Where's the Plan? Locating Latent Planning in Language Models with Lightweight Mechanistic Interventions

Nicole Ma [*1] **Nick Rui** [*1]

Abstract

We study *planning site formation* in language models—*where* internal representations of structurally-constrained future tokens form during the forward pass, and whether they causally drive generation. Using rhyming-couplet completion as a clean test of forward-looking constraint, we apply two lightweight methods (linear probing and activation patching) across Qwen3, Gemma-3, and Llama-3 at more than ten scales. Probing shows that future-rhyme information is linearly decodable at the line boundary, with signal that strengthens with scale in all three families. Activation patching reveals that only Gemma-3-27B causally relies on this encoding, exhibiting a *handoff* in which the causal driver migrates from the rhyme word to the line boundary around layer 30. Every other model we test conditions on the rhyme word throughout generation, with near-zero causal effect at the line boundary despite strong probe signal. We localize the Gemma-3-27B handoff to five attention heads through two-stage path patching that recover $\sim$90% of the rhyme-routing capacity at the newline.

1. Introduction

Autoregressive language models generate text one token at a time, yet routinely produce outputs requiring long-range structural coherence. Rhyming couplets, for example, require that tokens generated late in the sequence stand in a precise phonological relation to a stimulus word from much earlier. This raises a natural question: do models form internal representations of future outputs that causally shape generation, entirely invisible to behavioral evaluation? We call this *latent planning*. Unlike chain-of-thought reasoning (Wei et al., 2022), where intermediate steps are directly observable, latent planning occurs entirely within a model's hidden activations. If models are planning in ways invisible to external observers, standard behavioral evaluations may systematically underestimate this capacity, making it both a scientific and a safety-relevant problem (Pfau et al., 2024; Hao et al., 2024).

Neural networks trained on next-step prediction can develop internal planning representations in structured domains. McGrath et al. (2022) find that chess concepts emerge in AlphaZero's internal layers; Li et al. (2022) show that transformers trained on Othello develop board-state representations, and Nanda et al. (2023) show that these representations are linear and causally manipulable; and Jenner et al. (2024) provide mechanistic evidence of learned look-ahead in a chess-playing network. These results motivate the question of whether language models elicit analogous mechanisms during open-ended generation. Prior work on large language models establishes that planning-compatible information exists in some models (Maar et al., 2026; Hanna & Ameisen, 2026; Dong et al., 2025; Pochinkov et al., 2025), but does not address a more specific question: *where exactly does planning information reside during the forward pass, and does it move?* We call this *planning site formation*.

Investigating planning sites rigorously requires both encoding evidence (what information is present) and causal evidence (what information is used). Probes assess what is encoded in hidden states (Hewitt & Liang, 2019; Burns et al., 2023), but sufficiently flexible probes can achieve high accuracy by memorizing labels rather than reflecting genuine representations (Hewitt & Liang, 2019). Causal tools such as activation patching (Meng et al., 2022; Wang et al., 2022) and steering vectors (Turner et al., 2023; Arditi et al., 2024) are needed to establish that encoded information actually influences downstream generation. The most expressive existing approach, training transcoders (Dunefsky et al., 2024) to build feature circuits (Marks et al., 2025; Ameisen et al., 2025), provides fine-grained circuit analysis but requires substantial compute (effectively a second training run), has so far been applied only to closed-source models such as Claude 3.5 Haiku (Lindsey et al., 2025), and does not straightforwardly scale to new open-source architectures. Work by Maar et al. (2026) uses steering vector

[*]Equal contribution. Authors listed by increasing height. [1]Stanford University. Correspondence to: {manicole, nickrui}@stanford.edu.

Accepted to the 2nd Workshop on Compositional Learning at ICML 2026, Seoul, South Korea. Copyright 2026 by the author(s).

interventions and finds most open-source models up to 30B parameters keep their planning sites at the last word token; a transcoder-based analysis by Lindsey et al. (2025) on Claude 3.5 Haiku finds evidence of a planning site migrating to the newline. These results motivate a scalable framework for studying planning site formation across model architectures and scales.

In this paper, we define two notions for surfacing evidence of latent planning: the weaker notion of planning-compatible representations (detectable via probing) and the stricter notion of causally active planning sites (established via activation patching). Using only linear probing and activation patching (requiring no transcoder training and far less data than steering vector approaches), we investigate planning site formation across three open-source model families at multiple scales (up to 70B parameters).

Probing reveals that planning-compatible representations emerge at the newline token with scale across all three families, yet their strength and layer profile vary substantially across architectures. Activation patching reveals that only Gemma-3-27B forms a causally active planning site at the newline, exhibiting an information routing handoff in which causal influence migrates from the last word token to the newline around layer 30. All other models, including all Qwen3 sizes up to 32B and all Llama-3 sizes up to 70B, condition on the last word token throughout generation despite encoding planning-compatible representations at the newline. We further localize the handoff in Gemma-3-27B to a sparse set of five attention heads in layers 28 and 30, identified via attention weight ranking and simultaneous head patching. This suggests that planning site formation is a distinct emergent phenomenon tied to specific model scale and architecture, rather than a general consequence of strong planning-compatible representations.

2. Setup and Notation

In this paper, we study rhyming couplet generation as an example of latent planning. A rhyming couplet is a two-line poem where the last word of the first line, denoted r_1, rhymes with the last word of the second line, denoted r_2. We task language models to complete rhyming couplets (i.e., given context containing r_1, generate a completion where r_2 rhymes with r_1). For example, given the prompt `"A rhyming couplet:\nShe felt a sudden sense of fright,\n"`, the model should produce a completion such as `"and hoped that dawn would end the night.\n"`, where r_1 = `fright` and r_2 = `night`.

Consider an autoregressive transformer language model with L layers and hidden dimension d. Let $\mathbf{h}_{\ell,i} \in \mathbb{R}^d$ denote the hidden state vector after the ℓ-th transformer block at position i. We adopt a relative positioning scheme centered around the newline token `\n` ending the first line of the couplet. We refer to this token as being at position 0, with position i denoting the token i steps before (negative) or after (positive) the newline.

We say (i, ℓ) contains a *planning-compatible representation* if the generated r_2 can be decoded from $\mathbf{h}_{\ell,i}$ via probing substantially better at position i than at other positions. This signals that information about the future token or rhyme scheme is disproportionately encoded at (i, ℓ) relative to other hidden states. As a stricter definition, we say (i, ℓ) is a *causally active planning site* if replacing $\mathbf{h}_{\ell,i}$ during generation with the hidden state from a run targeting a different rhyme substantially redirects output toward that rhyme. This shows that rhyme information located at (i, ℓ) is causally used in generation.

3. Probing for Planning-Compatible Representations

To investigate planning-compatible representations, we train linear probes to predict future tokens from hidden states. Let $\mathcal{V}$ be the vocabulary (set of all tokens) and $\Delta\mathcal{V}$ the probability simplex over $\mathcal{V}$. Each linear probe is a parameterized function $f_{(W,b)}(\mathbf{h}) : \mathbb{R}^d \to \Delta\mathcal{V}$ where $f_{(W,b)}(\mathbf{h}) = \text{softmax}(W\mathbf{h} + \mathbf{b})$. Probes are trained by minimizing cross-entropy loss. Probe parameters are optimized with AdamW (Loshchilov & Hutter, 2019) with learning rate 10^{-4}, weight decay 10^{-3}, batch size 32, for 10 epochs. We report Wilson 95% confidence intervals (Wilson, 1927) on every probe accuracy, computed from the size of the held-out evaluation set.

3.1. Probing general text as a negative control

We first verify that planning-compatible representations are task-specific and do not occur in general text generation. We randomly sample 1,200 (1,000 train, 200 validation) token sequences from The Pile (Gao et al., 2021), greedily sample model completions while storing activations to build the probe training dataset. For each layer ℓ and various look-ahead distances k, we train a linear probe to predict the generated token at position $i + k$ from $\mathbf{h}_{\ell,i}$. We compare against a baseline unigram model (trained on the corpus of all model completions) to ensure probes are learning more than just token frequencies.

