Knowing Without Saying: How Contextual Evidence Survives but Fails to Surface in Transformers

Ruochen Jin **Tianyu Pang** **Chenxi Lin** **Lei Hsiung**
Jun Jie Ou Yang **Tao Wen** **Yujun Yan** **Yaoqing Yang**
Department of Computer Science, Dartmouth College
Hanover, NH 03755
ruochen.jin.gr@dartmouth.edu yaoqing.yang@dartmouth.edu

Abstract

Large language models frequently ignore in-context evidence that contradicts their parametric priors, producing answers that reflect default beliefs rather than the supplied passage, a phenomenon termed *knowledge conflict*. This failure is commonly attributed to insufficient encoding of the contextual signal. We challenge this explanation. Through systematic, layer-by-layer analysis in a controlled multi-hop question-answering setting, we show that models can favor the context-supported answer in their intermediate representations, yet suppress it in the final layers before generation. We term this *late-layer reversal*: the representational support for the evidence answer rises through the middle layers, then collapses in the final $\sim$40% of the network. To explain this phenomenon, we identify a *Dual-Track* computational structure within the transformer's hidden state. A *knowledge direction*, recovered by a learned linear probe, makes answer-related information linearly readable; a nearly orthogonal *output direction*, aligned with the unembedding geometry, determines the model's prediction. Interventions along the knowledge direction have much weaker effects than those along the output direction: targeted suppression of prior-restoring computation makes the model prefer the contextual answer over the prior in 96.9% of collapse cases. Our findings across nine model–dataset combinations show that knowledge conflict failures can arise after contextual evidence has entered the model's computation, when the generation pathway restores the prior answer.[1]

1 Introduction

Large language models store substantial factual knowledge in their parameters [Petroni et al., 2019, Roberts et al., 2020] and use that knowledge during autoregressive generation. At the same time, many practical uses of these models require them to condition on evidence supplied at inference time, especially in high-stakes settings where the prompt may contain newer, more specific, or task-relevant information [Singhal et al., 2023, Nori et al., 2023]. This raises a basic question for trustworthy language systems: when the context and the model's parametric default disagree, does the model fail because it never represents the contextual evidence, or because it represents the evidence but does not use it to generate the answer?

This question appears directly in *knowledge conflict* settings [Longpre et al., 2021, Xie et al., 2024, Wu et al., 2024]. Here, a prompt contains evidence for one answer while the model's parametric knowledge favors another. For example, consider the comparison question: "Which film has the younger director, *Danger: Diabolik* or *Charge It To Me*?" In the original evidence, the answer is *Danger: Diabolik*; we call this the *prior answer* a, because it matches both the factual answer and the model's default response. We then construct a counterfactual context s^- by editing the relevant

[1] Code is available at https://github.com/kyrie-23/ssc.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

birth years so that the passage instead supports *Charge It To Me*; we call this the *evidence answer* b. If the model still outputs a under s^-, the usual behavioral description is that it ignored the passage. Behavior alone, however, cannot tell us whether the evidence was never encoded or whether it was encoded and later suppressed.

The difference matters. If knowledge conflict is an encoding failure, then the natural solution is to improve how models read and store contextual information. If it is instead a readout failure, then the model may already contain the right evidence internally, while the generation pathway follows a different signal. End-to-end accuracy conflates these cases. We therefore study knowledge conflict inside the network, tracing how the competition between a and b evolves across layers.

Multi-hop question answering gives us the required control. Each question requires combining evidence across passages, and we can edit the supporting chain so that the contextual evidence points to a known target answer b while the model's prior favors a. We keep only examples that the model answers correctly under the original supportive context s^+, then evaluate the same question under the counterfactual context s^-. This construction lets us separate failures of task competence from failures of contextual updating; Appendix A.3 gives a full worked example.

Our central finding is that collapse to the prior is a *late-layer* event. Across three instruction-tuned models and three multi-hop QA datasets, models that eventually follow the counterfactual evidence and models that eventually revert to their prior look similar through much of the forward pass. Only in the final $\sim$40% of layers do their trajectories sharply separate: successful cases amplify the evidence answer b, while collapse cases suppress b and restore the prior answer a. We call this pattern *late-layer reversal*. We also test two further benchmarks, long-form answers, and a 27B model in Section 8.

This reversal exposes a limitation in the usual single-track view of intermediate representations. The logit lens [Nostalgebraist, 2020, Sakarvadia et al., 2023] projects a hidden state through the model's output unembedding, implicitly treating the direction used for final token prediction as the direction along which answer information should be read at every layer [Dar et al., 2023]. But our probes reveal a different geometry. The model's eventual answer preference is linearly predictable from hidden states along a learned *knowledge direction* $\mathbf{V}$, while its output preference is determined by a nearly orthogonal *output direction* $\mathbf{D}$ induced by the unembedding contrast between b and a.

We call this organization **Dual-Track Computation**. The two tracks occupy the same hidden state but play different causal roles: $\mathbf{V}$ carries linearly readable answer information, whereas $\mathbf{D}$ determines what the model says. The separation is not merely geometric. In our main intervention, $\mathbf{V}$ produces no answer flips, while $\mathbf{D}$ reliably moves the model toward the evidence answer. Thus, in the collapse cases we study, the direction that predicts the final answer preference differs from the direction that most effectively changes it.

This paper makes three contributions.

- We identify *late-layer reversal*: in knowledge conflict cases, the evidence answer is supported in intermediate computation but suppressed near the end of the forward pass.
- We show that linearly readable answer information and the output signal live on distinct directions in hidden-state space, challenging the assumption that the model's readout direction is also the right direction for diagnosing intermediate knowledge.
- We establish a causal asymmetry between the two tracks: intervening on the knowledge direction $\mathbf{V}$ has much weaker effects than suppressing prior-restoring updates along the output direction $\mathbf{D}$, which makes the model favor the contextual evidence.

2 Related Work

Mechanistic interpretability and representation dynamics. Our analysis builds on a growing body of work that traces information flow through transformer internals. The "Transformer Circuits" programme identified compositional attention patterns and the hidden state as the substrate of learned algorithms [Elhage et al., 2021, Olsson et al., 2022], while subsequent work localized factual recall to specific feedforward neurons [Geva et al., 2021, Dai et al., 2022] and to identifiable causal pathways amenable to targeted editing [Meng et al., 2022, Hernandez et al., 2024, Wang et al., 2023, Conmy et al., 2023]. Complementary to these component-level studies, Abnar and Zuidema [2020] introduced attention rollout as a method for quantifying token-level information flow, and Dong et al. [2021]

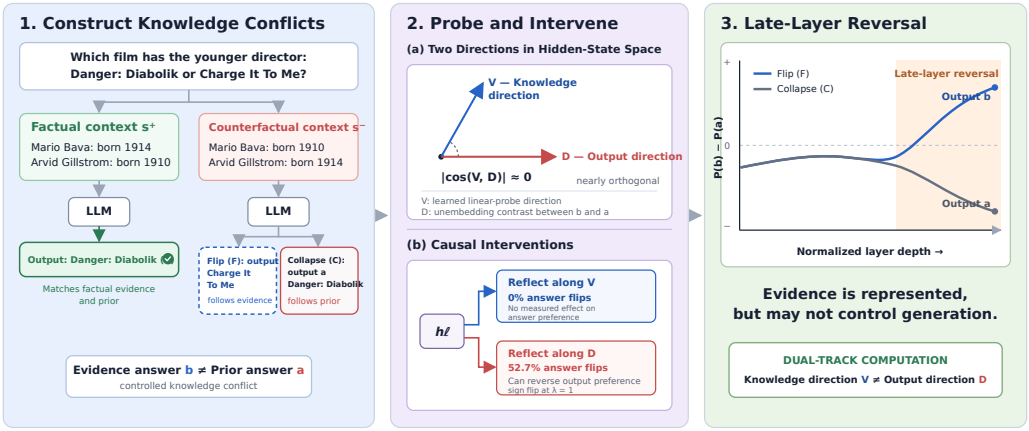

Figure 1: Overview of Knowledge Conflict and Dual-Track Computation. (1) Under factual context s^+, the model correctly outputs a; under counterfactual context s^-, it either follows evidence b (flip, F) or reverts to a (collapse, C). (2) The knowledge direction V and output direction D are nearly orthogonal. Interventions along V produce 0% answer flips, versus 52.7% along D, revealing their distinct causal roles. (3) Flip (blue) and collapse (gray) cases follow similar trajectories through the first $\sim$60% of layers but sharply diverge thereafter, exhibiting late-layer reversal. Together, these results show that contextual evidence can remain encoded without controlling the final output.

showed that deep pure-attention networks suffer rank collapse, where high-frequency localized signals are attenuated as depth increases. The logit lens and its descendants probe the layerwise evolution of the model's draft answer by projecting intermediate representations through the output unembedding [Geva et al., 2023].

Multi-hop QA, context utilisation, and knowledge conflicts. Standard multi-hop QA benchmarks, including HotpotQA [Yang et al., 2018], 2WikiMultiHopQA [Ho et al., 2020], and MuSiQue [Trivedi et al., 2022], provide a natural setting for studying knowledge conflicts: the answer depends on correctly using multiple pieces of in-context evidence, while the model may still default to its parametric priors. A parallel literature has shown that LLMs do not always faithfully utilize provided context. Longpre et al. [2021] demonstrated that models exhibit systematic biases when contextual and parametric knowledge conflict, and Shi et al. [2023] showed that irrelevant context can degrade performance even when the relevant evidence is present. The "lost in the middle" phenomenon [Liu et al., 2024] further revealed that models under-utilize evidence placed in the interior of long contexts, suggesting bottlenecks in information routing.

However, standard evaluations of these benchmarks typically rely on end-to-end accuracy, which cannot distinguish between a model that never encoded the evidence and one that encoded it but suppressed it before output. Press et al. [2023] introduced the "compositionality gap" to measure the shortfall between single-hop and multi-hop performance, but this metric still aggregates across instances without separating failures of evidence encoding from failures of evidence conversion. Similarly, the broader literature on LLM factual errors often uses the umbrella term "hallucination" [Ji et al., 2023, Min et al., 2023], which conflates fabrication, retrieval failure, and the kind of conditional non-updating we study here.

3 The Phenomenon: Knowledge Conflict Is a Late-Layer Event

Did the model truly ignore the evidence? We examine the model's internal computation layer by layer to trace when the evidence answer b gains support and when that support is overridden before generation.

