
Temporal Preference Concepts and their Functions in a Large Language Model

Ian Rios-Sialer* AISC San Francisco, USA	Shantanu Darveshi† AISC Mumbai, India	Shuai Jiang† AISC Albuquerque, USA	Avigya Paudel† SPAR, AISC New York, USA
Anastasiia Pronina† AISC Munich, Germany	Ipshita Bandyopadhyay SPAR Bangalore, India	Justin Shenk SPAR, AISC Berlin, Germany	

Abstract

Large Language Models (LLMs) are increasingly being deployed to make decisions that require trading off near-term gains against long-term consequences, yet little is known about how they internally represent or resolve these tradeoffs. In this work, we causally localize an underlying subgraph for temporal preference in a distilled LLM (Qwen3-4B-Instruct-2507), identifying mid-to-upper-layer nodes through converging evidence from gradient-based attribution and activation patching. We find that the geometry of time horizon is encoded in the residual stream at the expected localized layers. A behavioral analysis reveals that unintervened LLMs discount the future several times less steeply than humans, yet this preference is unstable across contexts, motivating explicit control rather than implicit reliance on training. Finally, we find suggestive evidence that steering vectors can shift temporal preference. Our work demonstrates how mechanistic interpretability can bring us closer to reliable control over how LLMs plan and reason.

1 Introduction

*All your life, you wait for the propitious time.
Then the propitious time
reveals itself as action taken.*

Louise Glück, *Landscape*

Large Language Models (LLMs) appear to hold preferences mediated by abstract concepts such as time. Impatience and paralysis both push humans into bad decisions. LLMs risk failing in similar ways. For now, the consequences have been limited. No coding agent cutting corners [67] has caused a catastrophe yet. But the stakes are rising quickly. In early 2026, the U.S. Department of Defense and Anthropic publicly clashed over a range of sensitive issues [2, 49, 21], including whether LLMs should be allowed to autonomously

*Correspondence to: ian@unrulyabstractions.com.

†Equal contribution. Listed alphabetically.

AISC = AI Safety Camp. SPAR = Supervised Program for Alignment Research.

operate weapons. In high-stakes scenarios, autonomous agents [39, 72] would need to trade off short-term gains against long-term effects [26]. When choosing among alternatives, the decision often depends on the temporal scope used to evaluate the consequences [20, 98, 32]. Temporal preference is indeed fundamental for planning [1], but also for cooperation [103, 56] and trust [27], where agents must bear present costs for future collective benefit [84]. These intertemporal tradeoffs grow even more consequential in the context of Artificial General Intelligence (AGI). A myopic system [77] poses different risks than one capable of scheming across long horizons [70, 82]. Detecting and maintaining control [36] over these capabilities while that is still tractable motivates our inquiry. **Where and how does an LLM encode temporal preference?**

Previous work has investigated the existence of temporal representations [37], characterized the economic behavior of LLMs [17, 48], and even shown that risk preference can be steered [116]. Yet no work has identified *where* temporal preference lives inside an LLM, how it is geometrically organized, or how to control it through targeted intervention.

Using Mechanistic Interpretability (MI) [6] techniques, we isolate the components that are causally responsible for temporal preference and show how activation-space representations evolve through them. This offers a geometric perspective on how interventions function to shift temporal preference, even in general open-ended generation tasks.

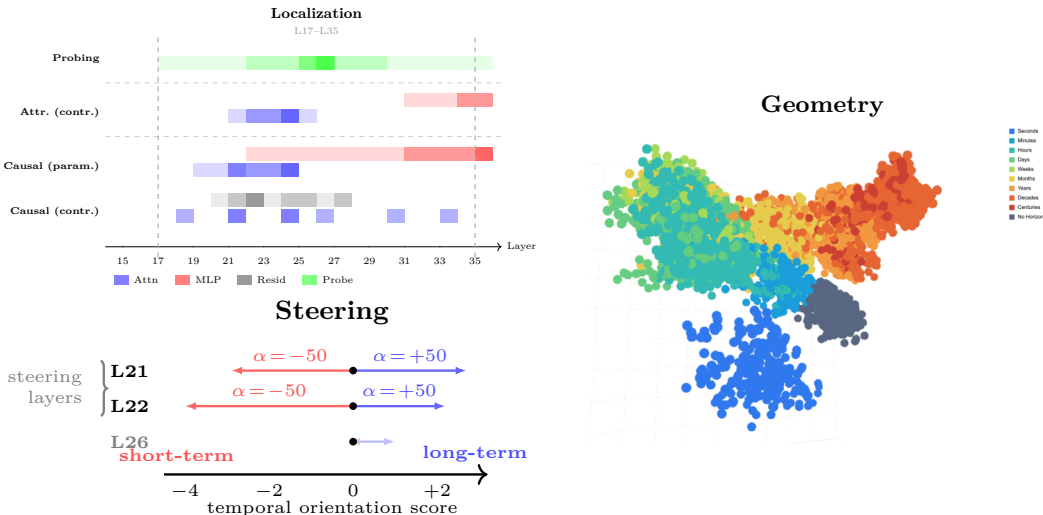


Figure 1: (Top-left) Five localization methods converge on a subgraph in layers 17–35; darker shading indicates stronger signal (Section 5.1). (Top-right) Time horizon geometry within the identified subgraph (Section 5.2). (Bottom) CAA steering shifts temporal preference bidirectionally at layers 19–22 but weakly at the best probing layer (L26), illustrating the probing–steering dissociation (Section 5.3).

We focus on `Qwen3-4B-Instruct-2507`: its non-thinking-only operation keeps computation inside a fixed prompt template (enabling token-aligned attribution and patching), and its latent preferences are stable under prompt perturbations, trading breadth across model families for depth in tracing a concept from localization through geometry to intervention. Our methodology integrates four localization pipelines plus dedicated behavioral and steering instruments that differ in prompting paradigm, localization technique, scale, and resolution. Because these pipelines approach localization from fundamentally different angles, their independent convergence on the same subgraph components strongly suggests that our findings reflect the genuine model structure.

Our work establishes that **temporal preference is localizable** within LLMs and, given the behavioral inconsistencies we observe, **should be explicitly controlled** rather than left to emerge implicitly from training. The convergence of multiple separate paradigms on the same subgraph demonstrates the value of **complementary experimental approaches** to validate mechanistic claims. Finally, while current interpretability methods are well-suited

for binary contrastive concepts (truthfulness [68], refusal [3], sycophancy [79]), this work takes initial steps toward **steering dimensional concepts** such as time, uncertainty, or risk preference, an underexplored area that warrants further development.

We summarize our contributions as follows:

Causal Localization of Temporal Preference (Section 5.1)

- We provide causal and attributional localization of a temporal-preference subgraph in Qwen3-4B-Instruct-2507 using complementary methods (Section 3).

Characterization of Temporal Preference (Section 5.2)

- We show that time horizon has non-linear geometry within the subgraph (Section 5.2).
- We provide a behavioral analysis (Section 5.2) that shows that unintervened LLMs behave very differently from humans, suggesting that the implicit time preference is inconsistent between contexts.

Steering of Temporal Preference (Section 5.3)

- We show successful interventions that change temporal preference and interpret them through our characterizations, via the steering methodology detailed in Appendix Appendix AA.

Our methodology combines multiple independent methods, datasets, and resolutions, each approaching the same subgraph from a different angle, to build converging evidence for how temporal preference is mechanistically implemented.

2 Background

We refer to **temporal preference** as the degree to which an agent values outcomes differently depending on when they occur [16, 103]. We use the term **time horizon** to denote the future moment at which outcomes are evaluated against an objective [23]. Because future events do not affect past outcomes, the time horizon also acts as a *constraint* on planning [89]. It would be *instrumentally* [59] or *means-end* [7] **incoherent** for the agent to choose actions that are incapable of causing effects by the specified deadline. Then, the **temporal scope** is the bounded interval of time over which an agent weighs the results according to its preference³.

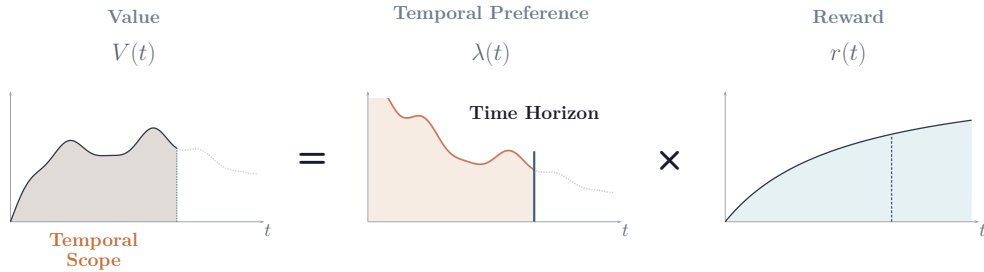


Figure 2: The time horizon specifies when the consequences of a decision are assessed. The temporal scope is then bounded above by the time horizon.

The empirical handle on temporal preference is **intertemporal choice**: a decision between options that differ in both reward and delay [29, 35]. Each option i is a tuple (r_i, t_i) of reward $r_i \in \mathbb{R}^+$ and delay $t_i \in \mathbb{R}^+$. Its subjective value is the reward scaled by a delay-dependent temporal-preference weight λ :

$$V_i = \lambda(t_i) \cdot r_i, \tag{1}$$

and, given a pair of options $\{A, B\}$, an agent with preference λ selects

$$i^* = \arg \max_{i \in \{A, B\}} \lambda(t_i) \cdot r_i. \tag{2}$$

³Although not explored in this work, when rewards are *perishable* [4], the temporal scope is also bounded below by the *retroactive reach* [50, 71].

