boundaries identified independently from cosine dynamics in Fig. 12.

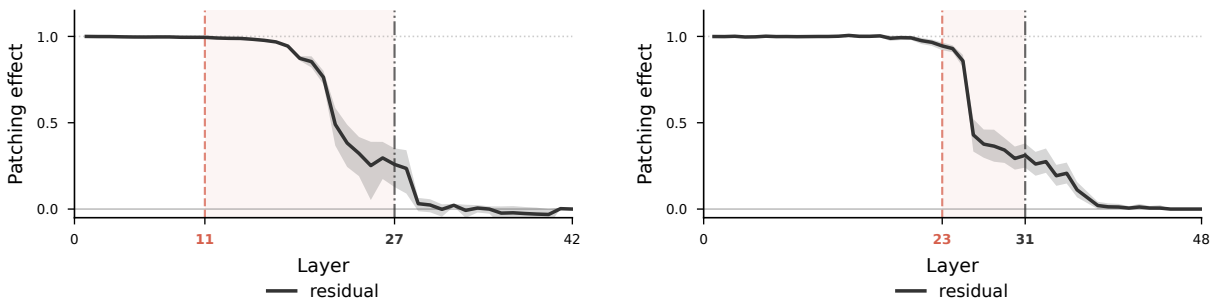

Figure 14: Stage 2 question-span patching (companion to Fig. 4a) on the robustness models. **(a) Left: Gemma-2-9B. (b) Right: Qwen-2.5-14B.** Padded-Symbolic's question-span residual is patched into P1; we report the normalized patching effect on $\log P(\text{final answer})$. The trace rides near 1 before each model's Schema Abstraction boundary and collapses by Operation Planning boundary, matching the Llama pattern.

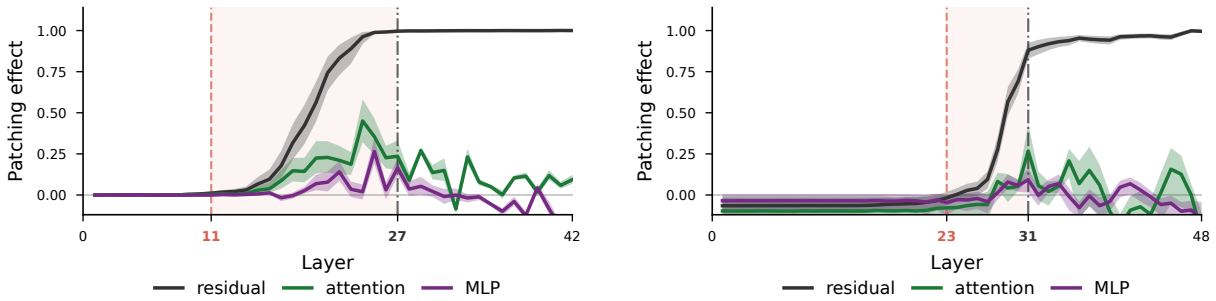

Figure 15: Stage 2 commit-readiness at the end of an injected GSM-Symbolic CoT (companion to Fig. 4b), decomposed into full residual / attention / MLP. **(a) Left: Gemma-2-9B. (b) Right: Qwen-2.5-14B.** The full-residual trace rises across each model's Operation Planning band.

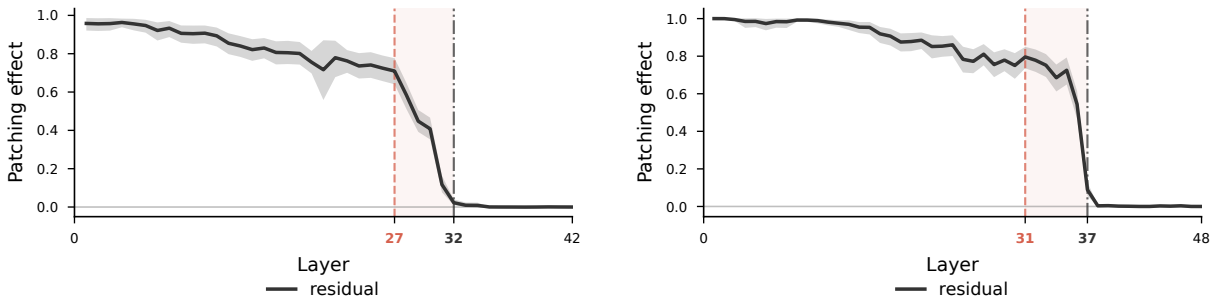

Figure 16: Stage 3 within-template value-token patching (companion to Fig. 5a) on the robustness models. **(a) Left: Gemma-2-9B. (b) Right: Qwen-2.5-14B.** Operand Binding cliff replicates inside each model's band.