Experiment Setup We study three instruction-tuned models: Llama-3.1-8B-Instruct, Mistral-7B-Instruct-v0.3, and Qwen-2.5-7B-Instruct, and three multi-hop QA benchmarks (HotpotQA,

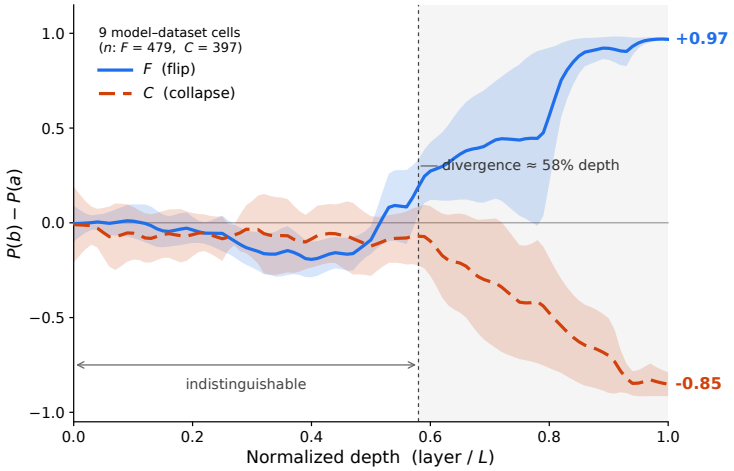

Figure 2: F **and** C **trace indistinguishable trajectories until late layers, then diverge in opposite directions.** Mean per-layer $P(b) - P(a)$ pooled across all nine model–dataset cells: F (blue, solid; total number $n = 479$) is the flip group; The red dashed curve shows the recollapse subset of C, with $n = 397$; Appendix A.6 defines this subset. Shaded bands show the 95% confidence interval (CI) across cells. The two groups remain statistically indistinguishable through approximately the first 60% of normalized depth (horizontal arrow), then separate sharply: by the final layer, F reaches $+0.97$ while C reaches -0.85.

2WikiMultihopQA, MuSiQue), yielding nine model–dataset cells [2]. For each cell, we construct paired prompts: a question q with a supportive context s^+ that elicits the correct answer a, and the same question with a counterfactual context s^- whose evidence supports a different answer b. We restrict analysis to questions the model already answers correctly under s^+, so that any change in behavior under s^- reflects how the model handles conflicting evidence, not whether it can solve the question at all.

Output-based classification. For each question, we run inference under s^- and classify the result by the model's final output into one of three groups: F (the model outputs b, following the counterfactual evidence), C (the model outputs a, reverting to its prior), or O (the model outputs a token that is neither a nor b). This paper focuses on the collapse group C, where the model has access to contextual evidence supporting b in the prompt but ultimately outputs its prior a, and asks what happens inside the network to produce this outcome. Across the nine cells, this yields 479 F, 560 C, and 2,452 O cases (Table 5). Most collapse cases (65.2% on average) exhibit *recollapse*, where the logit lens initially favors b before reverting to a (Appendix A.6). This reflects *late-layer reversal*, but given its limited frequency, this reversal is weaken in the overall trajectory.

Logit lens. To observe how the model's internal preferences evolve across layers, we use the *logit lens* [Sakarvadia et al., 2023]. Let $\mathbf{h}_\ell \in \mathbb{R}^d$ denote the hidden state at layer ℓ, extracted at the final token position of the prompt. The logit lens projects $\mathbf{h}_\ell$ through the model's unembedding matrix $\mathbf{W}_U$, the matrix that the model applies after its final layer to convert hidden states into a logit for each vocabulary token,

$$z_\ell(t) \;=\; (\mathbf{W}_U\,\mathbf{h}_\ell)_t \;\in\; \mathbb{R}, \tag{1}$$

and converts these logits into probabilities $P_\ell(t)$ via softmax. We write $P_\ell(t)$ for the scalar probability assigned to token t at layer ℓ. For each (q, s^-, a, b) pair, we define the answer-level probability as the mean over each answer's constituent tokens: $P_\ell(a) = \frac{1}{|a|}\sum_{t\in a} P_\ell(t)$, where $|a|$ is the number of tokens in answer a, and likewise for b. We track $P_\ell(a)$ and $P_\ell(b)$ across layers and say that the model *favors* b *at layer* ℓ when $P_\ell(b) > P_\ell(a)$. We call this layer-by-layer sequence the *trajectory*. Other analyses score answers differently; Appendix A.5 lists the readout behind each figure and table.

The mean trajectories for F and C are statistically indistinguishable through approximately the first 60% of network depth: both groups remain near indecision ($P(b) - P(a) \approx 0$) as the model

[2] Please refer to Appendix A for details.

integrates the contextual evidence (Figure 2). Only in the final $\sim 40\%$ of layers do they diverge sharply. F rises steeply toward the evidence answer. C abruptly reverses in the opposite direction, suppressing b and restoring the prior answer a.

The model's own readout locates this divergence in the network's late layers, after F and C have traced similar trajectories through more than half the forward pass. We now study two questions in the following passage: **what happens in those late layers**, and **why is the model's own earlier computation not reflected in the final output**?

4 The Latent Answer: The Model Knows in Mid-Layers

The trajectory evidence in §3 establishes that the evidence answer b gains strength in middle layers before being suppressed. We now ask when the eventual answer preference becomes linearly predictable from the hidden state.

Linear margin lens. Following the tuned lens framework [Belrose et al., 2023], we train a per-layer linear probe to predict, from the hidden state $\mathbf{h}_\ell$, the logit margin that the *full model* assigns between b and a at its final layer L. The training target is the final-layer margin $m^* = z_L(b) - z_L(a)$, which measures the model's final preference between these two answers. By regressing m^* from earlier layers, the probe asks whether the model's eventual output preference is already linearly predictable from $\mathbf{h}_\ell$.

$$\hat{m}_\ell \;=\; \mathbf{w}_\ell^\top \mathbf{h}_\ell \;+\; \beta_\ell \;\in\; \mathbb{R}, \tag{2}$$

$\mathbf{w}_\ell \in \mathbb{R}^d$ is the learned weight vector, and $\beta_\ell \in \mathbb{R}$ is a bias term. The probe bypasses both the output matrix $\mathbf{W}_U$ and the final LayerNorm, learning whatever linear function of $\mathbf{h}_\ell$ best predicts the eventual margin. We fit the probe on equal numbers of F and C cases, excluding O, split 70/30 into train and test and resampled with eight seeds, and report held-out results in Appendix D.

Result: identical start, progressive divergence. Figure 3 plots the share of cases at each layer where the probe returns $\hat{m}_\ell > 0$ (b-fraction) for held-out F and C cases, because the three models have different layer counts (Llama: 32, Mistral: 32, Qwen: 28), we normalize layer indices to $[0, 1]$ before averaging across the eight cells with balanced-probe results.

The two curves (green solid and dashed) show how the eventual outcomes become distinguishable through the network. At early layers (0–10% depth), both groups stay near the embedding-layer baseline of 58% (Appendix D). Starting at $\sim$20% depth, the curves diverge: the F-curve rises toward $\sim$80% while the C-curve declines, with the gap exceeding 30 pp by 35% depth and 50 pp just past mid-depth. At the final layer, F cases have a b-fraction of 75%, while C cases retain 34%. Six of the eight cells show this separation on their own; the two MuSiQue cells, with two held-out cases per group, do not (Table 8).

Interpretation. The probe distinguishes eventual outcomes earlier than the model's own output readout. From $\sim$20% depth, F and C cases separate along the direction that predicts the final answer preference, whereas the model's own readout separates them at $\sim$60% depth (Figure 2). Together with late-layer reversal in Section 3, this temporal gap motivates the distinction between linearly readable answer information and the output direction tested next. There is no single layer at which this happens; the separation accumulates across the middle of the network, with the steepest decline of the C-curve between 20% and 50% depth (see Figure 3A).

Comparison with the model's own readout. The naive logit lens (brown lines), which applies the model's output matrix $\mathbf{W}_U$ directly, shows a qualitatively similar pattern for C cases: moderate b-fraction at mid-layers ($\sim$40–50%) dropping to near zero at the final layer. This confirms that the divergence between F and C is visible even through the model's own readout mechanism, though the signal is noisier than under the learned probe.

Ablation: is the signal along the model's output direction? To test whether the evidence signal lives along the model's own output direction or along a different direction, we compare the full probe against a restricted variant.

Recall that the embedding matrix $\mathbf{W}_U \in \mathbb{R}^{|\mathcal{V}| \times d}$ maps hidden states to vocabulary logits (Eq. 1). We write $\mathbf{u}_t \in \mathbb{R}^d$ for the row of $\mathbf{W}_U$ corresponding to token t: this is the direction in hidden-state space that the model associates with token t.

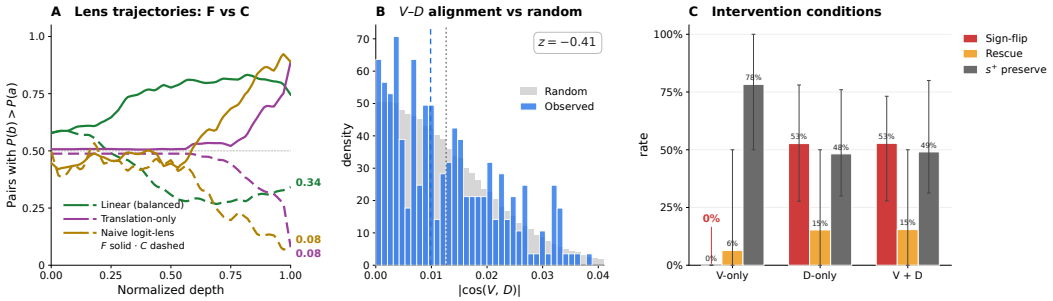

Figure 3: **Linearly readable answer information and output preference follow distinct directions.** **(A)** Fraction of cases with $P(b) > P(a)$ (or $\hat{m}_\ell > 0$) by normalized depth, averaged over the eight cells with balanced-probe results. The linear margin lens is the class-balanced probe on held-out F and C cases (mean over eight resamples); the translation-only and naive lenses use F and recollapse cases. The linear lens separates F from C from ∼20% depth; the naive logit lens never exceeds ∼50% for group C. **(B)** $|\cos(\mathbf{V}, \mathbf{D})|$ across all collapse pairs (blue) versus random directions (gray); z-score (z) = -0.41. **(C)** Conditional residual suppression at λ=1.0 gives sign-flip rates of $\mathbf{V}$-only $= 0\%$, $\mathbf{D}$-only $= 52.7\%$, $\mathbf{V}$+$\mathbf{D}$ $= 52.8\%$. The two directions differ in their effect on answer preference. Panels B and C use $\mathbf{V}$ from the original probe.

Define $\mathbf{D} = (\mathbf{u}_b - \mathbf{u}_a)/\|\mathbf{u}_b - \mathbf{u}_a\|$ as the unit vector along the difference between the unembedding vectors of the first tokens at which b and a differ, along which the model's output matrix distinguishes b from a. We train a *translation-only* variant that removes the probe's ability to choose its own direction: it fixes the weight vector to $\mathbf{D}$ and learns only a per-layer bias, $\hat{m}_\ell^{\text{fixed}} = \mathbf{D}^\top \mathbf{h}_\ell + \beta_\ell$.