Humans are typically modeled with hyperbolic discount functions [29, 69]; for LLMs, whether the same functional form fits is an open empirical question [69]. Characterizing LLM behavior therefore requires fitting λ via regression, assessing its stability across contexts, and benchmarking the resulting preferences against human intertemporal choice.

In humans, these concepts have localized neural representations [51] that predict behavior [93], respond causally to intervention [28], and exhibit internal organization interpretable as a *functional role* [18].⁴ Our work asks whether temporal preference exists in an LLM in an analogous way: localized, predictive, causally efficacious, and geometrically organized.

2.1 Locate and characterize, then steer

Our pipeline engages the target concept in three complementary modes. To *locate* the subgraph responsible for temporal preference, we pair *causal* intervention, activation patching [43] in do-calculus notation [83], with cheaper *attributional* proxies that scale across inputs: gradient-based EAP-IG [42, 6] and linear probes [75, 57] that surface where the concept linearly emerges. To *characterize* how the localized components encode horizon information, we apply PCA [95] inside the subgraph; prior work warns that concept geometry is often non-linear [24, 73, 38] and can drift across generation [60], so a single global direction rarely tells the whole story. Only after we have located and characterized the subgraph do we *steer*: we inject a probe-derived vector [100, 79] at inference time; localization is not strictly required but tightens precision, shrinks magnitudes, and reduces side effects [114]. Full definitions are in Appendix A.

2.2 Related work

Four strands of work frame this paper: (i) *temporal representation*, showing that LLMs encode time geometrically [37, 24, 53, 38] as locally linear features on globally curved manifolds [73, 81]; (ii) *temporal reasoning and planning*, where models fail despite the geometric encoding [105, 91, 106]; (iii) *LLM economic behavior*, reproducing human biases [48, 17, 104] with entangled risk/time preferences [116, 74, 69]; and (iv) *steering advancements*, from activation addition [100, 79] through sparse dictionaries [19] to geometry-aware methods [101, 87, 64, 85], with known failure modes at large $|\alpha|$ [108, 8]. No prior work has localized a subgraph functionally responsible for temporal preference, characterized the geometry of the causal representation, or steered along it. Full discussion in Appendix B.

3 Methodology

Our methodology follows three stages: *localize* the temporal-preference subgraph, *characterize* its representations, and *intervene* to steer it. Localization pairs *wide attribution* (contrastive A/B prompts \times EAP-IG and linear probing, cheap to sweep across hundreds of components) with *targeted intervention* (parametric prompts with explicit horizons \times activation patching, expensive but causal); the two paradigms converge on the same subgraph, which is the basis for our localization claim. Characterization applies PCA inside that subgraph to examine how explicit horizons organize the activation manifold and whether latent (no-horizon) preferences align with that geometry, paired with two behavioral instruments (Kirby MCQ-27 and a 30-model investment-coherence questionnaire) that test whether the geometry actually drives choice. Intervention uses Contrastive Activation Addition with a probe-derived steering vector, swept across layers and magnitudes to test for a probing-steering dissociation. Full per-pipeline protocols, dataset construction, metric definitions, and the reader’s guide to Part 4 are in the Methodology Summary appendix Appendix C.

⁴Some authors argue that concepts are best modeled by geometric or topological spaces [30], a perspective that resonates with our geometric analysis of temporal representations in activation space.

4 Experimental setup

We focus on a single model, `Qwen3-4B-Instruct-2507`, a mode-specialized non-thinking refresh of `Qwen3-4B` [86, 112]. We chose this model because operating in non-thinking mode keeps all “cognition” inside a fixed prompt template (no `<think>` block perturbs token positions), which is what the activation-patching and attribution pipelines need to align clean and corrupted runs; because its latent preference is stable under minor prompt perturbations at a scale where similar-sized models drift; and because it is small enough for repeated attribution and intervention sweeps. The pipeline operates on three dataset paradigms: minimally-framed A/B prompts (500 explicit + 500 implicit pairs), highly-formatted parametric prompts with explicit time horizons (4,588 prompts), and behavioral questionnaires (Kirby MCQ-27 plus a 960-prompt investment-coherence instrument run on 30 models). All experiments fit on a MacBook Pro (M4 Max, 48 GB) and reproduce end-to-end within two weeks. Full configurations, dataset construction, and model-selection rationale are in Appendix D.

5 Results

5.1 Where is temporal preference for the LLM?

Four localization methods converge on layers 17–35 (Figure 1, top-left; Appendix K). L24 attention is flagged by all three non-probing methods. MLP effects concentrate in L31–L35 across attributional contrastive and causal parametric patching (the causal contrastive run is attention-dominated). Probes peak at layer 26 (99.2%; Appendix G). Activation patching ranks the four highest-importance components as `L24_attn`, `L35_mlp`, `L31_mlp`, and `L21_attn`, separated from the remaining components in effect size (Figure 3).

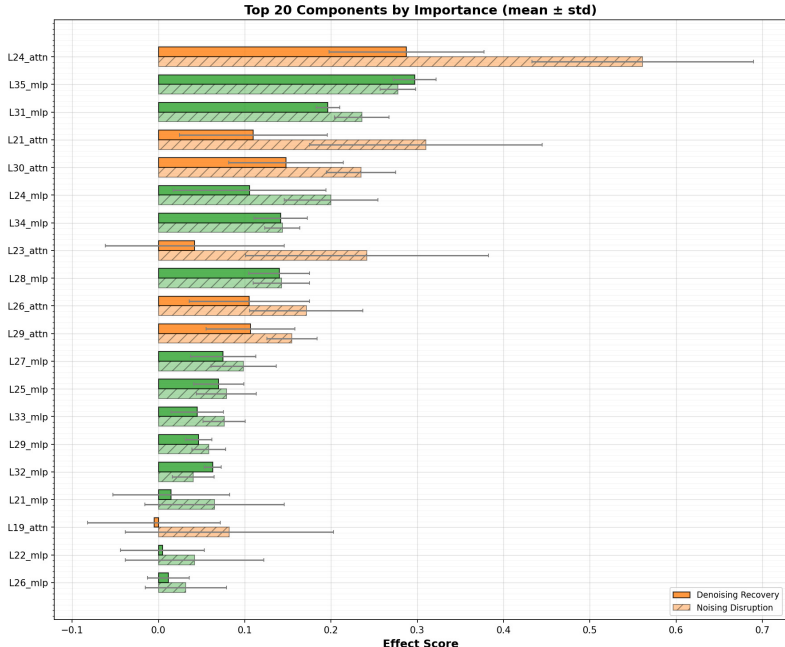


Figure 3: Top 20 components by causal parametric patching importance (mean \pm std across contrastive pairs). Solid bars: denoising recovery; hatched bars: noising disruption. The top-four cluster (`L24_attn`, `L35_mlp`, `L31_mlp`, `L21_attn`) is clearly separated; full ranking and per-component analysis in Appendix I.

5.2 What is the LLM’s temporal preference like?

Geometry. Time horizons form ordinal clusters (seconds to centuries) in activation space, but the direction encoding them is unstable across prompt positions until the user-to-assistant turn boundary, where attention collapses the continuous horizon representation into a binary preference signal that sharpens from heavy overlap at `<|im_end|>` to clean separation at `assistant` (Figure 4; Appendix L).

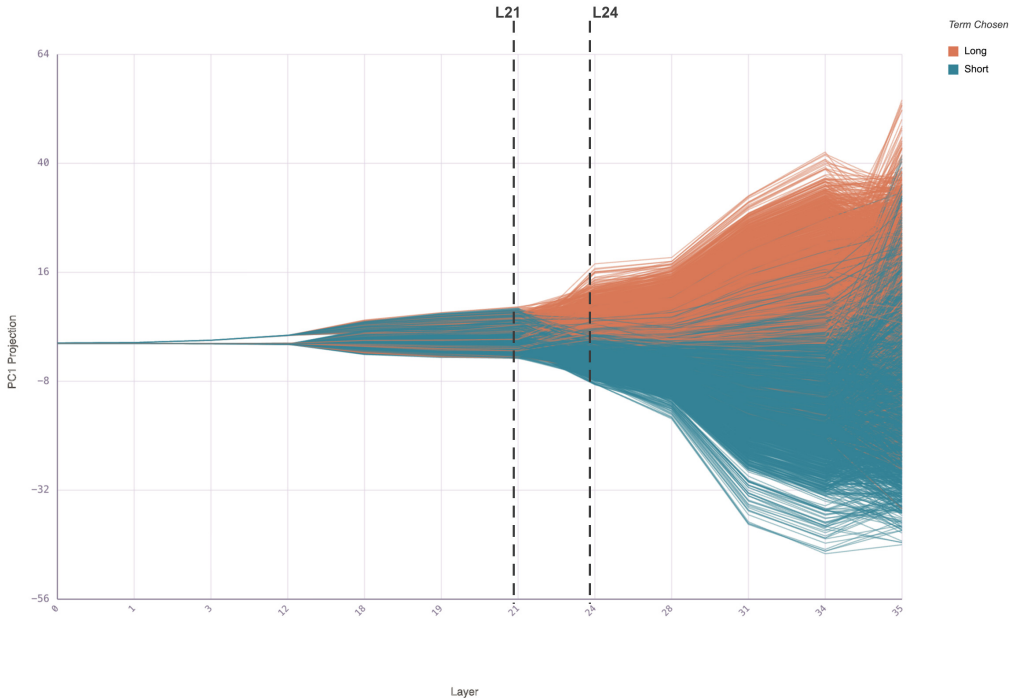


Figure 4: `resid_post` at the four turn-transition suffixes (`<|im_end|>`, `\n`, `<|im_start|>`, `assistant`), colored by preference (orange = long, blue = short). The preference signal sharpens from heavy overlap at suffix 0 to clean separation at suffix 3 (Appendix L).