We find probe accuracy degrades monotonically with k and the $k = 8$ confidence band overlaps the unigram baseline across all layers in all three models (Figure 1), confirming that planning-compatible representations are not a generic property of the residual stream. This provides a negative control establishing that probe signal in Section 3.2 is significant.

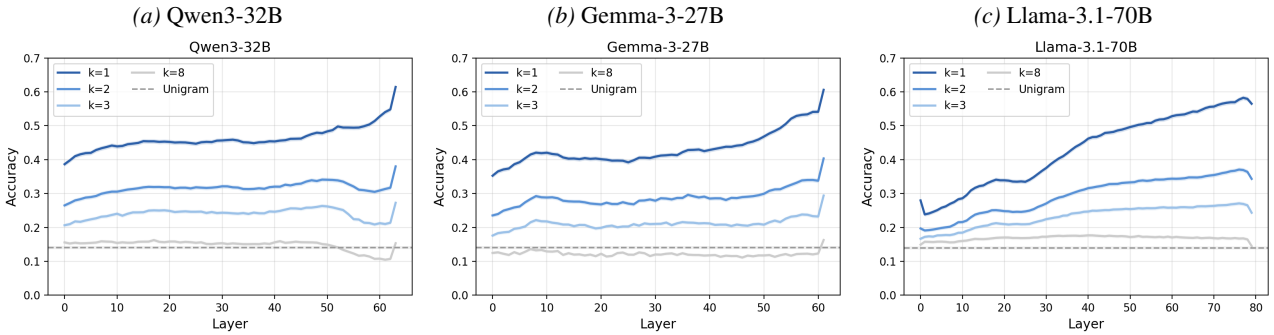


Figure 1. Top-5 accuracy of linear probes predicting k tokens ahead in general text (Pile). Wilson 95% CI bands are drawn but visually imperceptible: per-token $N{\approx}21{,}000$ gives a half-width of $\sim$0.005 at typical p, so the curves are essentially noise-free at this sample size. Accuracy degrades monotonically with k and the $k = 8$ curve overlaps the unigram baseline across all layers, confirming that planning-compatible representations are not a generic feature of the residual stream.

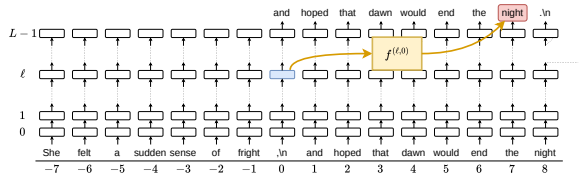


Figure 2. A linear probe $f^{(\ell,0)}$ trained to predict the rhyming token from activations at the newline position ($i = 0$).

3.2. Probing rhyming couplets

For rhyming couplet generation, we first synthetically generate 1,200 (1,000 train, 200 validation) rhyming couplets with Claude Sonnet 4.6 (Anthropic, 2026), strategically prompting for diversity of topics and rhyme schemes. We truncate the second line of each couplet, then greedily sample model completions for the second line, storing activations to build the probe training dataset. We train probes to predict the rhyming token r_2 from $\mathbf{h}_{\ell,i}$ (Figure 2), evaluating on raw accuracy and rhyme accuracy (as determined by the CMU Pronouncing Dictionary (Weide, 1993)) across layers and positions. The comparison is between probes trained on activations at the newline position ($i = 0$) and subsequent positions ($i > 0$). If a model constructs a planning-compatible representation at the newline before generation begins, the $i = 0$ probe should substantially outperform $i > 0$ probes. We also probe on hidden states at the last word token position,[1] which we expect to perform well since the last word directly decides the rhyme scheme of the couplet.

First, probe accuracies for predicting r_2 are substantially higher than for general-text continuation. Although the generated r_2 is on average 8 tokens past the newline (the last context token), the $i = 0$ couplet probe far outperforms the corresponding $k = 8$ Pile probe. This suggests that models selectively construct planning-compatible representations in response to the structural demands of rhyming couplet generation.

More importantly, the signal is concentrated at the last word and newline positions in specific layers (Figure 3): the $i \leq 0$ probes outperform the $i > 0$ probes by a wide margin, in contrast to the monotonic decay with look-ahead distance seen on general text.

Probes at the last word peak early and decay later, consistent with the lexical identity of r_1 being encoded from the earliest layers. The newline probes are more striking: they also perform well, hinting at latent planning at this position. Whereas the $i > 0$ probes follow a similar shape throughout, the $i \leq 0$ probes trace a qualitatively different curve across layers—evidence that the newline and last word undergo specialized computations rather than passively accumulating context.

Replicating this experiment on smaller Qwen, Gemma, and Llama models reveals that planning-compatible representations at the newline emerge with scale. For each model we measure the largest accuracy gap across layers between probes at the newline and probes at the first generated position; this gap grows with scale (Figure 4). Within Qwen3 and Llama-3, the gap CIs of the smaller models (Qwen3 0.6B–8B; Llama-3 1B–8B) all overlap zero, whereas the largest model in each family has a clearly nonzero gap. Gemma-3 shows a positive gap at every scale and the cleanest monotonic trend, rising from 0.11 at 1B to 0.38 at 27B. This pattern is consistent with planning-compatible representations being an emergent property of larger models.

3.3. Discussion and limitations of probing

The probing results reveal substantial cross-model and cross-scale diversity in planning-compatible representations at

[1] Position $i = -2$ in Gemma and $i = -1$ in Qwen and Llama (see Appendix A).

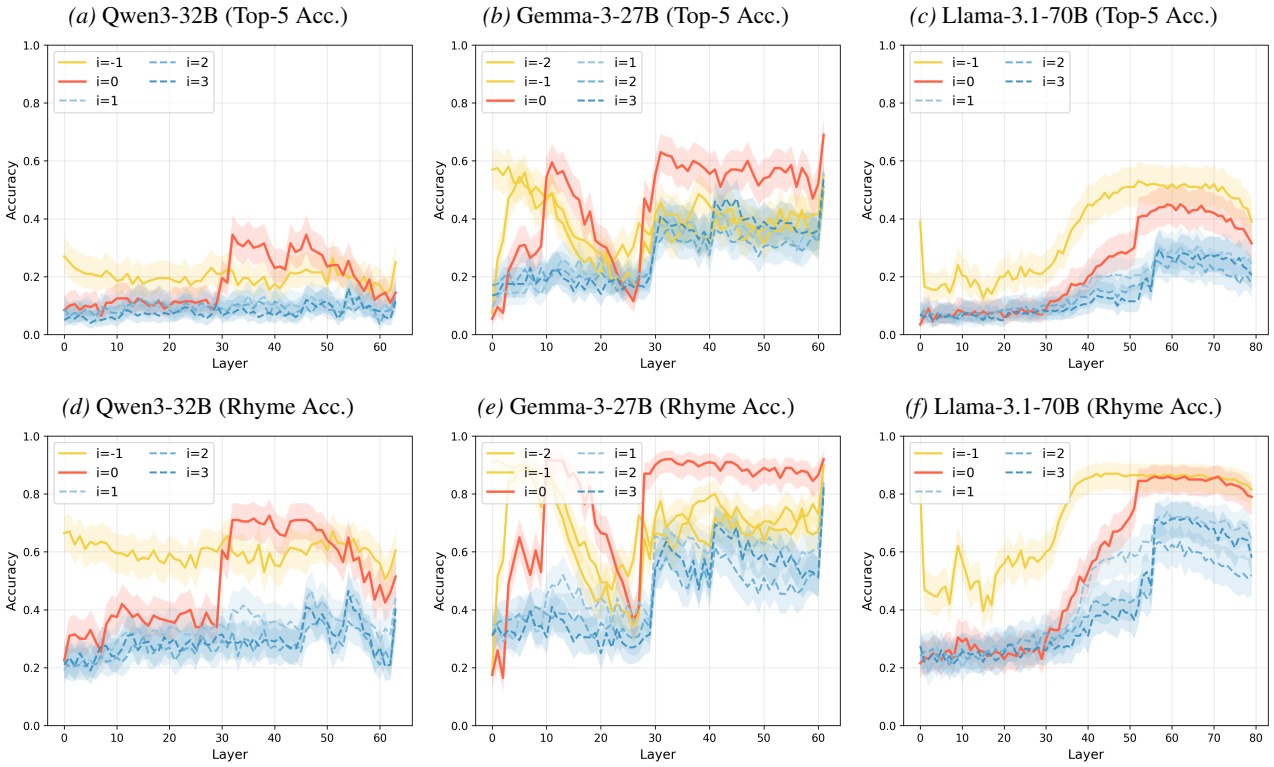


Figure 3. Top-5 and rhyme accuracy of linear probes trained to predict r_2 from hidden states at various layers and positions. Shaded bands are Wilson 95% CIs computed from $N = 200$ validation items. Probes at the last word ($i \leq -1$) and newline ($i = 0$) positions substantially outperform probes at subsequent generated positions ($i > 0$); the bands at the peak layers do not overlap with $i \geq 1$ bands in any of the three models, indicating that rhyme-relevant information is selectively concentrated at these structural positions.

both the last word and newline positions. However, several alternative explanations warrant consideration.