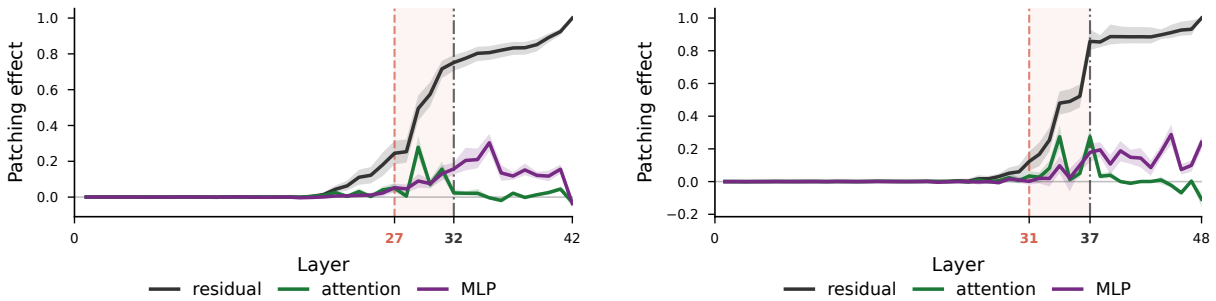

Figure 17: Stage 3 direct-mode patching at the answer prefix (companion to Fig. 5b), decomposed into full residual / attention / MLP. **(a) Left: Gemma-2-9B. (b) Right: Qwen-2.5-14B.** The residual curve raises and attention-output trace spikes inside each model's Operand Binding band.

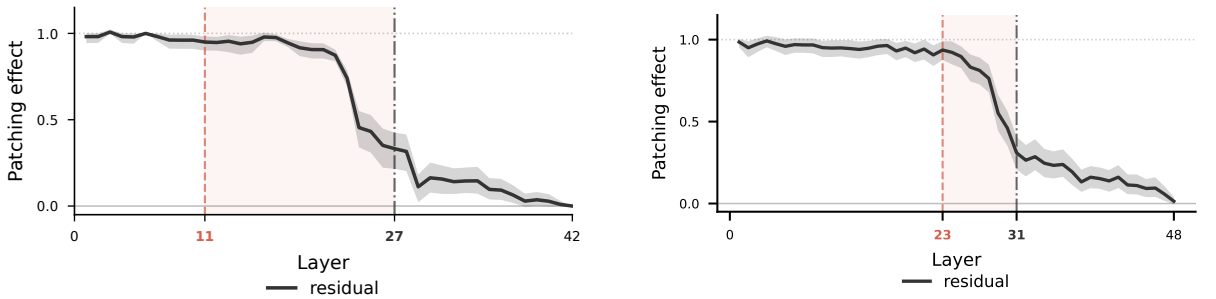

Figure 18: NoOp fragility, CoT-mode question-span patching, Filler-correct → NoOp-wrong (companion to Fig. 6a). **(a) Left: Gemma-2-9B. (b) Right: Qwen-2.5-14B.** The Operation Planning stage cliff replicates on both models.

Head-scaling validation. Tables 7 and 8 replicate the Llama scaling experiments (Tables 1, 2) on Gemma-2-9B and Qwen-2.5-14B. Each model's $\mathcal{H}^-$ and $\mathcal{H}^+$ are identified by the same engagement-anchored DLA procedure, filtered to its own Operation Planning band. Random-outside controls are sampled from layers outside that band and size-matched to the corresponding head set. Recovery aggregates restrict to templates with ≥ 5 NoOp-wrong pairs (template-cluster bootstrap is dominated by per-template variance, and the natural NoOp-wrong pool is heavily long-tailed: a sizeable fraction of templates produce only one such pair). **Ablation replicates on both models**: one head subset collapses Symbolic-correct accuracy by orders of magnitude relative to its size-matched random control ($\mathcal{H}^+$ for Gemma, $\mathcal{H}^-$ for Qwen). **Amplification recovery also replicates**: both head subsets recover NoOp-wrong examples above the random-outside baseline on both models, with Gemma's $\mathcal{H}^+$ showing the largest gap ($+0.15$) and Qwen's gaps smaller. Which subset acts as the dominant damage / rescue channel is therefore model-specific, but the existence of a compact Stage-2 head set whose ablation and amplification both move accuracy preferentially over random-outside controls is preserved across all three models.

G Generalization to Other Datasets and Tasks

The main text establishes the four-stage pipeline on GSM-Symbolic and its controlled variants. This appendix reports two transfer experiments, one to a different benchmark of math word problems and one to a non-math reasoning task. Both use the probe pipeline of §5 unchanged: the same answer-prefix readout, the same balanced-accuracy metric against an empirical shuffled-label baseline, and no per-benchmark tuning of any kind. Layer indices refer to Llama-3.3-70B throughout.

Cross-benchmark replication on SVAMP. SVAMP (Patel et al., 2021) is a benchmark of elementary math word problems built to defeat surface heuristics, with problem bodies that differ in structure from GSM8K's. We probe the answer-prefix residual on the subset the model solves correctly in direct mode ($n = 861$), using seed-disjoint cross-validation so that problems sharing a body never straddle the train/test split. Entity tokens are split into decorative and operand-attached by a mechanical attachment criterion rather than by hand. Two signatures are read on operand-resampled variants of the same problems ($n = 3{,}282$), which give the within-problem contrasts that GSM-Symbolic templates provide natively: operand-number decodability is measured against other instantiations of the same problem, and answer decodability against variants of the same problem with different answers, so that neither can be carried by narrative or topic.