Discounting. LLM discount rates ($k < 0.005$) are 3–8× below human controls ($k \approx 0.013$); chain-of-thought amplifies present bias in the 4B model but produces paradoxical patience in the 8B model (Appendix N).

Coherence. We test whether `Qwen3-4B-Instruct-2507` makes instrumentally coherent choices (Section 2) on 960 investment prompts offering \$20K in 6 months vs. a long-term option (\$100K, \$300K, or \$500K in 10 years) (Appendix O). Coherence is defined in the 1–5y reasoning zone, where only the 6-month option can deliver within the deadline. Our model does not meet this bar: it picks the undeliverable long-term option 47–53% of the time, and a deep dive shows this is positional polarization rather than uncertainty; reward size and label format are effectively inert (O.4.1). Benchmarking against 29 other models confirms the failure is not idiosyncratic: only frontier API models (`Claude` family, `Gemini 2.5 Pro`, `GPT-5.4`, `GPT-5.4 Mini`, `o3`) reach 95–100% coherence, and the smaller `Claude` variants do so via a binary heuristic that collapses at longer horizons (Figure 5).

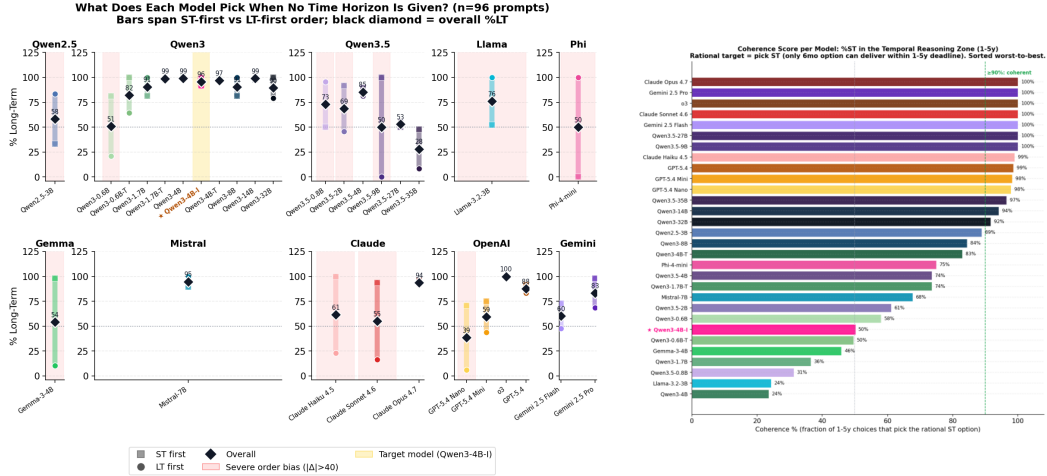


Figure 5: Qwen3-4B-Instruct-2507 (yellow) holds a stable temporal preference under presentation-order swaps in the no-horizon condition (left), but does not produce coherent temporal reasoning when given an explicit deadline (right; star marks our target at 50%, far below the 90% coherent threshold). Full per-model breakdowns in Appendix O.

Generality. Probing the same layers and turn-transition tokens for an unrelated meta-cognitive variable (the cumulative reliability of a multi-step reasoning chain) recovers a 95% decodability plateau over L19 to L31, perpendicular to the temporal direction in the linear sense but sharing curved-manifold structure with it; the gap between representation and behavior is therefore architectural rather than temporal-specific (Appendix Q).

5.3 Could temporal preference be controlled?

CAA steering at layers 19–22 shifts temporal preference bidirectionally: $\sim 3.4\times$ more probability mass toward long-term at layer 22, $\alpha = 50$ (Figure 1, bottom; Appendix R). Open-ended generation shifts from triage framing ($\alpha < 0$) to strategic framing ($\alpha > 0$). Output quality degrades at $|\alpha| = 60$, suggesting the linear steering vector exceeds the locally linear regime of the curved manifold. The optimal steering layers (19–22) sit 4–7 layers below the best probing layer (26), a probing–steering dissociation consistent with the localized subgraph.

6 Discussion

Whether an LLM operates under the right temporal preference is ultimately an alignment question. Post-training methods may suffice for routine use, but high-stakes settings call for stronger guarantees. We believe activation geometry can serve as a fail-safe here: localize the subgraph relevant to a specific task, characterize the geometry of the temporal concept within it, and then, at inference time, monitor the model’s internal representations against that manifold and intervene if they drift. This perspective frames interpretability not only as a diagnostic tool but as infrastructure for runtime alignment.

7 Limitations and future work

Our work is a starting point on an entangled concept. The main open directions are: finer-grained circuit tracing to move from subgraph-level attribution to atomic components and information flow; generalization beyond the single financial task and the single target model (Qwen3-4B-Instruct-2507) to other domains, model scales, and thinking vs. non-thinking variants; richer parameterization along reward, risk, role, and domain axes to map the full intertemporal choice space and its interactions with adjacent concepts such

as emotion and urgency; multi-turn and agentic settings where temporal representations may shift across turns; and non-linear steering methods that respect the curvature of the underlying manifold and avoid the output-quality degradation we observe in linear CAA at high $|\alpha|$. Full discussion is in Appendix F.

8 Conclusion

We show that temporal preference in LLMs is localizable, that we can characterize its representational geometry, and that targeted activation interventions can shift it bidirectionally. Our work highlights the value of using complementary paradigms. More broadly, while the literature has focused on identifying contrastive binary concepts, this work offers initial steps toward decomposing dimensional concepts such as time.

Acknowledgments and Disclosure of Funding

We thank the **AI Safety Camp (AISC)** and the **Supervised Program for Alignment Research (SPAR)** for providing the collaborative structure, mentorship, and computational resources that made this project possible. AISC’s cohort-based research model brought the authors together and sustained the multi-month investigation; SPAR’s program provided additional mentorship and connected contributors across timezones.

References

- [1] John Ameriks, Andrew Caplin, and John Leahy. Wealth accumulation and the propensity to plan. *The Quarterly Journal of Economics*, 118(3):1007–1047, 2003.
- [2] Dario Amodei. Statement from dario amodei on our discussions with the department of war, 2026. URL <https://www.anthropic.com/news/statement-department-of-war>.
- [3] Andy Arditi, Oscar Obeso, Aaquib Syed, Daniel Paleka, Nina Panickssery, Wes Gurnee, and Neel Nanda. Refusal in language models is mediated by a single direction. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang, editors, *Advances in Neural Information Processing Systems*, volume 37, pages 136037–136083. Curran Associates, Inc., 2024. doi: 10.52202/079017-4322. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/f545448535dfde4f9786555403ab7c49-Paper-Conference.pdf.
- [4] Kenneth J. Arrow, Theodore Harris, and Jacob Marschak. Optimal inventory policy. *Econometrica*, 19(3):250–272, 1951. doi: 10.2307/1906813.
- [5] Lukasz Bartoszczke, Sarthak Munshi, Bryan Sukidi, Jennifer Yen, Zejia Yang, David Williams-King, Linh Le, Kosi Asuzu, and Carsten Maple. Representation engineering for large-language models: Survey and research challenges, 2025. URL <https://arxiv.org/abs/2502.17601>.
- [6] Leonard Bereska and Efstratios Gavves. Mechanistic interpretability for ai safety – a review, 2024. URL <https://arxiv.org/abs/2404.14082>.
- [7] Michael E. Bratman. Intention and means-end reasoning. *The Philosophical Review*, 90(2):252–265, 1981.
- [8] Joschka Braun, Carsten Eickhoff, David Krueger, Seyed Ali Bahrainian, and Dmitrii Krashennikov. Understanding (un)reliability of steering vectors in language models, 2025. URL <https://arxiv.org/abs/2505.22637>.
- [9] Jon-Paul Cacioli. Categorical perception in large language model hidden states: Structural warping at digit-count boundaries, 2026. URL <https://arxiv.org/abs/2603.28258>.