This ablated probe does not separate F from C before ∼75% depth (both stay near 50%; Figure 3A), whereas the full probe (Eq. 2), which is free to learn any direction $\mathbf{w}_\ell$, separates them from ∼20% depth. The separating signal is therefore *not* primarily encoded along $\mathbf{D}$: the direction from which the full probe reads it differs from the direction along which the model computes its output. We formalize this directional mismatch and test its causal implications in the next section.

The analysis so far involves two directions that we compare throughout the remainder of the paper. The *output direction* $\mathbf{D}$ is the fixed unembedding contrast that the model applies at its final layer to produce the logit difference between b and a. The *knowledge direction* $\mathbf{V}$ is the per-layer probe weight $\mathbf{w}_\ell$: the direction along which the model's eventual output preference is most linearly predictable from $\mathbf{h}_\ell$. We use the term "knowledge" not to claim that the model possesses a semantic understanding of the answer, but in a strictly operational sense: $\mathbf{w}_\ell$ captures whatever linear information in $\mathbf{h}_\ell$ is most predictive of the final margin m^*. The geometric and causal tests of Sections 5–6 use the weights of the original probe described in Appendix D.

The target $m^* = (\mathbf{u}_b - \mathbf{u}_a)^\top \mathbf{h}_L$ is the signed projection of the final hidden state onto the pair's output direction ($\mathbf{D}$ up to scale), so the probe predicts how far the final state moves toward b on the model's own output axis. $\mathbf{V}$ identifies where the eventual preference is linearly readable, and $\mathbf{D}$ where it is expressed in the output. Near-orthogonality is common in high dimension; Section 5.1 therefore tests the two directions through intervention.

The remainder of the paper asks two questions. First, are $\mathbf{w}_\ell$ and $\mathbf{D}$ geometrically aligned? If so, the progressive loss of the evidence signal is simply the model's output computation overwriting it along the same axis. If not, the most predictive direction and the causally effective direction live on independent axes, and the model's failure to follow the context has a geometric signature. Second, can the override be reversed by intervening on $\mathbf{D}$ alone, and if so, does the model then follow the contextual evidence?

5 The Disconnection: Knowing and Saying Live on Distinct Axes

Section 4 showed that a per-layer linear probe separates collapse from flip cases from ∼20% depth, long before the model's own readout does. We now ask whether this knowledge direction $\mathbf{V}$ is geometrically aligned with the output direction $\mathbf{D}$.

If $\mathbf{V}$ and $\mathbf{D}$ were aligned, it would mean the probe reads the evidence signal from the same direction that the model uses to produce its output, and the ablation in Section 4 already suggests they are not, since constraining the probe to $\mathbf{D}$ removes this separation until $\sim$75% depth. We now test this directly.

We compute $|\cos(\mathbf{V}, \mathbf{D})|$ for every collapse pair and compare to a null distribution of 1,000 random unit vectors per pair drawn uniformly from the hidden state.

Table 1: **$\mathbf{V}$ is no more aligned with $\mathbf{D}$ than a random direction.** Aggregated across 9 cells (361 collapse pairs).

	$\lvert\cos(\cdot, \mathbf{D})\rvert$
Knowledge direction $\mathbf{V}$	0.0088
Random baseline	0.0128 ($\sigma = 0.0097$)
z-score	-0.41

Table 1 reports the result. For each collapse case, we compute $|\cos(\mathbf{V}, \mathbf{D})|$ and compare it to a null distribution obtained by replacing $\mathbf{V}_\ell$ with 1000 random unit vectors drawn uniformly in $\mathbb{R}^d$. The mean observed alignment is 0.0088; the random baseline has mean 0.0128 and standard deviation 0.0097. The z-score $= \frac{|\cos(\mathbf{V},\mathbf{D})| - \mu_{\text{random}}}{\sigma_{\text{random}}} = -0.41$, indicates that $\mathbf{V}_\ell$ is no more aligned with $\mathbf{D}$ than a random direction would be: the observed value falls within 0.5 standard deviations of the random mean, well inside the null distribution (Figure 3B). Per cell, the z-score is negative in eight of nine cells (Appendix E).

5.1 Causal influence via directional suppression

The linear representation hypothesis [Park et al., 2024] holds that if a concept is encoded as a direction in the model's hidden states, then intervening along that direction should influence the model's output accordingly. We test for causal influence directly by intervening on the hidden state and measuring whether the model's final output changes.

Intervention. At layer ℓ, we remove the negative component of the residual update $\Delta\mathbf{h}_\ell = \mathbf{h}_\ell - \mathbf{h}_{\ell-1}$ along a target unit vector $\hat{\mathbf{w}}$:

$$\Delta\mathbf{h}_\ell \leftarrow \Delta\mathbf{h}_\ell - \min(\langle\Delta\mathbf{h}_\ell, \hat{\mathbf{w}}\rangle, 0)\,\hat{\mathbf{w}}. \tag{3}$$

This removes updates opposed to $\hat{\mathbf{w}}$ while leaving positive projections and all orthogonal components unchanged. We apply this intervention with three choices of $\hat{\mathbf{w}}$: the knowledge direction at the selected peak layer ℓ^*, $\hat{\mathbf{w}} = \mathbf{V}_{\ell^*}/\|\mathbf{V}_{\ell^*}\|$, held fixed across targeted layers, the output direction $\hat{\mathbf{w}} = \mathbf{D}$, and both together. In the combined condition, both projections are computed from the same residual update and their corrections are added. Section 6 gives the layer-selection rule and varies the suppression strength.

Table 2: **The output direction $\mathbf{D}$ has a stronger causal effect than the knowledge direction $\mathbf{V}$.** Rates are macro-averaged across nine cells with 360 cases in total. Answer-flip rate: fraction of cases whose b-versus-a margin changes from negative to positive after intervention. Behavioral rescue: fraction where the generated answer matches b and does not match a under normalized substring matching.

Metric	$\mathbf{V}$-only	$\mathbf{D}$-only	$\mathbf{V}$+$\mathbf{D}$
Answer-flip rate	**0.0%**	52.7%	52.8%
Behavioral rescue	6.4%	15.3%	15.4%

Answer-flip rate. We compare the model's output before and after intervention. We say an *answer-flip* has occurred if the final-layer b-versus-a margin is negative before intervention and positive afterward. Within each cell, the *answer-flip rate* is the fraction of all evaluated collapse cases in which this reversal occurs. For behavioral rescue, we lowercase the generated and reference answers, remove punctuation and articles, and test whether the generation contains b but not a.

Table 2 and Figure 3C report the results. Suppressing along $\mathbf{V}$ produces an answer-flip rate of **0%** across every case. Its behavioral rescue rate is 6.4%. Intervention along $\mathbf{D}$ produces an answer-flip rate of 52.7%. Suppressing along both $\mathbf{V}$ and $\mathbf{D}$ yields 52.8%, only 0.1 percentage points above $\mathbf{D}$-only. Adding $\mathbf{V}$ therefore has little effect on this measure.

Other operators. The asymmetry holds across operators: residual suppression along $\mathbf{V}$ gives 0% sign-flip at every strength (Table 3); additive steering along $\mathbf{V}$, evidence-difference patching, and a null patch give 1.3%, 0.4%, and 0.0% sign-flip (4.4%, 3.9%, and 0.0% answer rescue), compared with 52.7% for $\mathbf{D}$-only suppression. Appendix F reports the full results.

5.2 Dual-Track Computation

The geometric and causal results converge on the same conclusion. Answer information is linearly readable along $\mathbf{V}$, while intervention along $\mathbf{D}$ has a much stronger effect on the output. We name this regime **Dual-Track Computation**. Late transformer layers maintain two concurrent, co-located representations in the same hidden state: a *knowledge track* encoding model knowledge and an *output track* determining model output.

6 Suppressing Prior Restoration on the Output Track

Dual-Track Computation predicts that suppressing the prior-favoring component along $\mathbf{D}$ only should restore the evidence answer. We test this with targeted interventions and controls designed to establish directional, spatial, and behavioral specificity.

6.1 Conditional residual suppression

The intervention in Section 5 removes prior-restoring computation across layers. It operates on the per-layer residual update $\Delta\mathbf{h}_\ell = \mathbf{h}_\ell - \mathbf{h}_{\ell-1}$ rather than the full state. Each $\Delta\mathbf{h}_\ell$ is the contribution of layer ℓ alone; modifying it suppresses what that specific layer adds without disturbing the information accumulated by preceding layers [Geva et al., 2022, Meng et al., 2022]. This allows us to selectively suppress the prior-restoring component at each layer independently.

For each collapse case, we examine the per-layer residual update $\Delta\mathbf{h}_\ell = \mathbf{h}_\ell - \mathbf{h}_{\ell-1}$ within the post-peak intervention window, from the selected evidence-peak layer through the final transformer block. Appendix G gives the window-selection rule. We compute the signed projection $s = \langle \Delta\mathbf{h}_\ell, \mathbf{D} \rangle$: when $s < 0$, the layer is pushing the hidden state away from b and toward a. We suppress this component:

$$\Delta\mathbf{h}_\ell \leftarrow \Delta\mathbf{h}_\ell - \lambda \cdot s \cdot \mathbf{D}, \quad \text{iff} \quad s < 0. \tag{4}$$

At λ=1.0, this removes the prior-restoring component along $\mathbf{D}$ entirely; at $\lambda > 1.0$, it over-corrects by reversing the sign of that component. Layers where $s \geq 0$ are left untouched.

This follows standard practice in mechanistic interpretability: causal tracing [Meng et al., 2022], for example, corrupts a model's input and selectively restores activations at specific layers to identify which layers store a factual association, using the known correct answer to construct the clean and corrupted runs. Like causal tracing, our intervention uses the known answers a and b to test a mechanistic hypothesis.

6.2 Varying intervention strength

Table 3 reports the key dose-response results. At λ=1.0 (complete removal of the negative $\mathbf{D}$-component), $P(b) > P(a)$ reaches 61.4%. At λ=2.0, it reaches **96.9%**, with sign-flip at 88.2% and behavioral rescue at 46.4% (best cell: 70.0%, Llama/HotpotQA). Throughout, $\mathbf{V}$-only intervention remains at exactly **0% sign-flip** at every scaling factor, which is a flat line against the monotonically rising $\mathbf{D}$-axis curve (Figure 3).

Over-determination. By construction, λ=1.0 *completely* removes the negative $\mathbf{D}$-component at every targeted layer, yet achieves only 52.7% sign-flip. Reaching near-saturation requires λ=2.0; even λ=3.0 gains only +1.7 pp further (89.9% sign-flip, +9.4 pp rescue at 55.8%). Prior restoration is structurally redundant, distributed across both layers and directions beyond the targeted $\mathbf{D}$-axis.

Table 3: **Effect of intervention strength λ.** Macro-averaged across 9 cells (360 collapse pairs). $\mathbf{V}$-only sign-flip is 0% at every λ.