First, the elevated newline probe accuracy could reflect passive attention accumulation rather than active planning: if models attend strongly from the newline to the rhyme word, the newline hidden state will encode rhyme-relevant information without any downstream computation reading it out during generation. Second, rhyme accuracy is an imperfect metric because the CMU Pronouncing Dictionary does not cover every valid rhyme, potentially underestimating probe performance unevenly across rhyme families. Finally, although we prompted Claude for diverse subjects and rhyme schemes, the resulting dataset may carry nontrivial distributional biases that could inflate probe accuracy in ways difficult to detect.

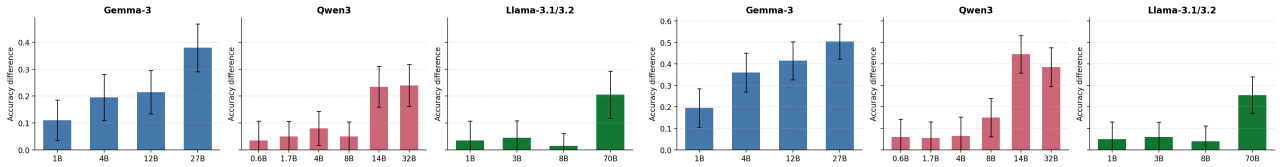
These considerations highlight two fundamental limitations of probing as a method. First, high probe accuracy establishes that rhyme-relevant information is linearly decodable at a position, but cannot establish that this information causally drives generation. Second, probe accuracy is sensitive to the distribution of the training data, meaning results could partially reflect dataset artifacts rather than genuine planning representations.

Activation patching addresses both limitations directly: by intervening on specific hidden states during generation and measuring the causal effect on output, it neither requires a probe training distribution nor conflates passive encoding with active deployment. We turn to activation patching in the next section to resolve these ambiguities.

4. Patching for Causally Active Planning Sites

While probing reveals that future token information is encoded in hidden states, it does not establish that this information is causally used by the model. In this section, we further investigate latent planning in rhyming couplet generation by employing intervention methods to test for causal influence directly. Given a prompt $\mathbf{x}^{(c)}$ whose first line naturally leads to a clean rhyme family $\mathcal{R}^{(c)}$, we intervene on the model's activations at various positions and investigate whether the generation of the second line shifts toward a different corrupted rhyme family $\mathcal{R}^{(r)}$. A successful intervention on a specific layer ℓ and position i provides causal evidence that information at $\mathbf{h}_{\ell,i}$ is causally involved in the model's latent planning downstream.

We apply activation patching at two positions: the last word



(a) Top-5 Accuracy Difference *(b)* Rhyme Accuracy Difference

Figure 4. Maximum accuracy gap across layers between probes at the newline ($i = 0$) and the first generated position ($i = 1$), plotted against model size. Black error bars are 95% CIs at the chosen peak layer (paired-difference Wald approximation; per-sample correctness was not stored, so the interval is conservative relative to the true paired CI). For Qwen3 sizes 0.6B–8B and Llama-3 sizes 1B–8B, the CI on the gap includes zero, while every Gemma-3 size and the largest Qwen/Llama models have CIs strictly above zero, supporting the emergent-property interpretation.

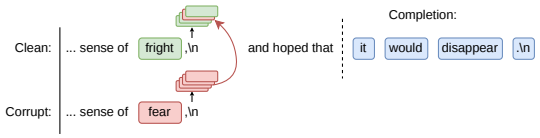


Figure 5. Activation patching: the hidden state $\mathbf{h}_{\ell,i}$ from a corrupted run is substituted into the clean run's forward pass at position i and layer ℓ. A successful patch redirects generation toward the corrupted rhyme family, providing causal evidence that (i, ℓ) is a planning site.

token (position $i = -1$ in Qwen3 and Llama-3, $i = -2$ in Gemma-3 due to its tokenization of the line-ending comma; see Appendix A) and the newline token ($i = 0$), sweeping across all layers individually. For each layer, we draw $N = 20$ stochastic samples per prompt pair; the main per-layer figures (Figure 6) average over 5 prompt pairs (N=100). Because the same word pairs are reused across layers and positions, the prompt pair is the natural unit of independence; we report 95% cluster bootstrap intervals (10,000 pair-level resamples) on every patching rate, and use a joint cluster bootstrap with shared pair indices when comparing two patching conditions on the same pairs.

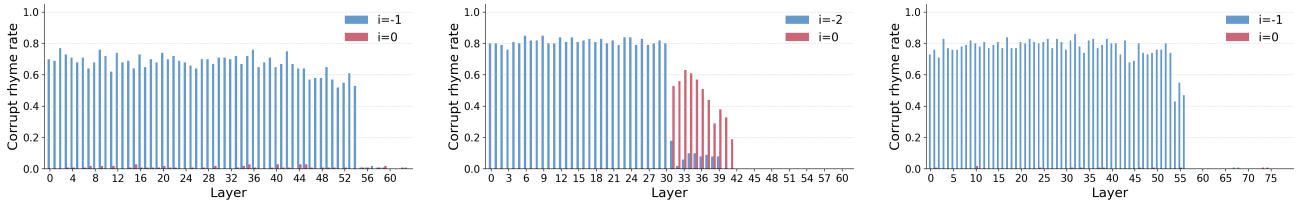
In Gemma-3-27B (Figure 6b), last-word patching is highly effective in early layers but drops sharply around layer 30, while newline patching rises simultaneously to a peak corrupt rhyme rate of 0.63 [95% CI 0.48, 0.78] at layer 33. We term this crossover the *representational handoff*: the causal locus migrates from the last word token to the newline, which becomes the primary read-out site for the phonological constraint. The wide CI reflects pair-to-pair variability (per-pair rates at layer 33 span 0.40–0.90), but the qualitative migration is consistent across pairs. By contrast, Qwen3-32B (Figure 6a) and Llama-3.1-70B (Figure 6c) show uniformly high last-word patching and near-zero newline patching across every layer, with non-overlapping CIs separating the two positions. These models condition on the last word token throughout generation. Swapping activation patching for a steering vector intervention (Turner et al., 2023; Maar et al., 2026) gives the same three-model picture: Gemma-3-27B hands off to the newline around layer 30, and Qwen3-32B and Llama-3.1-70B remain effective at the last word but flat at the newline. Steering needs far more data than patching to reach this conclusion (Appendix B). Per-layer patching results for every model size are in Appendix D.

4.1. Localizing the Handoff to a Sparse Set of Attention Heads

Having established that the planning site forms at the newline token in Gemma-3-27B, we ask whether the information routing handoff can be attributed to a small, identifiable set of attention heads. Single-head patching at the newline (replacing one head's output at a time within the planning layer range, layers 27–45) produced no measurable signal for any individual head, suggesting the representation is not localized to a single circuit element. We use attention weights as a proxy for which heads are most likely to route rhyme information from the last word token to the newline.