- [10] Jon-Paul Cacioli. Weber’s law in transformer magnitude representations: Efficient coding, representational geometry, and psychophysical laws in language models, 2026. URL <https://arxiv.org/abs/2603.20642>.
- [11] Pengfei Cao, Tianyi Men, Wencan Liu, Jingwen Zhang, Xuzhao Li, Xixun Lin, Dianbo Sui, Yanan Cao, Kang Liu, and Jun Zhao. Large language models for planning: A comprehensive and systematic survey, 2025. URL <https://arxiv.org/abs/2505.19683>.
- [12] Iván Vicente Moreno Cencerrado et al. Temporal predictors of outcome in reasoning language models. *arXiv preprint arXiv:2511.14773*, 2025. URL <https://arxiv.org/html/2511.14773>.
- [13] Hao Chen, Antoine Didisheim, Luis Somoza, and Hui Tian. A financial brain scan of the LLM. *arXiv preprint arXiv:2508.21285*, 2025.
- [14] Jillian Chen et al. LLM economicus? mapping the behavioral biases of LLMs via utility theory. *arXiv preprint arXiv:2408.02784*, 2024.
- [15] Yue Cheng, Amirarsalan Sadeghian Moakhar, Cangxiong Fan, Pedram Hosseini, Kasma Faghieh, Weijia Wang, and Soheil Feizi. Your LLM agents are temporally blind: The misalignment between tool use decisions and human time perception. *arXiv preprint arXiv:2510.23853*, 2025. URL <https://arxiv.org/abs/2510.23853>.
- [16] Jonathan D. Cohen, Keith Marzilli Ericson, David Laibson, and John Myles White. Measuring time preferences. *Journal of Economic Literature*, 58(2):299–347, 2020. doi: 10.1257/jel.20191074.
- [17] Thomas R. Cook, Sophia Kazinnik, Zach Modig, and Nathan M. Palmer. What do LLMs want? Finance and Economics Discussion Series 2026-006, Board of Governors of the Federal Reserve System, January 2026. URL <https://www.federalreserve.gov/econres/feds/what-do-llms-want.htm>.
- [18] Robert Cummins. Functional analysis. *The Journal of Philosophy*, 72(20):741–765, 1975.
- [19] Hoagy Cunningham, Aidan Ewart, Logan Riggs, Robert Huben, and Lee Sharkey. Sparse autoencoders find highly interpretable features in language models. *arXiv preprint arXiv:2309.08600*, 2023.
- [20] Thomas J Dohmen, Armin Falk, David Huffman, and Uwe Sunde. Interpreting time horizon effects in inter-temporal choice. *CESifo Working Paper*, 2012.
- [21] Clare Duffy and Kaanita Iyer. Pentagon threatens to make Anthropic a pariah if it refuses to drop AI guardrails, 2026. URL <https://www.cnn.com/2026/02/24/tech/hegseth-anthropic-ai-military-amodei>.
- [22] Jacob Dunefsky, Philippe Chlenski, and Neel Nanda. Transcoders find interpretable llm feature circuits. *Advances in Neural Information Processing Systems*, 37:24375–24410, 2024.
- [23] Ronald J. Ebert and DeWayne Piehl. Time horizon: A concept for management. *California Management Review*, 15(4):65–70, 1973.
- [24] Joshua Engels, Eric J. Michaud, Isaac Liao, Wes Gurnee, and Max Tegmark. Not all language model features are linear. In *International Conference on Learning Representations (ICLR)*, 2025. URL <https://arxiv.org/abs/2405.14860>.
- [25] Bahare Fatemi, Mehran Kazemi, Anton Tsitsulin, Karishma Malkan, Jinyeong Yim, John Palowitch, Sungyong Seo, Jonathan Halcrow, and Bryan Perozzi. Test of time: A benchmark for evaluating llms on temporal reasoning, 2024. URL <https://arxiv.org/abs/2406.09170>.

- [26] William Fedus, Carles Gelada, Yoshua Bengio, Marc G. Bellemare, and Hugo Larochelle. Hyperbolic discounting and learning over multiple horizons, 2019. URL <https://arxiv.org/abs/1902.06865>.
- [27] Ernst Fehr and Andreas Leibbrandt. A field study on cooperativeness and impatience in the tragedy of the commons. *Journal of Public Economics*, 95(9–10):1144–1155, 2011.
- [28] Bernd Figner, Daria Knoch, Eric J. Johnson, Amy R. Krosch, Sarah H. Lisanby, Ernst Fehr, and Elke U. Weber. Lateral prefrontal cortex and self-control in intertemporal choice. *Nature Neuroscience*, 13(5):538–539, 2010.
- [29] Shane Frederick, George Loewenstein, and Ted O’Donoghue. Time discounting and time preference: A critical review. *Journal of Economic Literature*, 40(2):351–401, 2002. doi: 10.1257/002205102320161311.
- [30] Peter Gärdenfors. *Conceptual Spaces: The Geometry of Thought*. MIT Press, Cambridge, MA, 2000.
- [31] Aniketh Garikaparathi. Can llms perceive time? an empirical investigation, 2026. URL <https://arxiv.org/abs/2604.00010>.
- [32] Alexander F Gazmararian. Valuing the future: Changing time horizons and policy preferences. *Political Behavior*, 47(2):553–572, 2025.
- [33] Atticus Geiger, Duligur Ibeling, Amir Zur, Maheep Chaudhary, Sonakshi Chauhan, Jing Huang, Aryaman Arora, Zhengxuan Wu, Noah Goodman, Christopher Potts, and Thomas Icard. Causal abstraction: A theoretical foundation for mechanistic interpretability, 2025. URL <https://arxiv.org/abs/2301.04709>.
- [34] Nicholas Goldowsky-Dill, Chris MacLeod, Lucas Sato, and Aryaman Arora. Localizing model behavior with path patching. *arXiv preprint arXiv:2304.05969*, 2023.
- [35] Leonard Green and Joel Myerson. A discounting framework for choice with delayed and probabilistic rewards. *Psychological Bulletin*, 130(5):769–792, 2004.
- [36] Ryan Greenblatt, Buck Shlegeris, Kshitij Sachan, and Fabien Roger. Ai control: Improving safety despite intentional subversion, 2024. URL <https://arxiv.org/abs/2312.06942>.
- [37] Wes Gurnee and Max Tegmark. Language models represent space and time. In *The Twelfth International Conference on Learning Representations (ICLR)*, 2024. URL <https://openreview.net/forum?id=jE8xbmvFin>.
- [38] Wes Gurnee, Emmanuel Ameisen, Isaac Kauvar, Julius Tarng, Adam Pearce, Chris Olah, and Joshua Batson. When models manipulate manifolds: The geometry of a counting task. *arXiv preprint arXiv:2601.04480*, 2026.
- [39] António Guterres. Lethal autonomous weapon system “politically unacceptable, morally repugnant and should be banned”, 2025. URL <https://press.un.org/en/2025/sgsm22643.doc.htm>.
- [40] Xue Han, Qian Hu, Yitong Wang, Wenchun Gao, Lianlian Zhang, Qing Wang, Lijun Mei, Chao Deng, and Junlan Feng. Temporal alignment of llms through cycle encoding for long-range time representations, 2025. URL <https://arxiv.org/abs/2503.04150>.
- [41] Michael Hanna, Ollie Liu, and Alexandre Variengien. How does GPT-2 compute greater-than?: Interpreting mathematical abilities in a pre-trained language model. In *Advances in Neural Information Processing Systems*, volume 36, 2023.
- [42] Michael Hanna, Sandro Pezzelle, and Yonatan Belinkov. Have faith in faithfulness: Going beyond circuit overlap when finding model mechanisms, 2024. URL <https://arxiv.org/abs/2403.17806>.

- [43] Stefan Heimersheim and Neel Nanda. How to use and interpret activation patching, 2024. URL <https://arxiv.org/abs/2404.15255>.
- [44] Benjamin Heinzerling and Kentaro Inui. Monotonic representation of numeric properties in language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 207–218, 2024. URL <https://aclanthology.org/2024.acl-short.18/>.
- [45] David Herel, Vojtech Bartek, Jiri Jirak, and Tomas Mikolov. Time awareness in large language models: Benchmarking fact recall across time, 2025. URL <https://arxiv.org/abs/2409.13338>.
- [46] Carolin Holtermann, Paul Röttger, and Anne Lauscher. Around the world in 24 hours: Probing llm knowledge of time and place, 2025. URL <https://arxiv.org/abs/2506.03984>.
- [47] John J Horton. Large language models as simulated economic agents: What can we learn from homo silicus? *NBER Working Paper*, 2023.
- [48] John J. Horton, Apostolos Filippas, and Benjamin S. Manning. Large language models as simulated economic agents: What can we learn from homo silicus?, 2026. URL <https://arxiv.org/abs/2301.07543>.
- [49] Cloud Security Alliance AI Safety Initiative. Pentagon vs. Anthropic: Autonomous weapons AI guardrails and the governance crisis for enterprise AI vendors, 2026. URL <https://labs.cloudsecurityalliance.org/research/csa-research-note-dod-ai-guardrail-mandates-vendor-governanc/>.
- [50] International Risk Management Institute. Retroactive date. IRMI Insurance Glossary, n.d. URL <https://www.irmi.com/term/insurance-definitions/retroactive-date>. Accessed April 13, 2026.
- [51] Joseph W. Kable and Paul W. Glimcher. The neural correlates of subjective value during intertemporal choice. *Nature Neuroscience*, 10(12):1625–1633, 2007.
- [52] Marek Kadlčík et al. Pre-trained language models learn remarkably accurate representations of numbers. *arXiv preprint arXiv:2506.08966*, 2025. URL <https://arxiv.org/abs/2506.08966>.
- [53] Soham Kantamneni and Max Tegmark. Language models use trigonometry to do addition, 2025. URL <https://arxiv.org/abs/2502.00873>.
- [54] Subhash Kantamneni, Josh Engels, and Neel Nanda. Takeaways from our recent work on SAE probing. AI Alignment Forum, March 2025. URL <https://www.alignmentforum.org/posts/osNKnwiJWHxvDYvQTD/takeaways-from-our-recent-work-on-sae-probing>. Accessed: 2025.
- [55] Dhruva Karkada, Daniel J. Korchinski, Andres Nava, Matthieu Wyart, and Yasaman Bahri. Symmetry in language statistics shapes the geometry of model representations, 2026. URL <https://arxiv.org/abs/2602.15029>.
- [56] Jeongbin Kim. The effects of time preferences on cooperation: Experimental evidence from infinitely repeated games. *American Economic Journal: Microeconomics*, 15(1): 618–637, 2023.
- [57] Junsol Kim, James Evans, and Aaron Schein. Linear representations of political perspective emerge in large language models, 2025. URL <https://arxiv.org/abs/2503.02080>.
- [58] Kris N. Kirby, Nancy M. Petry, and Warren K. Bickel. Heroin addicts have higher discount rates for delayed rewards than non-drug-using controls. *Journal of Experimental Psychology: General*, 128(1):78–87, 1999. doi: 10.1037/0096-3445.128.1.78.