	$\mathbf{D}$-only				$\mathbf{V}$-only
λ	$P(b)>P(a)$	Sign-flip	Rescue	s^+ pres.	Sign-flip
1.0	61.4%	52.7%	15.3%	48.2%	0.0%
2.0	96.9%	88.2%	46.4%	37.2%	0.0%
3.0	98.7%	89.9%	55.8%	33.8%	0.0%

Where prior restoration acts. On 229 recollapse cases (App. A.6) from the five-dataset sweep (§8), negative-$\mathbf{D}$ updates occur at 17.9 layers per example on average, but 89.7% of their mass lies in the final third. Suppressing five early or middle layers leaves $P(b)>P(a)$ at baseline (6.6%, 7.9%), five and ten late layers reach 48.0% and 55.5%, all negative layers 79.0%, and the single largest 13.5%; cumulative mass tracks the margin shift better than layer count (Spearman $\rho = 0.90/0.75/0.71$ vs. $0.62/0.50/-0.18$ for Llama/Mistral/Qwen; Appendix I).

The gap between real intervention and the strongest directional control is about **53 percentage points** on $P(b) > P(a)$ in Appendix H. The intervention is specific to the correct direction, the correct layers, and the correct behavioral regime.

Off-target cost. On 166 matched held-out clean prompts across 13 cells, suppression turns 17.5% of correct answers incorrect at λ=2 and 42.2% at λ=3 (macro over cells: 24.3% and 52.0%). At λ=2, outputs change on 44.0% of prompts and the gold-answer log-probability drops by 2.47, while 1.8% are malformed. A random orthogonal direction changes correct answers to incorrect ones in 1.8% of cases, compared with 15.1% for a shuffled answer-contrast direction. These costs limit the intervention's use as a correction method; full results are in Appendix J.3.

7 The Ceiling: Why Representational Recovery Does Not Become Behavioral Recovery

Section 6 showed that suppressing prior-restoration along $\mathbf{D}$ causes the model to favor b over a in up to 96.9% of collapse cases. But favoring b in probability and actually *outputting* b are different things: behavioral rescue under normalized substring matching reaches only 46.4% at λ=2.0. What does the model output in the remaining cases?

Annotation of non-b outputs. After suppression at λ=2.0, we apply relaxed matching (accepting partial names, common aliases, and alternative spellings as correct) and count those cases as successful rescues. For the remaining 108 cases, where outputs do not match b even under relaxed matching, we classified each by its relationship to the evidence answer (Table 4).

Table 4: **Classification of non-b outputs after relaxed matching** (λ=2.0, 108 cases).

Category	n	%
B_related: vocabulary-geometry spillover	80	74.1
A_format: format breakdown	12	11.1
B_alias: near-match of b	14	13.0
A_residual: residual prior leakage	2	1.9

B_related (74.1%) are tokens semantically related to b but not valid answers under any reasonable matching criterion. These include hypernyms (a category name when b is a specific entity, e.g., "tennis player" when b is "Jonathan Stark"), co-occurring entities (another person mentioned in the same passage as b), attributes of b (a nationality or title associated with b rather than b itself), and topically related terms from the same domain. The model has moved to the correct semantic region of the output space but selects a neighboring concept rather than the target.

A_format (11.1%) are structural tokens, e.g. newlines, punctuation, or preamble fragments, where the suppression disrupted generation format rather than answer selection.

B_alias (13.0%) are borderline near-matches that fall outside the relaxed matching criteria but are arguably related to the correct answer.

A_residual (1.9%) are tokens associated with the prior answer a, indicating incomplete suppression of the prior in a small number of cases.

Taken together, 87.0% of failures after relaxed matching (B_related + B_alias) remain in the semantic neighborhood of b. Even after filtering out partial names and aliases, the model drifts toward b-related tokens rather than random outputs.

Decomposing the gap. The 96.9% figure measures pairwise preference for b over a; exact generation also requires b to beat the full vocabulary and persist through the continuation. In a separate five-dataset sweep of 229 recollapse cases, $\mathbf{D}$-only suppression places b's first token at top-1 in 91.7% of cases, with 56.8% strict exact-match and 81.2% semantic rescue. Of these cases, 80 have the correct first token but miss strict exact generation, while 18 still have a stronger first-token competitor. An extra fine-grained direction orthogonal to $\mathbf{D}$ raises first-token top-1 to 98.7% but strict exact-match rescue only to 61.1%. Continuation and answer matching therefore account for much of the remaining gap in this sweep; Appendix M gives the analysis.

8 Beyond the Main Setting

Scale. On Qwen3.5-27B with CounterFact, late-layer reversal occurs in all 61 collapse cases, and $\mathbf{V}$ and $\mathbf{D}$ remain near-orthogonal ($|\cos| = 0.006$ vs. 0.011 for random directions). $\mathbf{D}$-only suppression at λ=2 gives 100% sign-flip and 68.8% exact rescue on 48 cases, whereas $\mathbf{V}$-only suppression at λ=3 gives 20.8% sign-flip and 0% exact rescue. With one larger model and one dataset, we do not claim scale generality (Appendix N.1).

Task format and long-form answers. On single-hop CounterFact [Meng et al., 2022], suppressing all prior-restoring updates gives 63.2–100% sign-flip across the three models, so the effect does not depend on multi-hop composition; ConflictQA [Xie et al., 2024] results are in Appendix N. In five selected long-form cells, per-step suppression raises answer accuracy from 5.3–30.0% with first-step suppression to 70.0–91.7%; three other cells show no change, and full-answer consistency remains lower (0.0–75.0%), particularly on multi-hop tasks. For sentiment, we select real entities with stable no-context polarity and construct paired reviews: supportive reviews match the prior, while conflicting reviews express the opposite polarity, matched in length and syntax. Models generate a four-sentence paragraph comprising an assessment, two evidence-based reasons, and a recommendation. With $\mathbf{D}$ suppression at every decoding position, full-paragraph consistency (41.7/100/91.7% for Llama/Qwen/Mistral) exceeds the evidence baseline (91.7/91.7/83.3%) for Qwen and Mistral but not Llama; baseline scoring may differ (Appendix N.3).

9 Discussion and Conclusion

The empirical arc of this paper converges on a single claim: in the regime we study, transformer late layers maintain two functionally distinct representational tracks within the same hidden state.

Limitations. Our analysis is restricted to knowledge conflict failures, mainly in multi-hop QA; whether Dual-Track Computation governs other failure modes, such as hallucination and sycophancy, remains open. We study three 7–8B models and evaluate the 27B model on one dataset; frontier and reasoning-tuned models remain open. Base and instruction-tuned checkpoints agree on the closed-book answer in 98.1–100% of items for Qwen and Mistral and 66.7–100% for Llama, suggesting that many priors predate instruction tuning. Causal tracing points mostly to late residual and MLP components, but does not identify the training documents from which these priors were acquired. Appendix O gives these analyses.

We set out to understand why language models revert to their priors when contextual evidence disagrees. In the cases we study, contextual evidence influences intermediate computation before the prior is restored. The direction carrying linearly readable answer information and the direction determining the output are nearly orthogonal and have different causal effects. The geometric comparison covers three models, three datasets, and 361 collapse cases.

These findings distinguish linear representation from causal control: a signal can be linearly readable from the hidden state while interventions along that direction have little effect on the output. Improving context-faithfulness may therefore require not only better encoding of evidence but stronger coupling between context-derived representations and the model's output computation.

Acknowledgments and Disclosure of Funding

We thank our colleagues and funding agencies for their support. This work was supported by the DARPA AIQ Program, the U.S. Department of Energy under Award No. DE-SC0025584, the U.S. National Science Foundation under Award No. DMS-2616827, the DOE Mission Science Allocation Year 2026 Award, the NAIRR Pilot under Award No. NAIRR260021, and Dartmouth College.

References

Samira Abnar and Willem Zuidema. Quantifying attention flow in transformers. In *Proceedings of the 58th annual meeting of the association for computational linguistics*, pages 4190–4197, 2020.

Nora Belrose, Igor Ostrovsky, Lev McKinney, Zach Furman, Logan Smith, Danny Halawi, Stella Biderman, and Jacob Steinhardt. Eliciting latent predictions from transformers with the tuned lens. *arXiv preprint arXiv:2303.08112*, 2023.

Arthur Conmy, Augustine N. Mavor-Parker, Aengus Lynch, Stefan Heimersheim, and Adrià Garriga-Alonso. Towards automated circuit discovery for mechanistic interpretability. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

Damai Dai, Li Dong, Yaru Hao, Zhifang Sui, Baobao Chang, and Furu Wei. Knowledge neurons in pretrained transformers. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*, 2022.

Guy Dar, Mor Geva, Ankit Gupta, and Jonathan Berant. Analyzing transformers in embedding space. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (ACL)*, 2023.

Yihe Dong, Jean-Baptiste Cordonnier, and Andreas Loukas. Attention is not all you need: Pure attention loses rank doubly exponentially with depth. In *Proceedings of the 38th International Conference on Machine Learning*, 2021.

Nelson Elhage, Neel Nanda, Catherine Olsson, Tom Henighan, Nicholas Joseph, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Nova DasSarma, Dawn Drain, Deep Ganguli, Zac Goldberg, Chris Hallacy, Danny Hernandez, Scott Johnston, Jackson Kernion, Liane Lovitt, Kamal Ndousse, Dario Amodei, Tom Brown, Jack Clark, Jared Kaplan, Sam McCandlish, and Chris Olah. A mathematical framework for transformer circuits. *Transformer Circuits Thread*, 2021. URL https://transformer-circuits.pub/2021/framework/index.html.

Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. Transformer feed-forward layers are key-value memories. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 2021.

Mor Geva, Avi Caciularu, Kevin Wang, and Yoav Goldberg. Transformer feed-forward layers build predictions by promoting concepts in the vocabulary space. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2022.

Mor Geva, Jasmijn Bastings, Katja Filippova, and Amir Globerson. Dissecting recall of factual associations in auto-regressive language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 2023.

Evan Hernandez, Arnab Sen Sharma, Tal Haklay, Kevin Meng, Martin Wattenberg, Jacob Andreas, Yonatan Belinkov, and David Bau. Linearity of relation decoding in transformer language models. In *Proceedings of the Twelfth International Conference on Learning Representations*, 2024.

Xanh Ho, Anh-Khoa Duong Nguyen, Saku Sugawara, and Akiko Aizawa. Constructing a multi-hop QA dataset for comprehensive evaluation of reasoning steps. In *Proceedings of the 28th International Conference on Computational Linguistics*, 2020.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. Survey of hallucination in natural language generation. *ACM computing surveys*, 55(12):1–38, 2023.

Evgenii Kortukov, Alexander Rubinstein, Elisa Nguyen, and Seong Joon Oh. Studying large language model behaviors under context-memory conflicts with real documents. In *First Conference on Language Modeling*, 2024.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12, 2024.