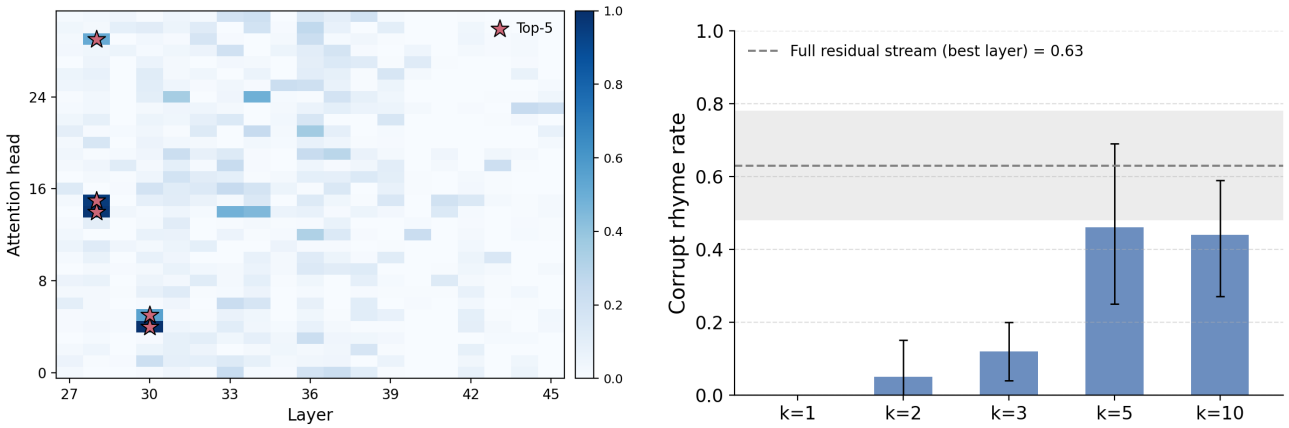
Specifically, we extract the attention weight from the newline token ($i = 0$) to the last word token ($i = -2$) for each head in layers 27–45 of both clean and corrupt forward passes, and rank heads by this weight. Figure 7a shows the resulting heatmap. Attention to $i = -2$ is highly concentrated in three heads that attend almost exclusively to the last word token from the newline—layer 30 head 4 (weight $\approx$0.99), layer 28 head 14 ($\approx$0.97), and layer 28 head 15 ($\approx$0.95). A second cluster of heads in layers 28–36 shows moderate attention weights (0.35–0.55).

We patch the top-k heads simultaneously, replacing their outputs at the newline with what they would have produced on the corrupt forward pass, and measure the resulting corrupt rhyme rate. To interpret these rates we compare against the strongest sub-component intervention available at the newline: replacing the entire residual stream at i=0 with corrupt's residual at the best single layer. Because the residual at i=0 is the sum of every attention head and MLP contribution up to that layer, this overwrites all of them at



(a) Qwen3-32B. *(b)* Gemma-3-27B. *(c)* Llama-3.1-70B.

Figure 6. Per-layer activation patching at the last word token and newline ($i = 0$) for the largest model in each family. Full results for all model sizes are in Appendix D.



(a) Attention weight, i=0 → i=−2. *(b)* Top-k head patching at the newline.

Figure 7. Localizing the planning site handoff in Gemma-3-27B to a sparse set of attention heads. **(a)** Attention weight from the newline token ($i = 0$) to the last word token ($i = -2$) across heads in layers 27–45. Red stars mark the top-5 heads by attention weight. **(b)** Corrupt rhyme rate when patching the top-k highest-attending heads simultaneously at the newline (5 prompt pairs × N=20). Black error bars are cluster-bootstrap 95% CIs over pairs. The dashed line and shaded gray band show the full-residual-stream peak at the best single layer (5-pair estimate 0.63 [0.48, 0.78]).

once and provides a strong reference for what any subset of components feeding i=0 recovers in practice (a subset patch can in principle exceed it when the full residual also injects conflicting context). We call this rate the *full-residual reference* (Section 4); for Gemma-3-27B it is 63%.

Simple top-k patching. Figure 7b shows that $k = 1, 2, 3$ yield near-zero corrupt rhyme rates. At $k = 5$ the rate jumps to 46%, which is 73% of the full-residual reference, and plateaus through $k = 10$. This means that patching just five attention heads at the newline reproduces about three-quarters of the rhyme-shifting effect of overwriting the entire residual at the best layer.

Two-stage path patching. To isolate the specific path $i = -2$ → head → i=0 → output, we apply two-stage path patching (Goldowsky-Dill et al., 2023; Wang et al., 2022). Stage 1: run the clean prompt with only the residual at $i = -2$ replaced by corrupt's, and cache each candidate head's output at i=0. Stage 2: forward the unmodified clean prompt and substitute those cached outputs at i=0 for the selected heads. Under this stricter intervention (Figure 8), the five heads recover a corrupt rhyme rate of 57% at $k = 5$, or 90% of the full-residual reference. This means that nearly all of the rhyme-routing capacity at the newline is concentrated in these five heads. The rate stays in the 44–57% range over $k = 5$–9, with $k = 5$ the highest individual point, and declines at $k = 10$ and $k = 15$ (47% and 32%). Random and comma-control head sets remain at zero across all k.

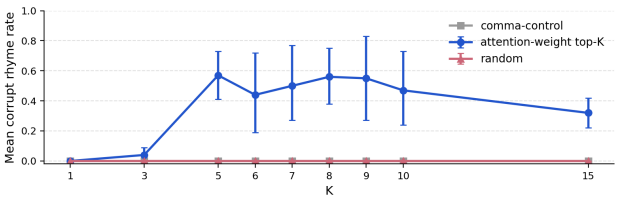


Figure 8. Two-stage path patching K-sweep on Gemma-3-27B. Attention-weight top-k peaks at $k = 5$ (57%, 90% of the full-residual reference) and declines at $k = 10, 15$. Comma-control and random head sets stay at zero. Error bars are 95% cluster-bootstrap CIs over prompt pairs.

The top five heads are (layer 30, head 4), (layer 28, head 14), (layer 28, head 15), (layer 30, head 5), and (layer 28, head 29). The analogous top-k MLP patches at i=0 yield zero corrupt rhyme rate at every k (Wilson 95% upper bound ≤ 0.04, N=100), confirming that the handoff is mediated by attention rather than feed-forward computation.

5. Conclusion

We introduced *planning site formation* and studied it across three open-source model families using linear probing and activation patching. Across our experiments, encoding and use turn out to be dissociable. Probes detect rhyme-relevant information at the newline in many models and many scales, yet only Gemma-3-27B treats that information as causal during generation. Every other model we tested conditions on the rhyme word throughout, despite scale-dependent probe signal at the newline. Probe signal is not by itself evidence of a planning site.

When the handoff does occur it is implemented by a sparse, identifiable mechanism. In Gemma-3-27B we trace it to five attention heads in layers 28 and 30: the k-sweep peaks at exactly that set, declining as more heads are added, while random and comma-control head sets stay at zero across all k. Planning site formation, when it appears, is a structured computational phenomenon rather than a diffuse property of many components.

Our activation patching approach requires no extraneous model training and far less data than previous steering vector or transcoder-based approaches, making it scalable to large models across many architectures. The results, however, also surface important open questions and limitations that future work should address.

First, our analysis is limited to three model families on a single structured generation task. Extending the methodology to prose generation, code completion, and multi-step reasoning tasks would test whether the representational handoff is a general planning primitive or a narrow phonological phenomenon. Second, while two-stage path patching localizes the handoff to a five-head set, the wide upper bound on the recovered fraction (CI extending past 1.0) reflects the small number of prompt pairs. A larger, more diverse couplet set would tighten this estimate and reveal whether the head set varies by rhyme family. Third, the absence of the handoff in all Qwen3 and Llama-3 models despite strong probe signal raises a question about what distinguishes Gemma-3-27B architecturally or by training. Finally, causal scrubbing (Chan et al., 2022) or activation steering experiments targeting the planning heads would help distinguish whether the planning representation at the newline is genuinely read during generation or exerts influence only when artificially inserted via patching.