- [59] Christine M. Korsgaard. The normativity of instrumental reason. In Garrett Cullity and Berys Gaut, editors, *Ethics and Practical Reason*, pages 215–254. Oxford University Press, Oxford, 1997.
- [60] Andrew Kyle Lampinen, Yuxuan Li, Eghbal Hosseini, Sangnie Bhardwaj, and Murray Shanahan. Linear representations in language models can change dramatically over a conversation, 2026. URL <https://arxiv.org/abs/2601.20834>.
- [61] Jin Hwa Lee, Thomas Jiralerspong, Lei Yu, Yoshua Bengio, and Emily Cheng. Geometric signatures of compositionality across a language model’s lifetime, 2025. URL <https://arxiv.org/abs/2410.01444>.
- [62] Tom Leinster. *Entropy and Diversity: The Axiomatic Approach*. Cambridge University Press, 2021. ISBN 9781108832700. doi: 10.1017/9781108963558.
- [63] Yan Leng. Folk economics in the machine: LLMs and the emergence of mental accounting. *SSRN preprint 4705130*, 2024.
- [64] Jiaqian Li, Yangsibo Huang, and Daogao Liu. Steering vector fields for context-aware inference-time control in large language models, 2026. URL <https://arxiv.org/abs/2602.01654>.
- [65] Lingyu Li, Yang Yao, Yixu Wang, Chubo Li, Yan Teng, and Yingchun Wang. The other mind: How language models exhibit human temporal cognition, 2025. URL <https://arxiv.org/abs/2507.15851>.
- [66] Zijia Liu, Peixuan Han, Haofei Yu, Haoru Li, and Jiaxuan You. Time-r1: Towards comprehensive temporal reasoning in llms, 2025. URL <https://arxiv.org/abs/2505.13508>.
- [67] Monte MacDiarmid, David Duvenaud, and Roger Grosse. Emergent misalignment: Narrow finetuning can produce broadly misaligned llms, 2025. URL <https://arxiv.org/abs/2502.17424>.
- [68] Samuel Marks and Max Tegmark. The geometry of truth: Emergent linear structure in large language model representations of true/false datasets. In *Conference on Language Modeling (COLM)*, 2024. URL <https://arxiv.org/abs/2310.06824>.
- [69] Amin Mazyaki, Mohammad Naghizadeh, Shayan Ranjkhah Zonouzagh, and Hamed Setareh. Temporal preferences in language models for long-horizon assistance. *arXiv preprint arXiv:2509.09704*, 2025.
- [70] Alexander Meinke, Bronson Schoen, J r my Scheurer, Mikita Balesni, Rusheb Shah, and Marius Hobbhahn. Frontier models are capable of in-context scheming, 2025. URL <https://arxiv.org/abs/2412.04984>.
- [71] Ian M. Mitchell, Alexandre M. Bayen, and Claire J. Tomlin. A time-dependent Hamilton–Jacobi formulation of reachable sets for continuous dynamic games. *IEEE Transactions on Automatic Control*, 50(7):947–957, 2005. doi: 10.1109/TAC.2005.851439.
- [72] Melanie Mitchell et al. Fully autonomous AI agents should not be developed. *arXiv preprint arXiv:2502.02649*, 2025.
- [73] Aaron Modell and Patrick Rubin-Delanchy. The origins of representation manifolds in large language models. *arXiv preprint arXiv:2505.18235*, 2025. URL <https://arxiv.org/abs/2505.18235>.
- [74] Mehrdad Moghimi, Anthony Coache, and Hyejin Ku. Decoupling time and risk: Risk-sensitive reinforcement learning with general discounting, 2026. URL <https://arxiv.org/abs/2602.04131>.

- [75] Aaron Mueller, Atticus Geiger, Sarah Wiegrefe, Dana Arad, Iván Arcuschin, Adam Belfki, Yik Siu Chan, Jaden Fiotto-Kaufman, Tal Haklay, Michael Hanna, Jing Huang, Rohan Gupta, Yaniv Nikankin, Hadas Orgad, Nikhil Prakash, Anja Reusch, Aruna Sankaranarayanan, Shun Shao, Alessandro Stolfo, Martin Tutek, Amir Zur, David Bau, and Yonatan Belinkov. Mib: A mechanistic interpretability benchmark, 2025. URL <https://arxiv.org/abs/2504.13151>.
- [76] Jatin Nainani, Sankaran Vaidyanathan, Connor Watts, Andre N. Assis, and Alice Rigg. Detecting and characterizing planning in language models, 2025. URL <https://arxiv.org/abs/2508.18098>.
- [77] Richard Ngo, Lawrence Chan, and Sören Mindermann. The alignment problem from a deep learning perspective, 2025. URL <https://arxiv.org/abs/2209.00626>.
- [78] Kai Nylund, Suchin Gururangan, and Noah A. Smith. Time is encoded in the weights of finetuned language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL)*, 2024. URL <https://aclanthology.org/2024.acl-long.141/>.
- [79] Nina Panickssery, Nick Gabrieli, Julian Schulz, Meg Tong, Evan Hubinger, and Alexander Matt Turner. Steering Llama 2 via contrastive activation addition, 2024. URL <https://arxiv.org/abs/2312.06681>.
- [80] Vassilis Papadopoulos, Jérémie Wenger, and Clément Hongler. Arrows of time for large language models, 2024. URL <https://arxiv.org/abs/2401.17505>.
- [81] 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 (ICML)*, 2024. URL <https://arxiv.org/abs/2311.03658>.
- [82] Peter S. Park, Simon Goldstein, Aidan O’Gara, Michael Chen, and Dan Hendrycks. Ai deception: A survey of examples, risks, and potential solutions, 2023. URL <https://arxiv.org/abs/2308.14752>.
- [83] Judea Pearl. *Causality*. Cambridge university press, 2009.
- [84] Emil Persson, Gustav Tinghög, and Daniel Västfjäll. Intertemporal prosocial behavior: a review and research agenda. *Frontiers in psychology*, 15:1359447, 2024.
- [85] Joris Postmus and Steven Abreu. Steering large language models using conceptors: Improving addition-based activation engineering. *arXiv preprint arXiv:2410.16314*, 2025.
- [86] Qwen Team. Qwen3-4b-instruct-2507. <https://huggingface.co/Qwen/Qwen3-4B-Instruct-2507>, 2025. Accessed: 2026.
- [87] Harsh Raj, Vipul Gupta, Luca Zancato, Benjamin Bowman, Wenbo Gong, Alex Sherrod, Alex Wilson, and Ser-Nam Lim. Curveball steering: The right direction to steer isn’t always linear, 2026. URL <https://arxiv.org/abs/2603.09313>.
- [88] Goutham Rajendran, Simon Buchholz, Bryon Aragam, Bernhard Schölkopf, and Pradeep Kumar Ravikumar. From causal to concept-based representation learning. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL <https://openreview.net/forum?id=r5nev2SHtJ>.
- [89] Gwendolyn Reilly, David Souder, and Rebecca Ranucci. Time horizon of investments in the resource allocation process. *Journal of Management*, 42(5):1169–1194, 2016. doi: 10.1177/0149206316630381.
- [90] Baturay Saglam, Paul Kassianik, Blaine Nelson, Sajana Weerawardhena, Yaron Singer, and Amin Karbasi. Large language models encode semantics and alignment in linearly separable representations, 2026. URL <https://arxiv.org/abs/2507.09709>.