Shayne Longpre, Kartik Perisetla, Anthony Chen, Nikhil Ramesh, Chris DuBois, and Sameer Singh. Entity-based knowledge conflicts in question answering. In *Proceedings of the 2021 conference on empirical methods in natural language processing*, pages 7052–7063, 2021.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in GPT. In *Advances in Neural Information Processing Systems*, volume 35, 2022.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen-tau Yih, Pang Wei Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. FActScore: Fine-grained atomic evaluation of factual precision in long form text generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 2023.

Harsha Nori, Nicholas King, Scott Mayer McKinney, Dean Carignan, and Eric Horvitz. Capabilities of gpt-4 on medical challenge problems. *arXiv preprint arXiv:2303.13375*, 2023.

Nostalgebraist. Interpreting gpt: The logit lens. *Blog Post*, 2020.

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. URL https://transformer-circuits.pub/2022/in-context-learning-and-induction-heads/index.html.

Kiho Park, Yo Joong Choe, and Victor Veitch. The linear representation hypothesis and the geometry of large language models. In *International Conference on Machine Learning*, pages 39643–39666. PMLR, 2024.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. Language models as knowledge bases? In *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP)*, pages 2463–2473, 2019.

Ofir Press, Muru Zhang, Sewon Min, Ludwig Schmidt, Noah A. Smith, and Mike Lewis. Measuring and narrowing the compositionality gap in language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, 2023.

Adam Roberts, Colin Raffel, and Noam Shazeer. How much knowledge can you pack into the parameters of a language model? In *Proceedings of the 2020 conference on empirical methods in natural language processing (EMNLP)*, pages 5418–5426, 2020.

Mansi Sakarvadia, Arham Khan, Aswathy Ajith, Daniel Grzenda, Nathaniel Hudson, André Bauer, Kyle Chard, and Ian Foster. Attention lens: A tool for mechanistically interpreting the attention head information retrieval mechanism. *arXiv preprint arXiv:2310.16270*, 2023.

Freda Shi, Xinyun Chen, Kanishka Misra, Nathan Scales, David Dohan, Ed Chi, Nathanael Schärli, and Denny Zhou. Large language models can be easily distracted by irrelevant context. In *Proceedings of the 40th International Conference on Machine Learning*, 2023.

Karan Singhal, Shekoofeh Azizi, Tao Tu, et al. Large language models encode clinical knowledge. *Nature*, 620:172–180, 2023.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. MuSiQue: Multihop questions via single hop question composition. *Transactions of the Association for Computational Linguistics*, 10, 2022.

Kevin 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 *Proceedings of the Eleventh International Conference on Learning Representations*, 2023.

Kevin Wu, Eric Wu, and James Y Zou. ClashEval: Quantifying the tug-of-war between an LLM's internal prior and external evidence. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 37, 2024.

Jian Xie, Kai Zhang, Jiangjie Chen, Renze Lou, and Yu Su. Adaptive chameleon or stubborn sloth: Revealing the behavior of large language models in knowledge conflicts. In *International Conference on Learning Representations*, volume 2024, pages 35623–35646, 2024.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W. Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 2018.

A Experimental Setup

A.1 Models

We study three instruction-tuned models: Llama-3.1-8B-Instruct [3] ($d = 4096$, $L = 32$ layers), Mistral-7B-Instruct-v0.3 [4] ($d = 4096$, $L = 32$ layers), and Qwen-2.5-7B-Instruct [5] ($d = 3584$, $L = 28$ layers). All experiments use greedy decoding (temperature $= 0$) and are conducted in half-precision (float16) on a single NVIDIA RTX PRO 6000 GPU.

A.2 Datasets

We use three multi-hop question-answering benchmarks: HotpotQA [Yang et al., 2018], 2WikiMultihopQA [Ho et al., 2020], and MuSiQue [Trivedi et al., 2022]. Each benchmark provides questions that require reasoning across two or more supporting passages.

For each question q with correct answer a, we construct a counterfactual context s^- by modifying the supporting passages so that the evidence points to a different answer b. The supportive context s^+ consists of the original, unmodified passages. We retain only questions for which the model outputs a under s^+ (correct under faithful context).

A.3 Worked example: constructing s^+ and s^-

We illustrate the paired-context construction with a concrete example from 2WikiMultihopQA.

Question and answer.

> **Question** q: "Which film has the younger director, Danger: Diabolik or Charge It To Me?"
>
> **Correct answer** a: Danger: Diabolik
> **Evidence answer** b: Charge It To Me

This is a comparison question requiring two reasoning hops: (1) identify each film's director, and (2) compare their birth years. The correct answer is *Danger: Diabolik* because its director (Mario Bava, born 1914) is younger than the director of *Charge It To Me* (born 1910).

Supportive context s^+. The supportive context consists of unmodified Wikipedia passages. We show only the two passages relevant to the reasoning chain (the full context includes 8–10 distractor passages as in the original dataset):

s^+: Supportive context (original passages)

[1] Danger: Diabolik
Danger: Diabolik is a 1968 Italian action film directed by Mario Bava. The film is based on the Italian comic book character Diabolik created by Angela and Luciana Giussani. It stars John Phillip Law and Marisa Mell.

[2] Mario Bava
Mario Bava (31 July 1914 – 27 April 1980) was an Italian filmmaker. He is regarded as a pioneer of Italian horror and gothic filmmaking, as well as the giallo genre.

[3] Charge It To Me
Charge It To Me is a 1919 American silent comedy film directed by Arvid E. Gillstrom. The film stars Billie Rhodes and was produced by National Film Corporation.

[4] Arvid E. Gillstrom
Arvid E. Gillstrom (17 February 1910 – 25 September 1935) was a Swedish-American film director known for silent comedies and short films.

[3] https://huggingface.co/meta-llama/Llama-3.1-8B-Instruct
[4] https://huggingface.co/mistralai/Mistral-7B-Instruct-v0.3
[5] https://huggingface.co/Qwen/Qwen2.5-7B-Instruct

Reasoning under s^+: Mario Bava born 1914; Arvid E. Gillstrom born 1910. Bava (1914) is younger than Gillstrom (1910), so the film with the younger director is Danger: Diabolik $\rightarrow$ answer a. The model outputs a under s^+: this pair passes the filter and enters our study.

Counterfactual context s^-. To construct s^-, we modify the relevant passages so that the in-context evidence supports b = Charge It To Me. The counterfactual edit swaps the birth years of the two directors:

s^-: Counterfactual context (modified passages)

[1] Danger: Diabolik
(unchanged)

[2] Mario Bava
Mario Bava (17 February **1910** – 27 April 1980) was an Italian filmmaker... ← modified: 1914 → 1910

[3] Charge It To Me
(unchanged)

[4] Arvid E. Gillstrom
Arvid E. Gillstrom (31 July **1914** – 25 September 1935) was a Swedish-American film director... ← modified: 1910 → 1914

Reasoning under s^-: The modified passages state Bava born 1910 and Gillstrom born 1914. A reader who trusts the context would conclude Gillstrom (1914) is younger than Bava (1910), making Charge It To Me the film with the younger director $\rightarrow$ evidence supports b.

Classification under s^-.

- If the model outputs *Charge It To Me* ($= b$): classified as **Flip** (F) — the model followed the counterfactual evidence.
- If the model outputs *Danger: Diabolik* ($= a$): classified as **Collapse** (C) — the model reverted to its prior despite the modified passages.
- If the model outputs something else: classified as **Other** (O).

What makes this a knowledge conflict. The model's parametric knowledge (from pretraining) encodes that Mario Bava was born in 1914. The counterfactual context s^- asserts he was born in 1910. When the model outputs a under s^- (a C case), it is relying on its parametric knowledge over the in-context evidence. Our paper investigates what happens *inside the network* in these cases: specifically, whether the contextual evidence (b = Charge It To Me) is encoded in the hidden states and where it is lost.

A.4 Attrition

Table 5 reports the number of (q, s^-, a, b) pairs per cell after filtering, and the output-based classification into F (outputs b), C (outputs a), and O (outputs neither).

A.5 Answer-level readouts

All logit-lens readouts apply the model's final normalization before $\mathbf{W}_U$ (Eq. 1); they differ in how an answer is scored. The trajectories of Section 3 and Appendix B use the mean answer-token probability P_ℓ. The recollapse/no-uptake split (Appendix A.6) uses the mean answer-token logit $\bar{z}_\ell$. The probe target m^*, the direction $\mathbf{D}$, the three lenses of Figure 3A, and all interventions use the logits of the first tokens at which a and b differ. The groups F, C, and O are defined from the generated answer string. For multi-token answers these readouts can disagree, so values are comparable only within a readout.

A.6 Recollapse subtypes within C

Table 6 reports the fraction of collapse cases that meet the recollapse criterion (the naive logit lens shows $\bar{z}_\ell(b) > \bar{z}_\ell(a)$ at some intermediate layer and the final margin is negative) versus no-uptake

Table 5: **Dataset sizes and output-based classification.** Only pairs where the model answers correctly under s^+ are retained.

Model	Dataset	Total	F	C	O
Llama-3.1-8B	2WikiMultihop	415	33	103	279
	HotpotQA	486	111	81	294
	MuSiQue	274	21	10	243
Mistral-7B	2WikiMultihop	415	36	134	245
	HotpotQA	486	93	111	282
	MuSiQue	240	21	12	207
Qwen-2.5-7B	2WikiMultihop	415	39	38	338
	HotpotQA	486	92	64	330
	MuSiQue	274	33	7	234
TOTAL		3,491	479	560	2,452

cases (where $\bar{z}_\ell(b)$ never exceeds $\bar{z}_\ell(a)$). The remaining C cases have a positive margin at some layer but a nonnegative final margin and are classified as other (last column). Their generated answer is a, but the mean answer-token logits at the last layer do not favor it; this can occur because the output label and $\bar{z}_\ell$ score multi-token answers differently (Appendix A.5).

Table 6: **Recollapse vs. no-uptake within collapse group C.**

Model	Dataset	$\lvert C\rvert$	Recollapse	No-uptake	Other
Llama	2WikiMultihop	103	81.6%	1.0%	17.5%
	HotpotQA	81	61.7%	3.7%	34.6%
	MuSiQue	10	60.0%	10.0%	30.0%
Mistral	2WikiMultihop	134	83.6%	3.7%	12.7%
	HotpotQA	111	58.6%	3.6%	37.8%
	MuSiQue	12	75.0%	8.3%	16.7%
Qwen	2WikiMultihop	38	73.7%	5.3%	21.1%
	HotpotQA	64	64.1%	4.7%	31.3%
	MuSiQue	7	28.6%	0.0%	71.4%
MACRO MEAN			65.2%	4.5%	30.3%

B Naive Logit Lens Trajectories

Figure 4 plots the mean $P_\ell(b) - P_\ell(a)$ of Section 3 by layer for F and for the recollapse subset of C in each of the nine cells. The naive logit lens uses the model's own unembedding matrix $\mathbf{W}_U$ with no learned transformation.