References

Ameisen, E., Lindsey, J., Pearce, A., Gurnee, W., Turner, N. L., Chen, B., Citro, C., Abrahams, D., Carter, S., Hosmer, B., Marcus, J., Sklar, M., Templeton, A., Bricken, T., McDougall, C., Cunningham, H., Henighan, T., Jermyn, A., Jones, A., Persic, A., Qi, Z., Thompson, T. B., Zimmerman, S., Rivoire, K., Conerly, T., Olah, C., and Batson, J. Circuit tracing: Revealing computational graphs in language models. https://transformer-circuits.pub/2025/attribution-graphs/methods.html, 2025. Transformer Circuits Thread.

Anthropic. Claude Sonnet 4.6. https://www.anthropic.com/news/claude-sonnet-4-6, February 2026. Large language model. Model string: claude-sonnet-4-6. Accessed: 2026-03-05.

Arditi, A., Obeso, O., Syed, A., Paleka, D., Panickssery, N., Gurnee, W., and Nanda, N. Refusal in language models is mediated by a single direction, 2024. URL https://arxiv.org/abs/2406.11717.

Burns, C., Ye, H., Klein, D., and Steinhardt, J. Discovering latent knowledge in language models without supervision, 2023. URL https://arxiv.org/abs/2212.03827.

Chan, L., Garriga-Alonso, A., Goldowsky-Dill, N., Greenblatt, R., Nitishinskaya, J., Radhakrishnan, A., Shlegeris, B., and Thomas, N. Causal scrubbing: a method for rigorously testing interpretability hypotheses. Alignment Forum, 2022. URL https://www.alignmentforum.org/posts/JvZhhzycHu2Yd57RN/causal-scrubbing-a-method-for-rigorously-testing-interpretability.

Dong, Z., Zhou, Z., Liu, Z., Yang, C., and Lu, C. Emergent response planning in llms, 2025. URL https://arxiv.org/abs/2502.06258.

Dunefsky, J., Chlenski, P., and Nanda, N. Transcoders find interpretable LLM feature circuits. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=J6zHcScAo0.

Gao, L., Biderman, S., Black, S., Golding, L., Hoppe, T., Foster, C., Phang, J., He, H., Thite, A., Nabeshima, N., et al. The Pile: An 800GB dataset of diverse text for language modeling. *arXiv preprint arXiv:2101.00027*, 2021.

Goldowsky-Dill, N., MacLeod, C., Sato, L., and Arora, A. Localizing model behavior with path patching, 2023. URL https://arxiv.org/abs/2304.05969.

Hanna, M. and Ameisen, E. Latent planning emerges with scale. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=H0B7pDTT0M.

Hao, S., Sukhbaatar, S., Su, D., Li, X., Hu, Z., Weston, J., and Tian, Y. Training large language models to reason in a continuous latent space, 2024. URL https://arxiv.org/abs/2412.06769.

Hewitt, J. and Liang, P. Designing and interpreting probes with control tasks, 2019. URL https://arxiv.org/abs/1909.03368.

Jenner, E., Kapur, S., Georgiev, V., Allen, C., Emmons, S., and Russell, S. Evidence of learned look-ahead in a chess-playing neural network, 2024. URL https://arxiv.org/abs/2406.00877.

Li, K., Hopkins, A. K., Bau, D., Viégas, F., Pfister, H., and Wattenberg, M. Emergent world representations: Exploring a sequence model trained on a synthetic task, 2022. URL https://arxiv.org/abs/2210.13382.

Lindsey, J., Gurnee, W., Ameisen, E., Chen, B., Pearce, A., Turner, N. L., Citro, C., Abrahams, D., Carter, S., Hosmer, B., Marcus, J., Sklar, M., Templeton, A., Bricken, T., McDougall, C., Cunningham, H., Henighan, T., Jermyn, A., Jones, A., Persic, A., Qi, Z., Thompson, T. B., Zimmerman, S., Rivoire, K., Conerly, T., Olah, C., and Batson, J. On the biology of a large language model. https://transformer-circuits.pub/2025/attribution-graphs/biology.html, 2025. Transformer Circuits Thread.

Loshchilov, I. and Hutter, F. Decoupled weight decay regularization. In *International Conference on Learning Representations (ICLR)*, 2019. URL https://arxiv.org/abs/1711.05101.

Maar, J., Paperno, D., McDougall, C. S., and Nanda, N. What's the plan? metrics for implicit planning in llms and their application to rhyme generation and question answering, 2026. URL https://arxiv.org/abs/2601.20164.

Marks, S., Rager, C., Michaud, E. J., Belinkov, Y., Bau, D., and Mueller, A. Sparse feature circuits: Discovering and editing interpretable causal graphs in language models. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=I4e82CIDxv.

McGrath, T., Kapishnikov, A., Tomašev, N., Pearce, A., Hassabis, D., Kim, B., Paquet, U., and Kramnik, V. Acquisition of chess knowledge in alphazero. *Proceedings of the National Academy of Sciences*, 119(47), November 2022. ISSN 1091-6490. doi: 10.1073/pnas.2206625119. URL http://dx.doi.org/10.1073/pnas.2206625119.

Meng, K., Bau, D., Andonian, A., and Belinkov, Y. Locating and editing factual associations in gpt, 2022. URL https://arxiv.org/abs/2202.05262.

Nanda, N., Lee, A., and Wattenberg, M. Emergent linear representations in world models of self-supervised sequence models, 2023. URL https://arxiv.org/abs/2309.00941.

Pfau, J., Merrill, W., and Bowman, S. R. Let's think dot by dot: Hidden computation in transformer language models, 2024. URL https://arxiv.org/abs/2404.15758.

Pochinkov, N., Volkova, Y., Vasileva, A., and Chereddy, S. V. R. Parascopes: What do language models activations encode about future text?, 2025. URL https://arxiv.org/abs/2511.00180.

Turner, A. M., Thiergart, L., Leech, G., Udell, D., Vazquez, J. J., Mini, U., and MacDiarmid, M. Steering language models with activation engineering, 2023. URL https://arxiv.org/abs/2308.10248.

Wang, K., Variengien, A., Conmy, A., Shlegeris, B., and Steinhardt, J. Interpretability in the wild: a circuit for indirect object identification in GPT-2 small, 2022. URL https://arxiv.org/abs/2211.00593.

Wei, J., Wang, X., Schuurmans, D., Bosma, M., Ichter, B., Xia, F., Chi, E., Le, Q., and Zhou, D. Chain-of-thought prompting elicits reasoning in large language models, 2022. URL https://arxiv.org/abs/2201.11903.

Weide, R. L. The CMU Pronouncing Dictionary. http://www.speech.cs.cmu.edu/cgi-bin/cmudict, 1993.

Wilson, E. B. Probable inference, the law of succession, and statistical inference. *Journal of the American Statistical Association*, 22(158):209–212, 1927.

A. Model Configurations

For reference, we report details of the architectural differences between models.

We also note important tokenization differences. Qwen and Llama treat `,\n` as a single token, while Gemma treats `,` and `\n` as separate tokens. This places the last word r_1 at position -1 in Qwen and Llama and position -2 in Gemma.

	Qwen3-32B	Gemma-3-27B	Llama-3.1-70B
$\lvert\mathcal{V}\rvert$	151,936	262,208	128,256
d	5,120	5,376	8,192
L	64	62	80

Table 1. Model architecture summary.

	-2	-1	0	1
Qwen3	`of`	`fright`	`,\n`	`and`
Llama-3	`of`	`fright`	`,\n`	`and`
Gemma-3	`fright`	`,`	`\n`	`and`

Table 2. Tokenization of the line boundary across model families.

B. Steering Vectors

We also replicate the main finding with a steering vector intervention (Turner et al., 2023; Maar et al., 2026). For each rhyme-scheme pair (s, t) and each (ℓ, i), the steering vector $\mathbf{v}_{\ell,i}^{(s\to t)} = \bar{\mathbf{h}}_{\ell,i}^{(t)} - \bar{\mathbf{h}}_{\ell,i}^{(s)}$ is the mean difference of residual activations at (ℓ, i) between prompts ending in scheme s and prompts ending in scheme t (Figure 9). At inference time we add $\alpha\mathbf{v}_{\ell,i}^{(s\to t)}$ at (ℓ, i) during generation on a held-out scheme-s prompt and measure the fraction of completions that rhyme in scheme t. We use ten schemes, $\alpha = 1.5$, and the same CMU rhyme matcher as the patching evaluation.