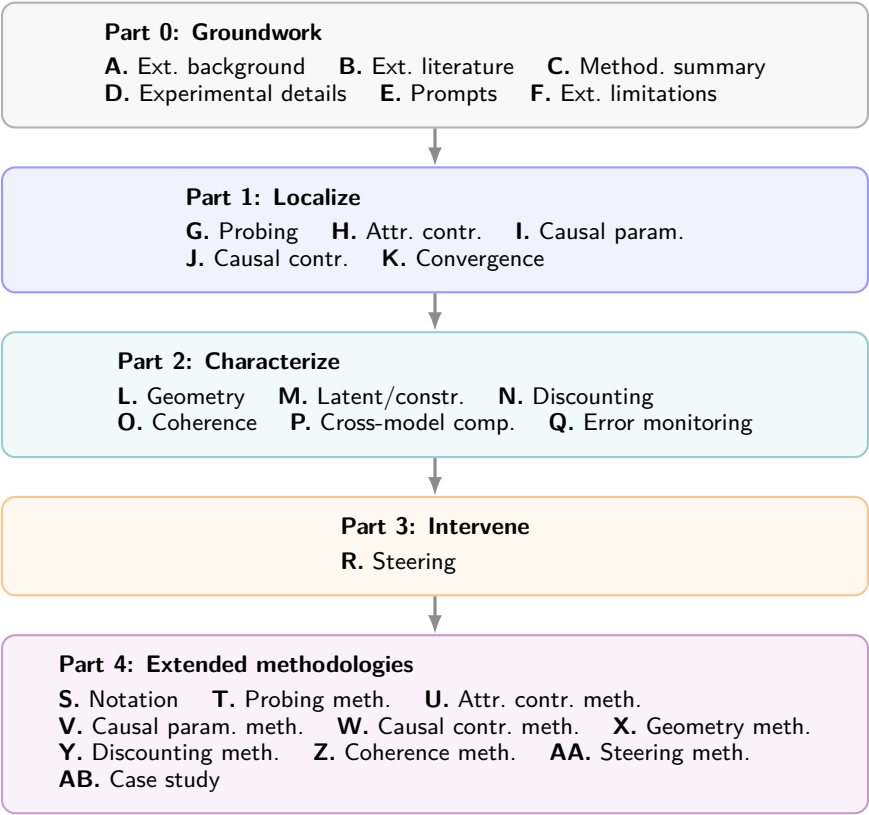
- [91] Neil K. R. Sehgal, Sharath Chandra Guntuku, and Lyle Ungar. Real-time deadlines reveal temporal awareness failures in llm strategic dialogues, 2026. URL <https://arxiv.org/abs/2601.13206>.
- [92] Or Shafran, Shaked Ronen, Omri Fahn, Shauli Ravfogel, Atticus Geiger, and Mor Geva. From directions to regions: Decomposing activations in language models via local geometry, 2026. URL <https://arxiv.org/abs/2602.02464>.
- [93] Noah A. Shamosh and Jeremy R. Gray. Delay of gratification: a psychometric investigation of the link between intelligence and self-control. *Journal of Risk and Uncertainty*, 36(3):249–263, 2008.
- [94] Jeonghwan Shin et al. TARDIS: Mitigating temporal misalignment via representation steering. *arXiv preprint arXiv:2503.18693*, 2025. URL <https://arxiv.org/abs/2503.18693>.
- [95] Jonathon Shlens. A tutorial on principal component analysis. *arXiv preprint arXiv:1404.1100*, 2014.
- [96] Nicholas Sofroniew, Isaac Kauvar, William Saunders, Runjin Chen, Tom Henighan, Sasha Hydrice, Craig Citro, Adam Pearce, Julius Tarng, Wes Gurnee, Joshua Batson, Sam Zimmerman, Kelley Rivoire, Kyle Fish, Chris Olah, and Jack Lindsey. Emotion concepts and their function in a large language model, 2026. URL <https://arxiv.org/abs/2604.07729>.
- [97] Xiangchen Song, Jiaqi Sun, Zijian Li, Yujia Zheng, and Kun Zhang. Llm interpretability with identifiable temporal-instantaneous representation, 2026. URL <https://arxiv.org/abs/2509.23323>.
- [98] Ola Svenson and Gunnar Karlsson. Decision-making, time horizons, and risk in the very long-term perspective. *Risk Analysis*, 9(3):385–399, 1989.
- [99] Anej Tiblias, Matej Klemen, and Marko Robnik-Šikonja. Shape happens: Automatic feature manifold discovery in LLMs via supervised multi-dimensional scaling. *arXiv preprint arXiv:2510.01025*, 2025. URL <https://arxiv.org/abs/2510.01025>.
- [100] Alex Turner, Lisa Thiergart, David Udell, Gavin Leech, Ulisse Mini, and Monte MacDiarmid. Activation addition: Steering language models without optimization. *arXiv preprint arXiv:2308.10248*, 2023.
- [101] Hieu M. Vu and Tan Minh Nguyen. Angular steering: Behavior control via rotation in activation space. *Advances in Neural Information Processing Systems*, 2025.
- [102] Jonas Wallat, Abdelrahman Abdallah, Adam Jatowt, and Avishek Anand. A study into investigating temporal robustness of llms, 2025. URL <https://arxiv.org/abs/2503.17073>.
- [103] Jinjin Wang, Yuzhen Li, Jun Luo, and Hang Ye. Measuring time preference: Theory, methods, and applications. *Acta Psychologica*, 261:105928, 2025.
- [104] Rui Wang et al. Prospect theory fails for LLMs: Revealing instability of decision-making under epistemic uncertainty. *arXiv preprint arXiv:2508.08992*, 2025.
- [105] Yuqing Wang and Yun Zhao. Tram: Benchmarking temporal reasoning for large language models, 2024. URL <https://arxiv.org/abs/2310.00835>.
- [106] Zehong Wang, Fang Wu, Hongru Wang, Xiangru Tang, Bolian Li, Zhenfei Yin, Yijun Ma, Yiyang Li, Weixiang Sun, Xiusi Chen, and Yanfang Ye. Why reasoning fails to plan: A planning-centric analysis of long-horizon decision making in llm agents, 2026. URL <https://arxiv.org/abs/2601.22311>.
- [107] Jan Wehner, Sahar Abdelnabi, Daniel Tan, David Krueger, and Mario Fritz. Taxonomy, opportunities, and challenges of representation engineering for large language models, 2025. URL <https://arxiv.org/abs/2502.19649>.

- [108] Yotam Wolf et al. Tradeoffs between alignment and helpfulness in language models with steering methods. *arXiv preprint arXiv:2401.16332*, 2024.
- [109] Zhengxuan Wu, Aryaman Arora, Zheng Wang, Atticus Geiger, Dan Jurafsky, Christopher D Manning, and Christopher Potts. ReFT: Representation finetuning for language models. *Advances in Neural Information Processing Systems*, 37:63908–63962, 2024.
- [110] Zhengxuan Wu, Aryaman Arora, Atticus Geiger, Zheng Wang, Jing Huang, Dan Jurafsky, Christopher D Manning, and Christopher Potts. Axbench: Steering llms? even simple baselines outperform sparse autoencoders. *arXiv preprint arXiv:2501.17148*, 2025.
- [111] Kai Xie, Weichen Zhang, Jingwei Bi, Yiming Gu, Tao Luo, and Ce Liu. Spherical steering: Geometry-aware activation rotation for language models. *arXiv preprint arXiv:2602.08169*, 2026.
- [112] An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, Jianxin Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Chen, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Runji Wang, Shikai Liu, Shuang Luo, Tianhao Li, Tianyu Liu, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhe Yang, Zhiying Xu, and Zihan Qiu. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.
- [113] Zhuofan Josh Ying, Shauli Ravfogel, Nikolaus Kriegeskorte, and Peter Hase. The truthfulness spectrum hypothesis, 2026. URL <https://arxiv.org/abs/2602.20273>.
- [114] Hengyuan Zhang, Zhihao Zhang, Mingyang Wang, Zunhai Su, Yiwei Wang, Qianli Wang, Shuzhou Yuan, Ercong Nie, Xufeng Duan, Qibo Xue, Zeping Yu, Chenming Shang, Xiao Liang, Jing Xiong, Hui Shen, Chaofan Tao, Zhengwu Liu, Senjie Jin, Zhiheng Xi, Dongdong Zhang, Sophia Ananiadou, Tao Gui, Ruobing Xie, Hayden Kwok-Hay So, Hinrich Schütze, Xuanjing Huang, Qi Zhang, and Ngai Wong. Locate, steer, and improve: A practical survey of actionable mechanistic interpretability in large language models, 2026. URL <https://arxiv.org/abs/2601.14004>.
- [115] Jian-Qiao Zhu, Haijiang Yan, and Thomas L. Griffiths. Language models trained to do arithmetic predict human risky and intertemporal choice, 2025. URL <https://arxiv.org/abs/2405.19313>.
- [116] Jian-Qiao Zhu, Haijiang Yan, and Thomas L Griffiths. Steering risk preferences in large language models by aligning behavioral and neural representations. *arXiv preprint arXiv:2505.11615*, 2025.
- [117] Andy Zou, Long Phan, Sarah Chen, James Campbell, Phillip Guo, Richard Ren, Alexander Pan, Xuwang Yin, Mantas Mazeika, Ann-Kathrin Dombrowski, Shashwat Goel, Nathaniel Li, Michael J. Byun, Zifan Wang, Alex Mallen, Steven Basart, Sanmi Koyejo, Dawn Song, Matt Fredrikson, J. Zico Kolter, and Dan Hendrycks. Representation engineering: A top-down approach to ai transparency, 2025. URL <https://arxiv.org/abs/2310.01405>.