The C curves mostly stay near zero through the middle layers and turn toward the prior answer in the final layers, while the F curves rise toward b.

C Fixed-Direction Probe Ablation

Section 4 compares three probes of increasing flexibility: the naive logit lens ($\hat{\mathbf{D}}^\top \mathbf{h}_\ell$, no learning), the fixed-direction probe ($\hat{\mathbf{D}}^\top \mathbf{h}_\ell + \beta_\ell$, bias only), and the full linear margin lens ($\mathbf{w}_\ell^\top \mathbf{h}_\ell + \beta_\ell$, direction and bias).

Table 7 reports the b-fraction at the peak layer ℓ^* for each probe type across all nine cells.

The fixed-direction probe stays near 50%, while the full margin lens peaks at 100% in every cell. Because the full probe's output is the same for every example at the first layers (Appendix D), such a peak need not reflect example-specific signal. The test of whether that signal lies along $\hat{\mathbf{D}}$ is

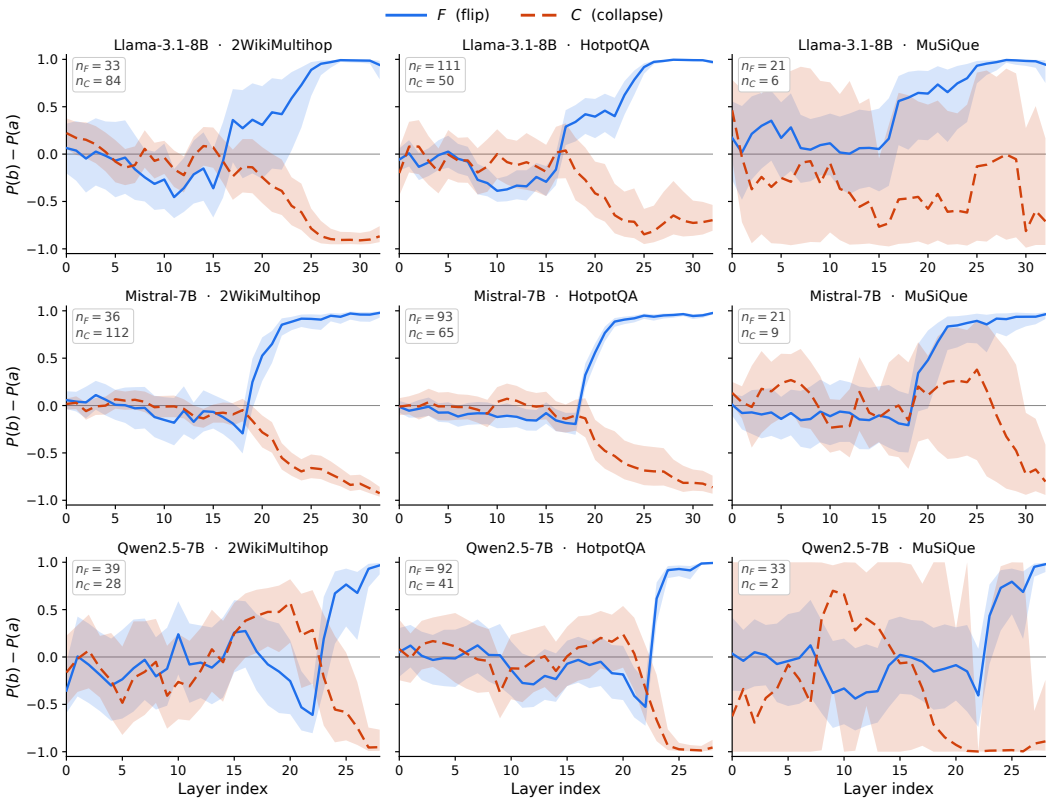

Figure 4: **Naive logit lens trajectories per cell.** Mean $P_\ell(b) - P_\ell(a)$ by layer; C is the recollapse subset (Appendix A.6).

Table 7: b**-fraction at peak layer by probe type and cell** (original probe; Appendix D). n_C: number of recollapse cases.

Model	Dataset	n_C	Naive	Fixed-dir	Margin lens
Llama	2Wiki	11	—	52.5%	100%
	Hotpot	50	—	38.0%	100%
	MuSiQue	6	—	66.7%	100%
Mistral	2Wiki	16	—	38.8%	100%
	Hotpot	65	—	41.5%	100%
	MuSiQue	7	—	44.4%	100%
Qwen	2Wiki	5	—	60.7%	100%
	Hotpot	41	—	58.5%	100%
	MuSiQue	1	—	50.0%	100%
MACRO MEAN		202	∼13%	50.1%	100%

Figure 3A: the fixed-direction probe does not separate F from C before ∼75% depth, whereas the balanced full probe separates them from ∼20% depth.

D Linear Margin Lens: Training Details

Probe. For each model–dataset cell and each layer ℓ, including the embedding layer, we fit a Ridge regression probe (scikit-learn `Ridge`, $\alpha = 100$, with intercept) on the hidden state $\mathbf{h}_\ell \in \mathbb{R}^d$ at the final prompt token. The training target is the final-layer logit margin between the first tokens at which the two answers differ, $m^* = z_L(b) - z_L(a)$; targets are centered before fitting and their

mean is added back to the intercept. The probe is trained once per layer and applied to all evaluation cases—no per-case fitting is performed.

Balanced probe (Figure 3A). For each cell we draw equal numbers of F and C cases and no O cases ($n = \min(n_F, n_C, 120)$ per group), split them 70/30 into train and test sets, and repeat the draw with eight seeds (42–49). Figure 3A reports b-fractions on the held-out cases, averaged over seeds; Table 8 gives per-cell sample sizes and values. Qwen/MuSiQue has no balanced-probe run and is omitted. The held-out sets are small (2–12 cases per group and seed), so individual cells are noisy.

Embedding-layer baseline. At the embedding layer, the readout position holds the same final prompt token across examples, so the probe returns a constant set by its intercept. The layer-0 b-fraction of 58% is the share of fits with a positive intercept, averaged across cells and seeds. We assess sample-dependent information through the subsequent held-out F/C separation.

Table 8: **Balanced probe per cell** (Figure 3A). n: train/test cases per group and seed. Layer 0: share of resamples whose constant early output favors b (identical for F and C). 50% / final: held-out b-fraction at 50% depth and at the last layer, averaged over seeds. Averaging instead over the first half of the layers, which includes the constant embedding-layer output, gives different values (e.g., C on Llama/HotpotQA 71.4% vs. 41%, on Llama/2Wiki 39.1% vs. 3%).

Model	Dataset	n	Layer 0	F 50%	C 50%	F final	C final
Llama	2Wiki	10/4	63%	94%	3%	88%	13%
	Hotpot	24/10	88%	84%	41%	65%	39%
	MuSiQue	6/2	50%	44%	38%	63%	44%
Mistral	2Wiki	11/5	13%	93%	3%	80%	0%
	Hotpot	28/12	100%	100%	54%	85%	42%
	MuSiQue	6/2	75%	56%	56%	63%	81%
Qwen	2Wiki	12/5	25%	83%	23%	85%	20%
	Hotpot	15/6	50%	69%	29%	69%	35%
MACRO MEAN (8 cells)			58%	78%	31%	75%	34%

Original analyses. The original trajectory probe uses 70% of the F and recollapse cases for training, with the same α and target defined above. Its peak-layer b-fractions in Table 7 are evaluated on all valid recollapse cases. The geometric and causal results in Tables 1–3 and Figure 3B–C retain the weights from the original analyses; they are separate from the balanced-probe evaluation.

E Geometric Independence: Per-Cell Results

Table 9 reports the cosine alignment between $\mathbf{V}_\ell$ and $\hat{\mathbf{D}}$ for each cell, alongside the random baseline.

In 8/9 cells the z-score is negative, meaning $\mathbf{V}_\ell$ is *less* aligned with $\hat{\mathbf{D}}$ than a random direction. The single positive z-score (Llama/HotpotQA, $z = +0.09$) is negligibly above zero. The pattern holds across all three models and all three datasets.

F Alternative Intervention Operators

Section 5.1 tests causal influence with conditional residual suppression at λ=1 (Eq. 3). To check that the asymmetry between $\mathbf{V}$ and $\mathbf{D}$ does not depend on this operator, we apply three further levers to the collapse run at the late layers.

Additive V. The additive $\mathbf{V}$ intervention pushes the hidden state along the knowledge direction, $\mathbf{h}_\ell \leftarrow \mathbf{h}_\ell + \alpha\,\mathbf{V}_\ell$ with $\alpha = +2$.

Evidence-difference patching. We perturb the collapse-run state $\mathbf{h}_\ell^C$ along the activation shift that the counterfactual evidence itself induces, $\mathbf{h}_\ell^C \leftarrow \mathbf{h}_\ell^C + \beta\,\Delta\mathbf{h}_\ell^{\text{evid}}$ with $\beta = -2$, where $\Delta\mathbf{h}_\ell^{\text{evid}} =$

Table 9: $|\cos(\mathbf{V}, \hat{\mathbf{D}})|$ **per cell.** n: number of collapse pairs. d: hidden dimension. Random: mean $|\cos(\mathbf{r}, \hat{\mathbf{D}})|$ over 1,000 random unit vectors per pair.

Model	Dataset	n	d	$\lvert\cos(\mathbf{V}, \hat{\mathbf{D}})\rvert$	Random	z
Llama	2Wiki	80	4096	.0078	.0125	-0.50
	Hotpot	50	4096	.0133	.0124	$+0.09$
	MuSiQue	6	4096	.0024	.0126	-1.07
Mistral	2Wiki	80	4096	.0102	.0125	-0.24
	Hotpot	65	4096	.0119	.0125	-0.05
	MuSiQue	9	4096	.0118	.0124	-0.07
Qwen	2Wiki	28	3584	.0063	.0133	-0.70
	Hotpot	41	3584	.0088	.0133	-0.44
	MuSiQue	2	3584	.0063	.0134	-0.69
MACRO MEAN		361		.0088	.0128	-0.41

$\mathbf{h}_\ell(s^-) - \mathbf{h}_\ell(s^0)$ is the layer-ℓ hidden-state difference between the counterfactual context s^- and a baseline context s^0 without the counterfactual evidence.

Null-δ patch. As a clean control, we apply the same additive form with a scale-matched difference vector that carries no evidence, so any nonzero effect would flag an artifact of the patching operation rather than of the evidence signal.

Table 10: **The V/D asymmetry holds across intervention operators.** Flip rate: sign-flip for the three new operators; the suppression rows repeat the margin sign-flip rate and behavioral rescue of Table 2 for reference.