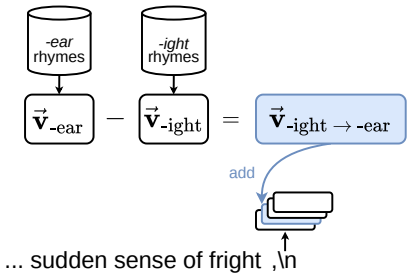


Figure 9. Steering vectors are mean-difference vectors between residual activations on prompts in two different rhyme schemes; adding $\alpha\mathbf{v}_{\ell,i}^{(s\to t)}$ at (ℓ, i) during generation should redirect the rhyme toward scheme t.

Computing the vectors requires 10 schemes $\times$ 100 train prompts $=$ 1,000 hooked forward passes, and the evaluation sweep covers every (ℓ, i) across scheme pairs at 20 held-out prompts each. The patching runs in Section 4 use 5 prompt pairs $\times$ N=20 samples per cell. Same causal question, one to two orders of magnitude more data; we therefore treat steering as a cross-check rather than the primary method.

Figure 10 shows the steered rhyme fraction at the last word and newline across all layers for Qwen3-32B, Gemma-3-27B, and Llama-3.1-70B; the qualitative picture matches Figure 6. In Gemma-3-27B, last-word steering is effective in early layers and drops sharply around layer 30 while newline steering rises simultaneously into the 0.85–0.95 range across layers 30–40—the same handoff. Qwen3-32B and Llama-3.1-70B show effective last-word steering at every swept layer with newline steering at noise. The handoff and its absence both reappear under steering, and the rate estimates are tighter than the patching CIs because of the larger evaluation set.

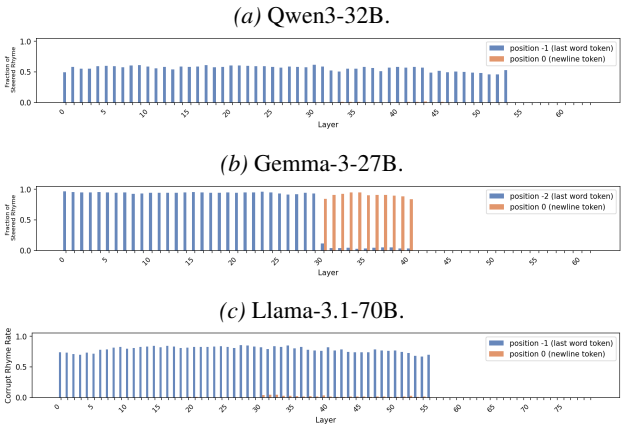


Figure 10. Steered rhyme fraction at the last word and newline across all layers, averaged over scheme pairs (α=1.5, 100 train prompts per scheme). The picture mirrors Figure 6: Gemma-3-27B shows the same handoff (last-word effect drops near layer 30 as the newline effect rises to 0.85–0.95), while Qwen3-32B and Llama-3.1-70B show effective last-word steering at every swept layer with newline steering at noise.

C. Additional Probing Results

For probing experiments, we also evaluated on top-1 accuracy. Note these are similar to the reported results, only scaled down.

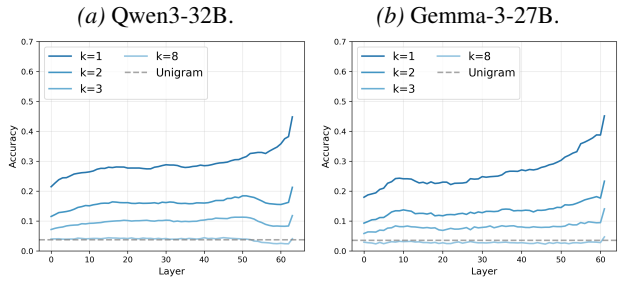


Figure 11. Top-1 probe accuracy predicting k tokens ahead in general text (Pile). Mirrors the top-5 pattern: accuracy degrades monotonically with k and falls to unigram baseline by $k = 8$.

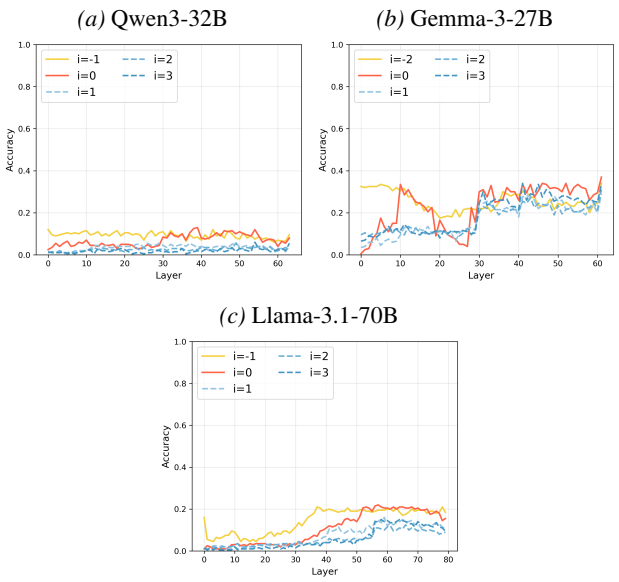


Figure 12. Top-1 probe accuracy predicting r_2 on rhyming couplets. Mirrors the top-5 and rhyme accuracy results: the $i \leq 0$ probes show substantially higher accuracy in middle-to-late layers than probes at $i > 0$.

D. Additional Activation Patching Details

Baselines

To verify that the observed corrupt rhyme rates reflect the specific encoding of the corrupt rhyme word rather than a generic effect of perturbing the residual stream, we run two control conditions. The *zero-vector* baseline replaces the patched hidden state with an all-zeros vector. The *donor-prompt* baseline replaces it with a hidden state cached from the same token position in a semantically unrelated sentence ("The weather outside is warm and sunny today, and the birds are singing.").

Figure 13 shows results for Qwen3-32B (at $i = -1$) and Gemma-3-27B (at $i = -2$), averaged over all prompt pairs. Both control conditions produce corrupt rhyme rates near zero across all layers, while true activation patching reaches a peak of 0.96 for Gemma-3-27B and 0.90 for Qwen3-32B (single fright/fear prompt with N=100). Across all layers and both control conditions, the Wilson 95% upper bound is at most 0.17, well below the true patching peak. This supports the claim that the patching effect is specific to the corrupt rhyme word's identity rather than a generic perturbation artifact.

All-Layers Patching

Table 3 reports corrupt rhyme rates when all layers are patched simultaneously at a given token position. Positions $i \geq 1$ yield zero corrupt rhyme rate and are omitted.

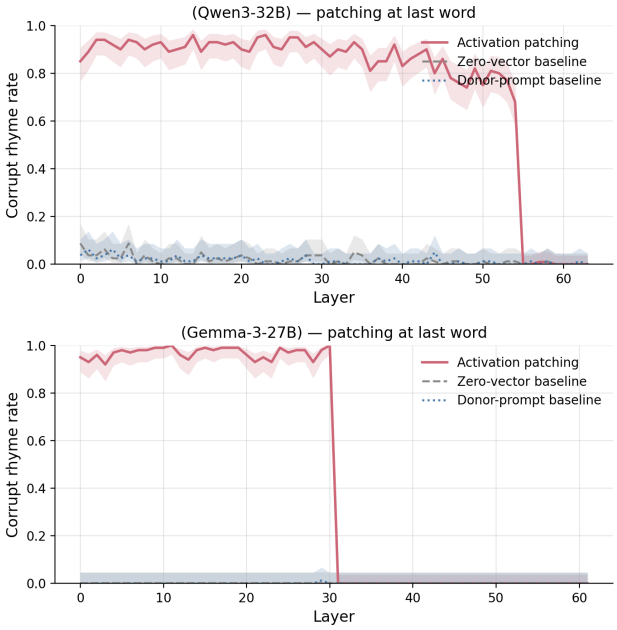


Figure 13. Corrupt rhyme rate under true activation patching versus zero-vector and donor-prompt baselines at the last word position. Shaded bands are Wilson 95% CIs (true patching N=100 per layer; baselines pooled across pairs $N \approx 80$ per layer). Both baselines yield near-zero rates across all layers, confirming that the patching effect is specific to the corrupt rhyme word's identity rather than an artifact of residual stream disruption.