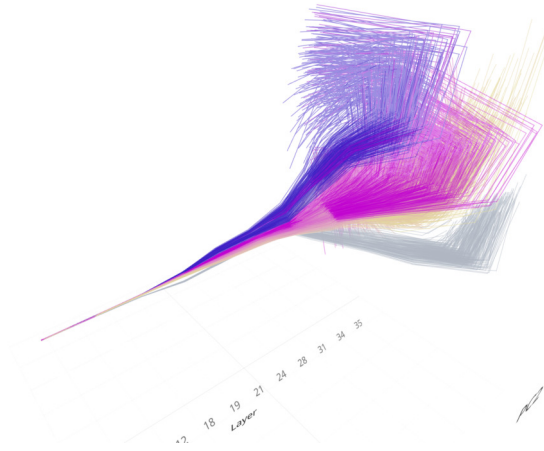
Appendices

Index of Appendices

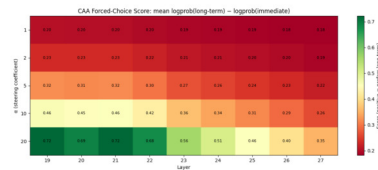
App.	p.	Paradigm	Content
Part 0: Groundwork			
A	22	n/a	Extended background: definitions and interpretability primitives.
B	24	n/a	Extended literature.
C	26	n/a	Methodology summary and overview of the extended methodologies.
D	28	n/a	Full experimental details.
E	29	n/a	Prompting settings and dataset construction.
F	34	n/a	Extended limitations and future work.
Part 1: Localize (ordered by method strength: probing \rightarrow attributional \rightarrow causal)			
G	36	Probing	Linear probing: 99.2% at L26, cross-dataset generalization.
H	39	Attr. contr.	EAP-IG attribution on contrastive prompts.
I	44	Causal param.	Activation patching: L21–24 attn, L31–35 MLP.
J	49	Causal contr.	Directional patching: symmetric effects, two-phase classification.
K	64	All	Cross-method convergence on layers 17–35.
Part 2: Characterize			
L	68	Parametric	PCA geometry: horizon \rightarrow preference transformation at turn boundary.
M	78	All	Latent vs. constrained: sparse attn vs. full subgraph.
N	80	Behavioral	Temporal discounting: LLMs 3–8 \times more patient than humans.
O	85	Behavioral	Behavioral coherence: order bias, instruct degradation.
P	99	Causal param.	Cross-model patching: circuit localizes at fractional depth 0.6–0.7 across scale.
Q	103	Probing	Error monitoring: subgraph encodes chain reliability orthogonally to time horizon.
Part 3: Intervene			
R	108	Contrastive	CAA steering: bidirectional, L19–22, probing–steering dissociation.
Part 4: Extended methodologies (same strength ordering as Part 1)			
S	114	n/a	Notation and key concepts.
T	115	Probing	Probing protocol and activation extraction.
U	116	Attr. contr.	EAP-IG methodology, bias controls, component taxonomy.
V	119	Causal param.	Activation patching setup, noise/denoise protocol.
W	121	Causal contr.	Directional patching on contrastive prompts.
X	122	Parametric	PCA geometry analysis pipeline.
Y	124	Behavioral	Kirby MCQ-27 instrument, decision boundary method.
Z	126	Behavioral	Behavioral coherence experiment design.
AA	128	Contrastive	CAA vector construction and steering setup.
AB	129	n/a	Worked case study: highly-formatted pair.



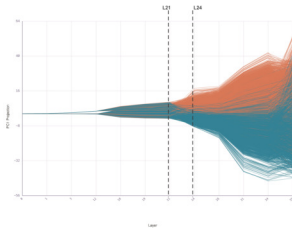
A visual tour of the appendices



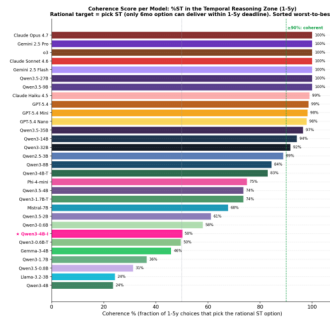
Appendix L, p. 68



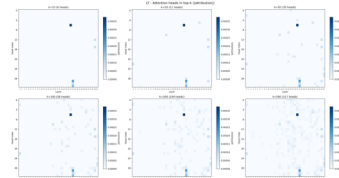
Appendix R, p. 108



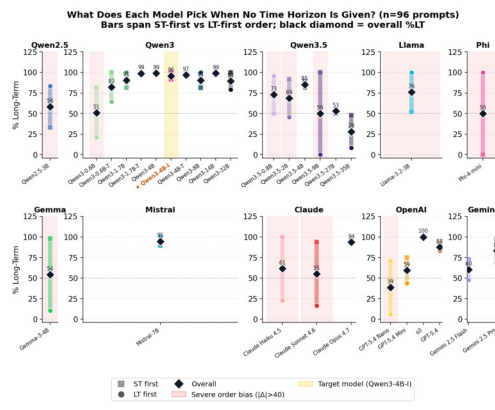
Appendix L, p. 68



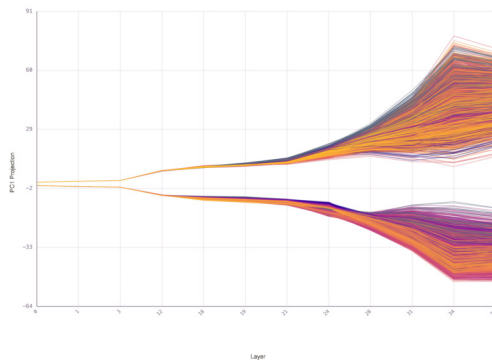
Appendix O, p. 85



Appendix H, p. 39



Appendix O, p. 85



Appendix L, p. 68

Part 0:

Groundwork

- **A.** Extended background
- **B.** Extended literature
- **C.** Methodology summary
- **D.** Experimental details
- **E.** Prompts
- **F.** Extended limitations and future work

Appendix A Extended background

This appendix preserves the full-length background that the main text condenses for space. It defines the temporal-preference concepts we use, introduces intertemporal choice as the empirical handle, and reviews the mechanistic-interpretability primitives our pipeline rests on: causal and attributional localization, representational geometry, and steering. The related-work discussion lives separately in Appendix B.

A.1 Temporal preference, horizon, and scope

We refer to **temporal preference** as the degree to which an agent values outcomes differently depending on when they occur [16, 103]. We use the term **time horizon** to denote the future moment at which outcomes are evaluated against an objective [23]. Because future events do not affect past outcomes, the time horizon also acts as a *constraint* on planning [89]. It would be *instrumentally* [59] or *means-end* [7] **incoherent** for the agent to choose actions that are incapable of causing effects by the specified deadline. Then, the **temporal scope** is the bounded interval of time over which an agent weighs the results according to its preference⁵.

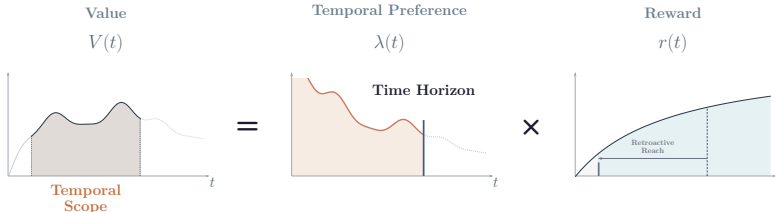


Figure A.1: The time horizon specifies when the consequences of a decision are assessed. The temporal scope is then bounded above by the time horizon.

In humans, these concepts have localized neural representations [51] that predict behavior [93], respond causally to intervention [28], and exhibit internal organization interpretable as a *functional role* [18].⁶ Our work asks whether temporal preference exists in an LLM in an analogous way: localized, predictive, causally efficacious, and geometrically organized.

A.2 Modeling behavior via intertemporal choice

We measure temporal preference through **intertemporal choice**: forced binary decisions between options that differ in reward and delay. This is the standard instrument in behavioral economics and neuroeconomics [58, 29, 51] because it isolates preference from effort, attention, and planning; separates reward from delay; and produces a single forced-choice token we can align across prompts and patch at the activation level. Each option i is defined as a tuple (r_i, t_i) , where $r_i \in \mathbb{R}^+$ denotes the reward and $t_i \in \mathbb{R}^+$ the delay until receipt. The subjective value of an option is the product of a temporal preference and the reward:

$$V(t) = \lambda(t) \cdot r(t) \tag{A.1}$$

Given two options $A = (r_A, t_A)$ and $B = (r_B, t_B)$, we predict the agent selects the option with the highest value:

$$i^* = \arg \max_{i \in \{A, B\}} \lambda(t_i) \cdot r_i \tag{A.2}$$

In the case of humans, temporal preference is often modeled using discount functions that capture our tendency to prefer immediate rewards over future ones [29, 35]. For AI agents, it is possible that different classes of functions better model their behavior [69]. Characterizing LLM behavior requires fitting a discount function via regression, assessing its stability across varying contexts, and benchmarking the resulting preferences against human intertemporal choice.

⁵Although not explored in this work, when rewards are *perishable* [4], the temporal scope is also bounded below by the *retroactive reach* [50, 71].

⁶Some authors argue that concepts are best modeled by geometric or topological spaces [30], a perspective that resonates with our geometric analysis of temporal representations in activation space.

A.3 Localizing a subgraph

The process of *subgraph localization* within an LLM involves identifying which components of the neural network are responsible for the behavior of interest. The gold standard is *causal localization* [33], which works by intervening within an LLM to measure the causal effect of specific components on the behavior of the model. We adopt **activation patching** [43] as our causal technique (Section Appendix I): replace one component’s activation with a counterfactual value from another input and measure the behavioral change. Attribution scores components via gradients and probing reads linearly decodable information, but neither intervenes on the forward pass; patching is the only one of the three that tests whether a component is causally necessary or sufficient for the output. Using the do-calculus notation [83], the patching intervention can be expressed as:

$$\Delta_i^{(l)}(x, x') = \mathbb{E}[Y \mid \text{do}(a_i^{(l)} = a_i^{(l)}(x')), X = x] - \mathbb{E}[Y \mid X = x] \quad (\text{A.3})$$

Unfortunately, performing targeted interventions is computationally expensive. As an alternative, *attributional localization* approximates causal localization [6]. We use **EAP-IG** [42], a gradient-based attribution method (Section Appendix H) that scores every head and MLP in a single backward pass. This makes a full-network scan tractable; the tradeoff is correlational estimates rather than causal guarantees. We also use **probes**, linear classifiers trained on a model’s internal activations [75, 57], to give us a complementary view by identifying which concepts a model encodes, where they emerge, and whether they are linearly represented.