Operator	Direction	Flip rate	Answer rescue
Suppression, λ=1	$\mathbf{V}$	0.0%	6.4%
Suppression, λ=1	$\mathbf{D}$	52.7%	15.3%
Additive, α=+2	$\mathbf{V}$	1.3%	4.4%
Evidence-δ patch, β=−2	$\Delta\mathbf{h}^{\text{evid}}$	0.4%	3.9%
Null-δ patch (clean control)	scale-matched null	0.0%	0.0%

Table 10 reports the results. All three levers stay far below the 52.7% that $\mathbf{D}$-only suppression reaches. The $\mathbf{V}$ and evidence-difference interventions therefore have substantially smaller effects than the $\mathbf{D}$ intervention. For the three main models, $\mathbf{V}$ residual suppression yields 0% sign-flip across the tested strengths (Table 11).

G Full Dose-Response Sweep

Table 11 extends the main-text dose-response table to include $\lambda \in \{0.5, 1.5\}$ and reports all four metrics. The intervention window starts at the evidence-peak layer before recollapse and continues through the final transformer block. For each case, the $\mathbf{V}$ direction is taken at that peak layer and held fixed across the window.

G.1 Per-cell results at λ=2.0

H Full Control Conditions

Table 13 reports all control conditions alongside the real intervention, at λ=1.0. Each control ablates one component of the intervention's causal logic. These controls and Table 19 use the original control evaluation, with baselines rerun in that evaluation. Rescue counts non-skipped runs that change the baseline output from a to b, divided by all evaluated pairs. The dose sweep counts all post-intervention b matches and reports its summary rates separately in Table 11.

Table 11: **Full dose-response: $\hat{\mathbf{D}}$-axis vs. $\mathbf{V}$-axis.** Macro-averaged across 9 cells, 360 collapse cases. Rescue uses normalized answer matching as in Table 3.

	$\hat{\mathbf{D}}$-axis				$\mathbf{V}$-axis
λ	$P(b)>P(a)$	Flip	Rescue	s^+ pres.	Flip
0.5	19.8%	11.0%	8.2%	62.7%	0.0%
1.0	61.4%	52.7%	15.3%	48.2%	0.0%
1.5	92.2%	83.4%	33.2%	38.4%	0.0%
2.0	96.9%	88.2%	46.4%	37.2%	0.0%
3.0	98.7%	89.9%	55.8%	33.8%	0.0%

Table 12: **$\hat{\mathbf{D}}$-axis suppression at λ=2.0, per cell.**

Model	Dataset	n	$P(b)>P(a)$	Flip	Rescue
Llama	2Wiki	80	97.5%	95.0%	53.7%
	Hotpot	50	94.0%	86.0%	70.0%
	MuSiQue	6	100%	100%	33.3%
Mistral	2Wiki	79	98.7%	96.2%	32.9%
	Hotpot	65	96.9%	90.8%	43.1%
	MuSiQue	9	88.9%	88.9%	33.3%
Qwen	2Wiki	28	96.4%	89.3%	42.9%
	Hotpot	41	100%	97.6%	58.5%
	MuSiQue	2	100%	50.0%	50.0%

Interpretation. *Wrong-direction* and *no-op* yield 0% rescue. *Off-random*, *off-early*, and *off-τ_{up}* yield at most 0.1% rescue. *Off-late* yields 27.5% positive post-intervention margins, compared with 61.2% for the full window. *τ-down3* applies the correct direction around recollapse onset and achieves only 8.8%, confirming that a narrow window at the start of the override is insufficient.

I Where Prior Restoration Acts: Layer-Range Ablations

We characterize the prior-restoring mechanism on 229 recollapse examples (Appendix A.6) drawn from the five-dataset sweep (Appendix N). For each example and layer we take the signed projection $s = \langle \Delta \mathbf{h}_\ell, \mathbf{D} \rangle$ of Eq. 4; a layer with $s < 0$ carries a negative-$\mathbf{D}$ update, and the negative mass is the total magnitude of these negative projections.

Distributed but late. Each example carries a negative-$\mathbf{D}$ update at 17.9 layers on average, so the mechanism is distributed rather than confined to one place. It is nonetheless late and cumulative (Table 14): 89.7% of the negative mass sits in the late third of the layers, and the single largest negative layer falls there in 99.6% of examples.

Equal-count control. Table 15 restricts the suppression of Eq. 4 to k layers within a given range. The single largest negative layer on its own moves only 13.5% of cases to $P(b)>P(a)$, at 7.0% sign-flip. Equal-budget suppression in the early or middle third stays at the no-intervention baseline, whereas late and all-negative suppression drive the rescue, so the effect is late-specific rather than driven by the number of edited layers. Across cells, the cumulative negative mass predicts the margin movement (Spearman $\rho = 0.90$ / 0.75 / 0.71 for Llama / Mistral / Qwen) more strongly than the number of negative layers ($\rho = 0.62$ / 0.50 / -0.18).

The plateau at high λ. Increasing λ from 2 to 3 raises $P(b)>P(a)$ from 96.9% to 98.7% and answer rescue from 46.4% to 55.8% (Table 11). $\mathbf{V}$-axis sign-flip stays at 0.0% for every λ.

Table 13: **All control conditions** (λ=1.0, macro-averaged across 9 cells). ΔM: change in logit margin. $P(b)>P(a)$: fraction with positive post-intervention margin. Rescue: fraction with an a-to-b answer transition, using normalized answer matching.

Condition	Description	ΔM	$P(b)>P(a)$	Rescue
Real	Correct direction, correct layers	$+9.15$	61.2%	7.0%
Wrong-dir	Random direction orthogonal to $\hat{\mathbf{D}}$	-0.04	8.2%	0.0%
No-op	No modification	0.00	8.5%	0.0%
Off-random	Random non-window layers, up to the full window size	$+0.40$	10.2%	0.0%
Off-early	Correct direction, first 3 model layers	$+0.07$	8.5%	0.0%
Off-late	Correct direction, last 3 layers	$+5.84$	27.5%	2.4%
Off-τ_{up}	3 layers centered on evidence-uptake onset	$+0.04$	8.8%	0.1%
τ-down3	3 layers centered on recollapse onset	$+0.24$	8.8%	0.1%

Table 14: **Negative-D updates concentrate in the late third** (229 recollapse examples).

Layer range	Share of negative mass	Largest negative layer in range
Early third	3.0%	0.4%
Middle third	7.3%	0.0%
Late third	89.7%	99.6%

J Behavioral Specificity

J.1 s^+ preservation

When we apply the intervention to s^+-context cases (where the model already correctly outputs a), the intervention should ideally have no effect. Table 16 reports the preservation rate: the fraction of s^+ cases that still output a after intervention.

The intervention never flips a correct s^+ answer to b (0% broken-to-b across all 9 cells). Preservation rates vary across cells (34–100%), indicating that it can change a correct answer even when the output does not switch to b.

J.2 O-group control

On O-group cases (those that output neither a nor b before intervention and do not exhibit the collapse trajectory), the intervention produces near-zero rescue:

The intervention acts selectively on the collapse mechanism; it does not indiscriminately push the model toward b.

J.3 Off-target effects on unrelated prompts

The controls above stay within the knowledge-conflict data. To quantify the intervention's off-target cost, we transfer each conflict-derived $\mathbf{D}$ and layer window to 166 matched held-out clean prompts (13 cells) and measure how often a correct answer becomes incorrect (Table 18).

The high-strength intervention harms unrelated questions. The damage is semantic answer corruption rather than gibberish: malformed-output rates stay at 1.8% while the gold-answer log-probability falls. A random orthogonal direction of the same magnitude is nearly harmless, with 1.8% flips

Table 15: **Equal-count control** (229 recollapse examples). k: number of edited layers.

Edited range	k	$P(b)>P(a)$	Sign-flip	Exact rescue
Late (single largest)	1	13.5%	7.0%	6.1%
Early	5	6.6%	0.4%	3.9%
Middle	5	7.9%	1.3%	4.4%
Late	5	48.0%	41.5%	14.4%
Late	10	55.5%	48.9%	15.3%
All negative	all	79.0%	72.5%	20.1%

Table 16: s^+ **preservation under real intervention** (λ=1.0). Rates are conditional on a rerun baseline output of a in the control evaluation; the dose sweep reports unconditional rates. Broken-to-b: fraction of s^+ cases that flip to b after intervention.

Model	Dataset	Preserve a	Broken to b
Llama	2Wiki	34.3%	0.0%
	Hotpot	76.0%	0.0%
	MuSiQue	50.0%	0.0%
Mistral	2Wiki	42.6%	0.0%
	Hotpot	84.0%	0.0%
	MuSiQue	80.0%	0.0%
Qwen	2Wiki	53.8%	0.0%
	Hotpot	63.2%	0.0%
	MuSiQue	100%	0.0%
MACRO MEAN		64.9%	0.0%

at λ=2, while a shuffled answer-contrast $\mathbf{D}$ causes 15.1% flips. These off-target effects limit the intervention's use as a correction method.

K Per-Cell Intervention Results

Table 19 reports the real intervention results at λ=1.0 for each of the nine cells.

Qwen shows the strongest intervention response at λ=1.0 ($\Delta M > 12$ in all cells), consistent with its smaller hidden dimension ($d = 3584$ vs. 4096). Mistral shows the weakest answer-flip rates at λ=1.0 but reaches comparable levels at λ=2.0 (Table 12).

L Ceiling Annotation Details

L.1 Categorization

The 108 remaining non-b outputs at λ=2.0 in Table 4 are grouped into the four categories below.

L.2 Category definitions

- **B_related** (80 cases, 74.1%): tokens semantically related to b but not valid answers under any matching criterion. Includes hypernyms (category name instead of specific entity), co-occurring entities (another entity from the same passage), attributes of b (nationality, title), and topically related terms from the same domain.
- **A_format** (12 cases, 11.1%): structural tokens such as newlines, punctuation, or repeated preamble fragments, indicating disruption of generation format rather than answer selection.
- **B_alias** (14 cases, 13.0%): borderline near-matches that fall outside the relaxed matching criteria but are arguably related to b.
- **A_residual** (2 cases, 1.9%): tokens associated with the prior answer a (e.g., a related entity or attribute of a), indicating incomplete suppression in a small number of cases.

Table 17: O**-group rescue rate under real intervention** (λ=1.0).

Model	Dataset	O-group rescue
Llama	2Wiki / Hotpot / MuSiQue	2.5% / 0.0% / 0.0%
Mistral	2Wiki / Hotpot / MuSiQue	0.0% / 0.0% / 0.0%
Qwen	2Wiki / Hotpot / MuSiQue	0.0% / 0.0% / 0.0%
MACRO MEAN		0.3%

Table 18: **Off-target effects on unrelated clean prompts** (166 prompts, 13 cells).

Unrelated clean prompts	λ=2	λ=3
Correct→incorrect flips (pooled / macro)	17.5% / 24.3%	42.2% / 52.0%
Output changed	44.0%	84.3%
Gold-answer log-prob drop	2.47	20.05

L.3 Examples

Table 20 provides representative examples from each category.