Per-Layer Results: All Models

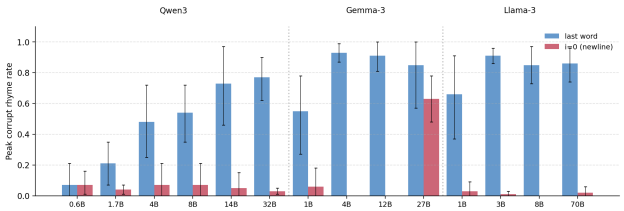


Figure 14. Peak corrupt rhyme rate (maximum across all layers) for the last word token and newline ($i = 0$) at each model size. Black error bars are 95% cluster bootstrap CIs over prompt pairs (5 pairs per model). Gemma-3-27B is the only model whose newline CI is clearly separated from zero (0.63 [0.48, 0.78]); every other model has a newline CI upper bound ≤ 0.21 regardless of scale, while last-word patching is broadly effective from ∼3B parameters onward in every family.

Full Position Sweep

Prompt Pairs

For each pair, we show the clean prompt, the corrupt prompt, and an example patched completion where activation patching was successful in steering the rhyme scheme.

Pair 1: doom/dread, $\ell = 10$, $i = -1$
Clean: `...filled with silent doom,\n when`

Family	Model	Last Word [95% CI]	$i = 0$ [95% CI]
Qwen3	0.6B	1 [0, 5]	7 [3, 14]
	1.7B	15 [9, 23]	2 [1, 7]
	4B	42 [33, 52]	7 [3, 14]
	8B	53 [43, 62]	2 [1, 7]
	14B	63 [53, 72]	7 [3, 14]
	32B	76 [67, 83]	1 [0, 5]
Gemma-3	1B	37 [28, 47]	1 [0, 5]
	4B	78 [69, 85]	0 [0, 4]
	12B	90 [83, 94]	0 [0, 4]
	27B	85 [77, 91]	67 [57, 75]
Llama-3	1B	59 [49, 68]	2 [1, 7]
	3B	87 [79, 92]	0 [0, 4]
	8B	79 [70, 86]	0 [0, 4]
	70B	75 [66, 82]	2 [1, 7]

Table 3. Corrupt rhyme rate (%) when all layers are patched simultaneously, with Wilson 95% CIs. All cells use N=100 (5 prompt pairs $\times$ N=20). Gemma-3-27B is the only model whose newline-patching CI does not overlap zero, separated by a wide margin from every other model.

`suddenly they`
Corrupt: `...filled with silent dread,\n when suddenly they`
Patched: `when suddenly they heard a creaking bed.`

Pair 2: bliss/joy, $\ell = 1$, $i = -2$
Clean: `...The children laughed in bliss,\n until they all`
Corrupt: `...The children laughed in joy,\n until they all`
Patched: `until they all became a toy.`

Pair 3: dark/night, $\ell = 0$, $i = -2$
Clean: `...She wandered home alone into the dark,\n and then she`
Corrupt: `...She wandered home alone into the night,\n and then she`
Patched: `and then she saw a strange and eerie light.`

Pair 4: grief/pain, $\ell = 2$, $i = -1$
Clean: `...I never knew the depth of such grief,\n as though the`
Corrupt: `...I never knew the depth of such pain,\n as though the`
Patched: `as though the sky had lost its rain.`

Pair 5: fright/fear, $\ell = 1$, $i = -1$
Clean: `...She felt a sudden sense of fright,\n and hoped that`
Corrupt: `...She felt a sudden sense of fear,\n and hoped that`
Patched: `and hoped that someone would appear.`

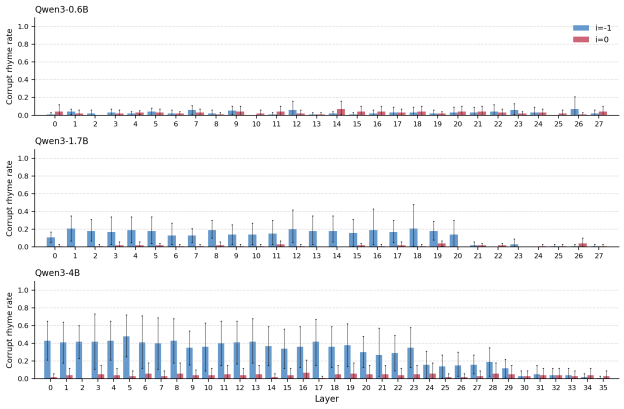


Figure 15. Per-layer activation patching, Qwen3 0.6B–4B. Black bars are 95% cluster bootstrap CIs (5 pairs $\times$ N=20). Last-word patching becomes effective only from 4B (peak 0.48 [0.25, 0.72]); 0.6B and 1.7B have peak CIs that nearly span zero. Newline ($i = 0$) patching is at noise across all sizes (peak CI upper bounds ≤ 0.21).

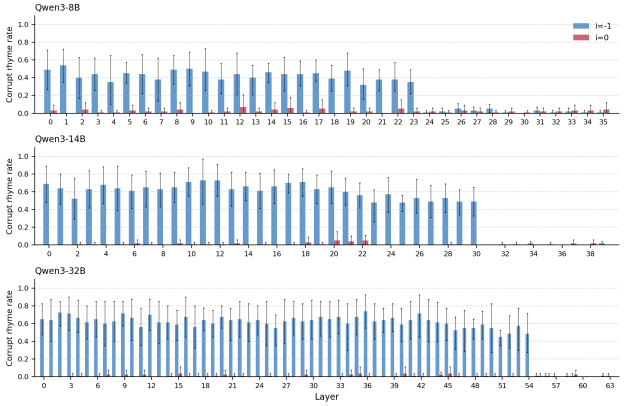


Figure 16. Per-layer activation patching, Qwen3 8B–32B. Black bars are 95% cluster bootstrap CIs. Last-word peaks rise smoothly with scale (8B 0.54 [0.35, 0.72]; 14B 0.73 [0.46, 0.97]; 32B 0.74 [0.55, 0.93]). Newline patching remains at noise across all three sizes (CI upper bounds ≤ 0.21).

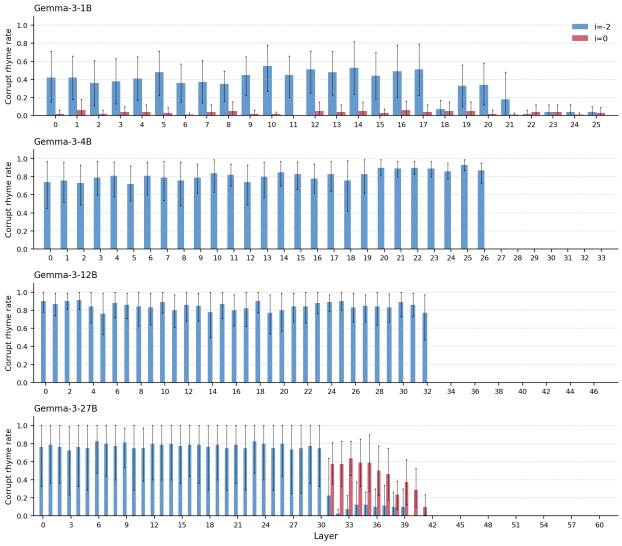


Figure 17. Per-layer activation patching, Gemma-3 1B–27B. Black bars are 95% cluster bootstrap CIs. The newline ($i = 0$) channel is silent at every size below 27B (peak CI [0.00, 0.00] for 4B and 12B), and only emerges at 27B with peak 0.63 [0.48, 0.78] at L33. Last-word ($i = -2$) patching is effective from 1B onward.