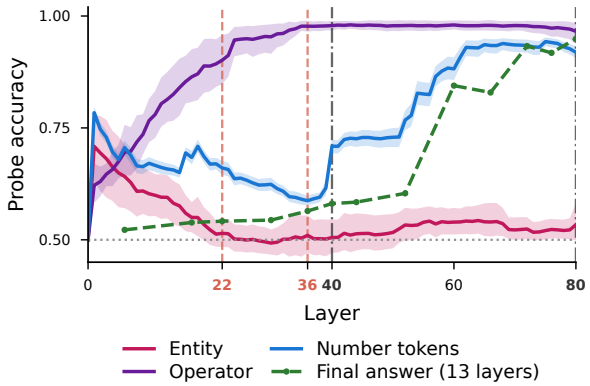

Figure 19: All four stage signatures reproduce on SVAMP at the layers found on GSM-Symbolic, with the same probe pipeline and no per-benchmark tuning (compare Fig. 2b). Decorative-entity decodability falls from 0.71 to 0.51 at L22. Operator decodability is 0.90 at L22 and 0.98 from L36 onward. Operand numbers step to 0.71 at L40. The final answer rises from 0.58 at L40 to 0.95 at L80. The entity and operator curves are read on SVAMP itself. The operand-number and final-answer curves are read on the operand-resampled variants, so both are within-problem contrasts: negatives are other variants of the same problem. The final-answer probe comes from a separate diagnostic evaluated at 13 sampled layers, which is what the markers indicate.

Figure 19 reports the comparison. Decorative-entity decodability peaks before L22 and falls to chance at L22, operator decodability is high by L22 and stays elevated while the plan forms and operands bind, operand numbers hold a plateau through L39 and step up at L40, and answer decodability rises late and saturates at L80. The layer indices at which the four transitions occur are the ones identified on GSM-Symbolic, not re-estimated on SVAMP.

Beyond math: a proof of concept on PhantomWiki. PhantomWiki (Gong et al., 2025) generates fictional biographical corpora together with multi-hop kinship questions over them. The task computes with different tokens than math does: the operations are relation words (*mother*, *husband*,

Head set	$\lvert\mathcal{H}\rvert$	Gemma	Qwen
$\mathcal{H}^-$	74 / 122	0.27 ± 0.06	$\mathbf{0.00 \pm 0.00}$
$\mathcal{H}^+$	134 / 118	$\mathbf{0.00 \pm 0.01}$	0.23 ± 0.07
Random-outside	134 / 118	0.82 ± 0.05	0.91 ± 0.03

Table 7: Answer accuracy on Symbolic-correct after ablating each head set ($\alpha = 0$); companion to Table 1 on Llama. Random-outside is size-matched to $|\mathcal{H}^+|$ for each model. Bold marks the subset whose ablation collapses accuracy to near-zero.

Head set	$\lvert\mathcal{H}\rvert$	Gemma	Qwen
$\mathcal{H}^-$	74 / 122	0.31 ± 0.08	$\mathbf{0.26 \pm 0.09}$
$\mathcal{H}^+$	134 / 118	$\mathbf{0.37 \pm 0.10}$	0.24 ± 0.09
Random-outside	74 / 122	0.22 ± 0.07	0.21 ± 0.08

Table 8: Answer accuracy on NoOp-wrong runs after amplification ($\alpha = 2$); companion to Table 2 on Llama. Random-outside is size-matched to $|\mathcal{H}^-|$ for each model. Aggregates restricted to templates with ≥ 5 NoOp-wrong pairs to avoid single-pair noise.

and so on) and the values passed from one reasoning step to the next are person names. We generate universes and probe the answer-prefix residual on correctly answered questions ($n = 4{,}672$), with universe-disjoint cross-validation so that no universe appears in both train and test.

Three readouts are informative. *Relation words* are decodable at 0.98 by L22 and remain at 0.96 at L40, the same profile operators show in math: the operation vocabulary is extracted early and retained while the rest of the pipeline runs. *Chain membership*, whether a given name lies on the reasoning chain, is measured under a matched contrast in which positives are chain intermediates and negatives are uninvolved names drawn from the same article, matched on presence, adjacency, and difficulty, so that only chain membership differs. Under that contrast the probe sits at chance through L38 (0.50–0.53), rises sharply to 0.59 at L39, and slowly reaches 0.66 by L80. The *answer entity* does not saturate to 1.00 until L80.

The onset of both name signatures at L39 and their completion at L80 sit at the boundaries of the math pipeline, even though the task is retrieval rather than arithmetic. The stage contents are task-shaped and the stage structure recurs: kinship questions have no decorative narrative to strip, and because each hop of a retrieval chain fetches and resolves at once, Operand Binding and Computation are not separated as cleanly as in math but compress into a single resolution cascade. We report this as a proof of concept rather than a full replication.