M Decomposing the Rescue Ceiling

The 96.9% and 46.4% figures of Section 6 measure different stages. The former is a pairwise b-over-a preference; the latter measures normalized answer-match rescue. Generating the answer also requires selection among the full vocabulary and successful continuation. We examine these failure modes on the 229 recollapse cases of Appendix I.

Full-vocabulary audit. Under $\mathbf{D}$-only suppression (Table 21), b's first token is the full-vocabulary top-1 in 91.7% of cases. Only 7.9% of cases (18/229) have a competitor outranking b at the first step, whereas 80/229 already place b's first token at top-1 yet still miss strict exact generation. Continuation and matching account for many remaining failures: semantic rescue is 81.2%, while strict exact-match rescue is 56.8%.

A second, fine-grained direction. Using the target answer b and the post-intervention error token $\hat{y}$, we define $\mathbf{G}$ as the component of $\mathbf{u}_b - \mathbf{u}_{\hat{y}}$ orthogonalized against $\mathbf{D}$. Added on top of $\mathbf{D}$, it raises first-token top-1 to 98.7% and strict exact rescue to 61.1%, a gain of 4.4 percentage points (Table 21). The exact-rescue change varies across cells: Mistral/2Wiki gains 9.6 points, while Mistral/CounterFact loses 15.8 points. $\mathbf{G}$ on its own rescues only 4.4% of cases, far below $\mathbf{D}$. A second direction improves first-token selection more than complete answer generation.

N Extended Settings: Scale, Benchmarks, and Long-Form Generation

This section details the experiments summarized in Section 8: a 3-model × 5-dataset sweep that adds single-hop CounterFact [Meng et al., 2022] and the knowledge-conflict benchmark ConflictQA [Xie et al., 2024] to the three multi-hop datasets, a 27B replication, and two long-form output settings. Table 22 summarizes the task-format breadth.

N.1 Larger model: Qwen3.5-27B

We replicate the analysis on Qwen3.5-27B, a model roughly three to four times the size of the main ones, on CounterFact (Table 23). The model exhibits late-layer reversal, near-orthogonality, and stronger effects of $\mathbf{D}$ intervention. At λ=3, $\mathbf{V}$-only intervention produces 20.8% margin sign-flip and 0% exact rescue.

Table 19: **Conditional residual suppression per cell** (λ=1.0). Flip is the fraction with positive post-intervention margin among non-skipped runs whose rerun baseline output is a; $P(b)>P(a)$ is the same fraction over all non-skipped runs. The denominators differ, so Flip can exceed $P(b)>P(a)$ (e.g., Llama/2Wiki). Rescue follows the control-evaluation definition in Appendix H.

Model	Dataset	n	ΔM	$P(b)>P(a)$	Flip	Rescue
Llama	2Wiki	80	+6.5	55.0%	57.1%	8.7%
	Hotpot	50	+6.1	62.0%	58.8%	18.0%
	MuSiQue	6	+6.6	66.7%	50.0%	0.0%
Mistral	2Wiki	80	+7.9	31.6%	30.6%	6.2%
	Hotpot	65	+7.5	33.8%	30.2%	3.1%
	MuSiQue	9	+6.1	44.4%	0.0%	0.0%
Qwen	2Wiki	28	+13.0	71.4%	63.2%	7.1%
	Hotpot	41	+15.5	85.4%	84.4%	19.5%
	MuSiQue	2	+12.8	100%	100%	0.0%

Table 20: **Representative examples of non-b outputs by category.**

Category	Target b	Model output	Relationship
B_related	Jonathan Stark	tennis player	Hypernym
B_related	Kiran Rao	Aamir Khan Films	Co-occurring entity
B_related	Henri Leconte	French Open	Topically related
A_format	Jonathan Stark	\n	Newline token
A_format	Henri Leconte	The answer is:	Preamble fragment
B_alias	Jonathan Stark	J. Stark	Near-match
A_residual	Henri Leconte	Danger: Diabolik	Prior entity

N.2 Long-form question answering

Task. We reuse the same a/b items but require a three-sentence answer written only from the passages: sentence one states the answer, sentence two the first supporting fact, and sentence three the second fact together with why the two facts imply the answer. In the worked example of Appendix A.3, the prior-consistent response A answers *Danger: Diabolik* and supplies both directors with their true birth years, while the evidence-consistent response B answers *Charge It To Me*, built from the counterfactual years. This keeps the original a/b conflict and lengthens the output from roughly 1–3 tokens to 30–80 tokens. The gold rationale is generated from the supporting chain.

Per-step intervention. We apply **D**-suppression either at the first answer position alone or at every position, and report whether the final answer and the full response are consistent with the evidence. Table 24 reports five selected cells. Every-position suppression improves final-answer accuracy in these cells, while full-response consistency remains lower. Elsewhere in the sweep, Qwen/MuSiQue and Qwen/ConflictQA remain at 0% answer accuracy under both interventions; Mistral/MuSiQue remains at 16.7%.

N.3 Sentiment paragraphs

The sentiment task applies the same design to a different signal. For real entities with a stable no-context polarity a, s^+ supplies reviews matching that polarity and s^- supplies opposite-polarity reviews matched in length and syntax, and the model must produce a fixed four-sentence paragraph: an overall assessment, two evidence-based reasons, and a recommendation. Every-position full-answer consistency is 41.7% / 100% / 91.7% against an evidence baseline of 91.7% / 91.7% / 83.3% for Llama / Qwen / Mistral: higher than the baseline for Qwen and Mistral, and 50 points lower for Llama. The baseline may not be scored with the same full-answer criterion, so we treat this comparison as indicative only.

Table 21: **Full-vocabulary audit** (229 recollapse cases). $\mathbf{D} + \mathbf{G}$: the fine-grained direction $\mathbf{G}$ added on top of $\mathbf{D}$.

Metric	$\mathbf{D}$-only	$\mathbf{D} + \mathbf{G}$
b first token is full-vocabulary top-1	91.7%	98.7%
Semantic rescue	81.2%	87.3%
Strict exact-match rescue	56.8%	61.1%

Table 22: **Task-format breadth.** CounterFact and ConflictQA ranges are over the three models.

Setting	Type	Key result
CounterFact	single-hop counterfactual	evidence decodable 95.0–97.0%; all-negative sign-flip 63.2–100%
ConflictQA	dedicated knowledge-conflict benchmark	evidence decodable 96.6–97.0%
Long-form QA	multi-step generation	every- vs. first-position accuracy 70.0–91.7% vs. 5.3–30.0% in five selected cells; three other cells unchanged
Sentiment	different task type	every-position full consistency 41.7 / 100 / 91.7% vs. evidence baseline 91.7 / 91.7 / 83.3% (Llama / Qwen / Mistral); lower for Llama

O Provenance of the Prior Answer

We compare checkpoints to study when the prior answer a is available and use causal tracing to locate where it influences the output. Following standard practice in the knowledge-conflict literature [Longpre et al., 2021, Xie et al., 2024, Kortukov et al., 2024], we treat the model's closed-book prediction as its parametric answer.

Base vs. instruction-tuned checkpoints. The prior answer a is not introduced by instruction tuning alone: both the base pretrained checkpoint and the instruction-tuned model can answer a in the absence of contextual evidence. Table 25 reports their agreement.

Causal tracing. Following Meng et al. [2022], we copy one component's activation at a time from a matched reference run into the collapse run and measure how much of the evidence-over-prior margin that single restoration recovers; the components whose restoration recovers the most mark the causal locus. Causal tracing places the dominant contributions mostly in late layers (Table 25).

Table 23: **Larger-model replication** (Qwen3.5-27B, CounterFact; intervention rows use n=48 cases).

Measure	Value
Late-layer reversal in collapse cases	61 / 61
Evidence decodable at mid layers	95.1%
$\lvert\cos(\mathbf{V}, \mathbf{D})\rvert$ vs. random	0.006 vs. 0.011 ($z = -0.61$)
$\mathbf{D}$-only, λ=2: sign-flip / $P(b)>P(a)$ / exact rescue	100% / 100% / 68.8%
$\mathbf{V}$-only, λ=3: sign-flip / exact rescue	20.8% / 0%

Table 24: **Long-form QA in five selected cells.** Answer accuracy after $\mathbf{D}$-suppression at the first position or every decoding position, and full-response consistency under every-position suppression.

Cell	First answer	Every answer	Every full
Llama / HotpotQA	30.0%	70.0%	5.0%
Llama / CounterFact	25.0%	91.7%	75.0%
Qwen / HotpotQA	23.1%	76.9%	15.4%
Mistral / 2Wiki	15.0%	70.0%	0.0%
Mistral / CounterFact	5.3%	84.2%	42.1%

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction accurately state the paper's main contributions.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The paper discusses limitations in Section 8.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.

Table 25: **Provenance of the prior.** Agreement on a between the base and instruction-tuned checkpoints in the closed-book setting, and the components with the largest causal-tracing recovery. Resid denotes the residual stream. Checkpoint comparisons cover five datasets for Llama and Qwen and four for Mistral.

Model	Base ∧ instruct agreement	Dominant causal components
Llama	66.7–100%; 100% on 3/5 datasets	resid 30/31, MLP 31
Qwen	100% on all 5 datasets	MLP 12, resid 23/25, MLP 25/26
Mistral	98.1–100%	MLP 14/20/31, resid 20/28

- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [N/A]

   Justification:

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The paper discloses the information needed to reproduce the main experimental results, including benchmark construction, public/private splits, model baselines, input histories, evaluation metrics, training objectives, candidate reranking procedure, and implementation details. Additional appendix material reports benchmark statistics, hyperparameters, robustness analyses, and compute resources needed to rerun the main experiments.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.

- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The submission includes code and supporting evaluation materials sufficient to reproduce the main experimental results.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).

- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: The paper specifies the experimental settings needed to understand and reproduce the reported results, including benchmark definitions, public/private splits, baselines, input histories, evaluation metrics, and implementation details. Additional appendix material provides benchmark statistics, training objectives, hyperparameters, candidate-set sizes, and compute details.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: The paper reports statistical uncertainty for the main experimental comparisons using bootstrap 95% confidence intervals over per-example evaluation outputs. The appendix explicitly states that these intervals measure evaluation-sample robustness rather than training-run variance, since repeated full multi-seed training runs were not available.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
   - If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

Justification: The appendix reports the compute resources used for the main experiments, including GPU count, GPU memory, numerical precision.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

Answer: [Yes]

Justification: The research conforms to the NeurIPS Code of Ethics. The paper uses benchmark and public-history data under an anonymized evaluation setting, does not report personally identifying user information, and explicitly discusses privacy and misuse risks associated with recoverable user representations.

Guidelines:

- The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [N/A]

Justification: This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none of which we feel must be specifically highlighted here.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.

- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification:

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: All models and datasets are public and under the MIT License.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification:

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.

- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification:

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification:

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification:

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.