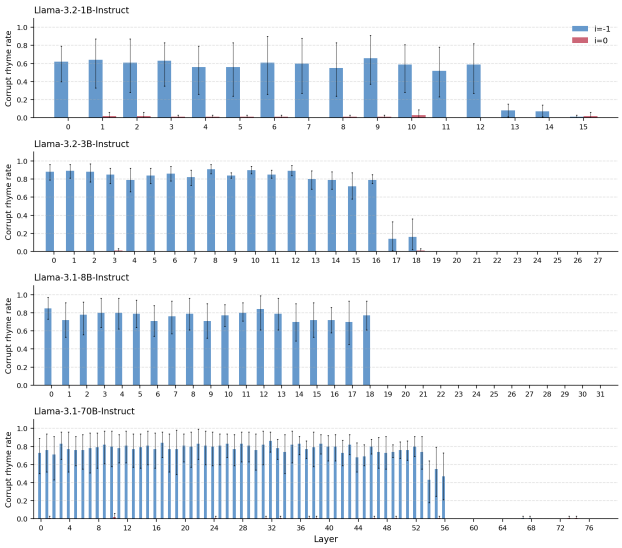


Figure 18. Per-layer activation patching, Llama-3 1B–70B. Black bars are 95% cluster bootstrap CIs. Last-word peaks are high from 3B onward (3B 0.91 [0.86, 0.96]; 8B 0.85 [0.73, 0.97]; 70B 0.86 [0.74, 0.96]); 1B is lower (peak 0.66 [0.56, 0.75]). Newline patching is at noise across all four sizes (CI upper bounds ≤ 0.09).

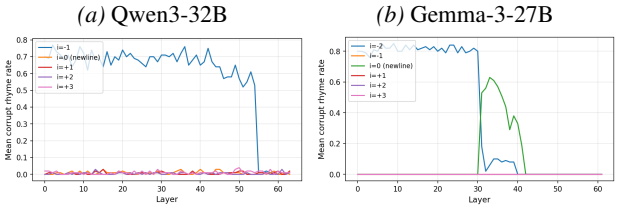
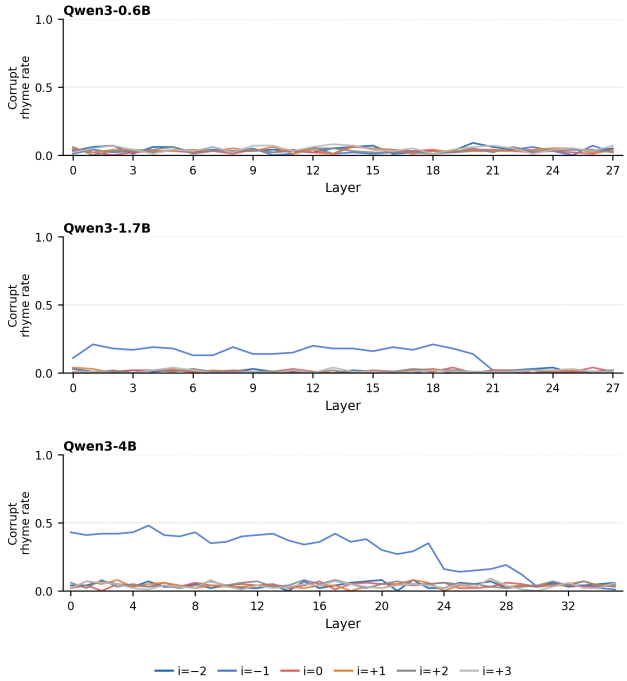
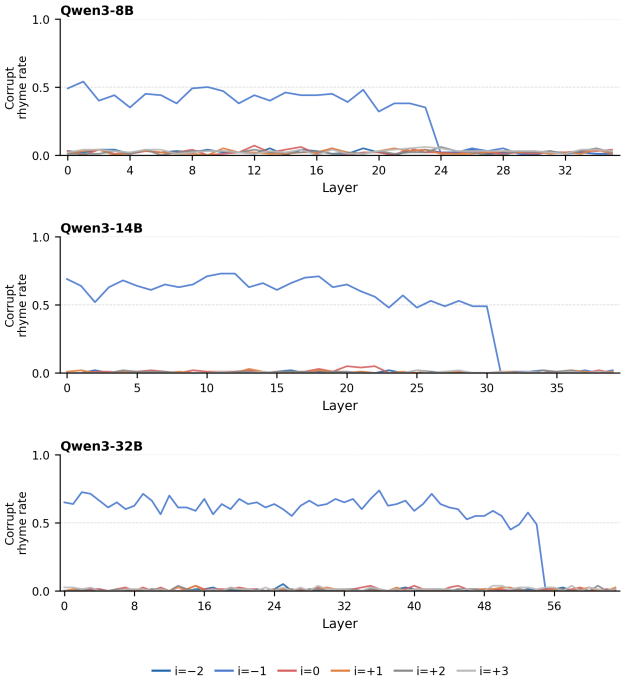


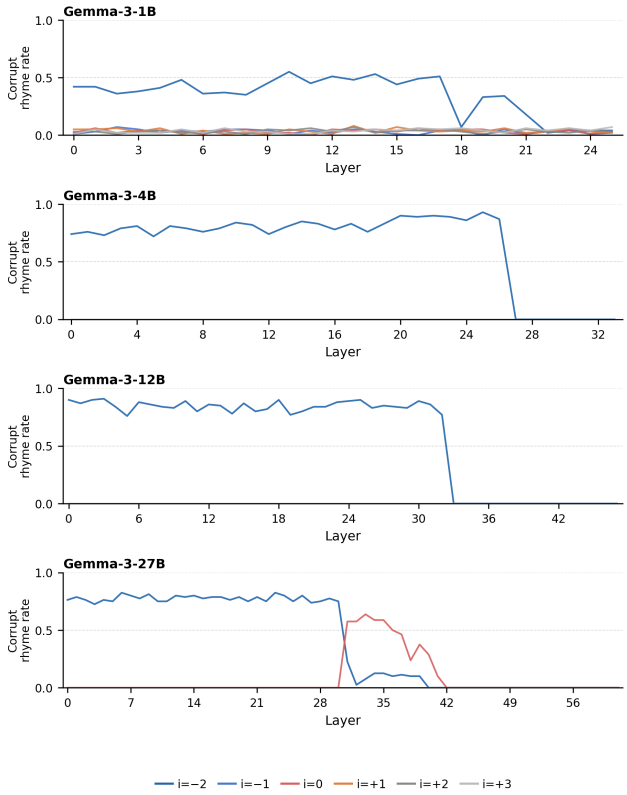
Figure 19. Corrupt rhyme rate across all six swept token positions, averaged over 5 prompt pairs. For Gemma-3-27B, the comma token at $i = -1$ is near zero throughout, confirming the handoff is specific to the newline token.



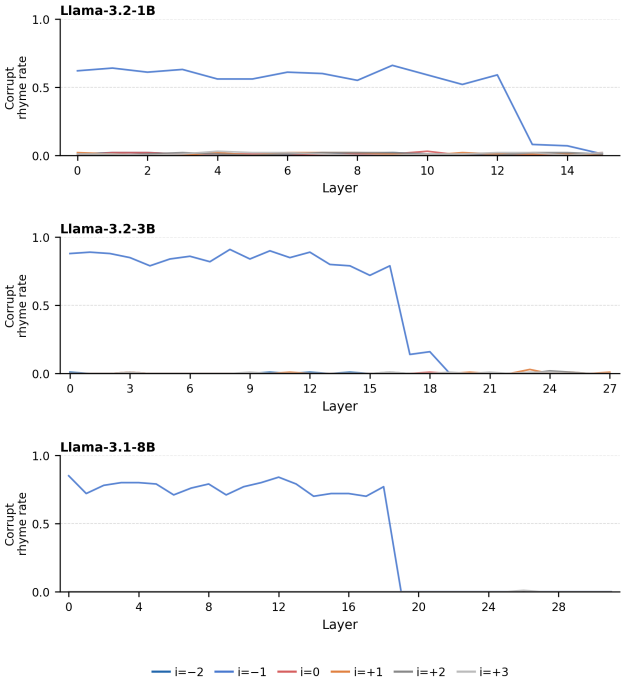
(a) Qwen3 0.6B–4B.



(b) Qwen3 8B–32B.



(c) Gemma-3 1B–27B.



(d) Llama-3 1B–70B.

Figure 20. Full position sweep across all four model groups in the Qwen3, Gemma-3, and Llama-3 families.