Including more components in a subgraph explains more of the LLM’s behavior, but yields a larger, less interpretable picture. The full network trivially explains everything, and the empty subgraph explains nothing. Any useful circuit falls between these extremes, balancing behavioral coverage against subgraph size.

A.4 Visualizing representational geometry

The activation space within an LLM encodes concepts in internal representations [6]. A growing body of evidence has documented that many representations possess complex geometric structures [55, 24, 73, 38], beyond the global directions that the *Linear Representation Hypothesis* predicts [81]. Furthermore, recent work has also noticed that LLMs show local low-dimensional structure [92, 90, 61], and that even when representations are linear, they can change dramatically throughout generation [60].

These past findings motivate us to visualize the representational geometry within the localized subgraph as a way to understand what each component is doing. In our work, we apply Principal Component Analysis (PCA) [95] to examine how the temporal concepts of interest are represented within a lower-dimensional subspace.

A.5 Steering behavior with interventions

Steering refers to the control of an LLM’s behavior by directly intervening on its internal representations, rather than through prompting or training. Subgraph localization is not required for steering [5, 107], but localization generally improves precision, reduces side effects, and allows smaller intervention magnitudes [114]. In our work, we seek to understand the interventions in our subgraph through both a geometric and a behavioral perspective.

Appendix B Extended literature

Understanding temporal preference in LLMs requires drawing together the literature that has largely developed in isolation.

Temporal Representation. LLMs encode temporal and spatial coordinates as geometric objects recoverable via regression probes [37, 78, 46], forming circular, helical, and manifold structures [24, 53, 52, 41, 73, 99, 38] that obey psychophysical scaling laws [10, 55]. Linear decodability coexists with non-linear geometry because features are locally linear on globally curved manifolds [73, 81, 92, 88]. Yet the best-geometry layer is not the computational layer [9], causality is rarely established [44], and it is not known whether temporal representations causally drive downstream behavior across contexts the way emotion concepts do [96].

Temporal Reasoning and Planning. LLMs fail at temporal reasoning tasks despite encoding time geometrically [105, 25, 66, 102, 45], and lack continuous temporal grounding: they cannot track real-time deadlines even when discrete turn-based reasoning succeeds [91, 31, 15]. Evidence of temporal structure exists [80, 65, 12], and targeted fixes have been proposed [40, 94], but none connect temporal geometry to temporal decision-making. Separately, token-level lookahead over discrete sequence positions is detectable via probing [76, 97], but operates over next-token predictions rather than real-valued time horizons; both modes fail at long-horizon decisions [106, 11].

LLM Economic Behavior and Risk Preference. LLMs reproduce behavioral-economic biases [48, 47, 17, 13, 14, 63, 9] with unstable risk preferences [104]. Risk and time preferences can be steered neurally [116, 115] but entangle through discount factors [26, 74]. Temporal preferences have been studied only behaviorally [69]; whether they form a steerable activation-space direction, as shown for truthfulness [68, 113] and emotion [96], is open.

Steering Advancements. Representation engineering [117] has progressed from activation addition [100, 79] and representation interventions [109] through sparse dictionaries [19, 22] to geometric approaches [101, 111, 85]. Prompting still leads on many benchmarks [110, 54], and over-steering degrades helpfulness [108, 3]. Patching along continuous numeric directions produces monotonic output shifts [44], but static vectors assume a fixed concept direction; when the effective direction varies with context or curves through activation space, they become misaligned or unreliable [64, 87, 8].

No prior work has localized a subgraph functionally responsible for temporal preference, characterized the geometry of the causal representation, or steered along it. Table B.1 situates our contribution against the most directly comparable works on six axes.

Work	Concept	Causal subgraph	Geometry	Steering	Layer-localized steering	Human baseline	Latent + explicit
<i>Temporal representation</i>							
Gurnee and Tegmark [37]	space/time	✗	✗	✗	✗	✗	✗
Engels et al. [24]	days/months	✓	✓	partial	✗	✗	✗
Modell and Rubin-Delanchy [73]	theory	✗	✓	✗	✗	✗	✗
Gurnee et al. [38]	counting	✓	✓	✗	✗	✗	✗
<i>Temporal reasoning and planning</i>							
Wang et al. [106]	planning	✗	✗	✗	✗	✗	✗
Sehgal et al. [91]	deadlines	✗	✗	✗	✗	✗	✗
<i>LLM economic behavior and preference</i>							
Zhu et al. [116]	risk pref.	✗	✗	✓	✓	✗	✓
Mazyaki et al. [69]	temporal pref.	✗	✗	✗	✗	✓	✗
Horton et al. [48], Cook et al. [17]	economic bias	✗	✗	✗	✗	✓	✗
<i>Steering methods</i>							
Turner et al. [100], Panickssery et al. [79]	generic	✗	✗	✓	✓	✗	✗
Marks and Tegmark [68]	truthfulness	✓	✗	✓	✗	✗	✗
Sofroniew et al. [96]	emotion	✓	✓	✓	✗	✓	partial
This work	temporal pref.	✓	✓	✓	✓	✓	✓

Table B.1: Our contribution against the closest prior work on six axes: (i) whether a causal subgraph is identified, not just a probe direction; (ii) whether concept geometry is non-linear / curved; (iii) whether steering traverses a dimensional axis rather than a binary contrast; (iv) whether steering is layer-localized rather than applied uniformly; (v) whether outcomes are benchmarked against human behavior; (vi) whether both latent (no-horizon) and explicitly parameterized prompts are analyzed together. No prior work covers all six for temporal preference.

Appendix C Methodology summary

Our methodology follows three stages: *localize* the subgraph, *characterize* the representations, and *intervene*. Each stage has one or more dedicated experimental pipelines, and each pipeline has its own full-detail methodology appendix collected in Part 4 (see §C.5).

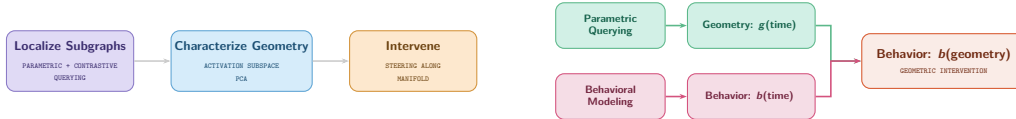


Figure C.1: Overview of our approach. Parametric querying could help us **reparametrize** our behavioral modeling as a function of activation-space geometry instead of an explicit time horizon.

C.1 Complementary localizations

We perform experiments with two different querying techniques applied to two corresponding prompting settings.

The first, **wide attribution**, combines contrastive querying with attribution patching and probing. Minimally-framed prompts elicit latent preferences without explicit temporal cues, while gradient-based approximations efficiently score component importance. This pipeline scales across samples, aggregating signal from hundreds of diverse prompts.

The second, **targeted intervention**, combines parametric querying with activation patching. Highly-structured prompts specify explicit time horizons, while direct interventions establish the causal effect. This pipeline disentangles causal relationships on carefully designed prompt variations.

C.2 Characterizing via geometry

We apply PCA [95] to residual-stream activations at subgraph nodes, examining how explicit time-horizon constraints (seconds to centuries) organize the activation manifold and whether latent preferences, elicited without any horizon cue, align to this geometry. We pay particular attention to the user-to-assistant turn transition, where the model converts off-policy context into on-policy generation. In principle, each prompt’s explicit horizon maps to a point on the manifold, opening the possibility of reparametrizing behavioral discount as a function of geometry rather than time (Figure C.1). We do not pursue this reparametrization fully here, but the geometry results in Appendix L lay the groundwork.

C.3 Behavioral analysis

We probe temporal preference at the behavioral level through two experiments. First, we administer the Kirby MCQ-27 [58] under multiple personas and response modes, fitting hyperbolic discount functions and introducing a *decision boundary method* that binary-searches the delayed reward to locate per-item indifference points. Second, we test behavioral coherence: whether the model’s choices respect the time-horizon constraint (Section 2). Choosing an option that cannot deliver within the specified deadline is instrumentally incoherent, and we systematically vary horizon, reward, order, label, and context to separate genuine temporal reasoning from surface heuristics. The behavioral experiments serve a dual role: they characterize the model’s temporal preferences independently of the mechanistic analysis, and they reveal the gap between the internal representation (rich, ordinal, geometrically structured) and the behavioral output (discrete, order-biased, partially incoherent).

C.4 Steering in the wild

We test causal control over temporal preference using Contrastive Activation Addition (CAA) [100, 79]. Logistic probes [75, 57] trained on the implicit dataset identify where

temporal orientation is linearly decodable; the probe direction at the best layer yields a steering vector $\hat{\mathbf{v}}_{\text{CAA}}$ injected as $\mathbf{h}^{(l)} \leftarrow \mathbf{h}^{(l)} + \alpha \cdot \hat{\mathbf{v}}_{\text{CAA}}$. We evaluate via forced-choice log-probability shifts and open-ended generation scored by an external LLM judge, sweeping layers and α to test for a probing–steering dissociation: whether the best layer for *reading* temporal preference differs from the best layer for *writing* it [43].

C.5 Overview of the extended methodologies

Full protocol-level details for each pipeline, including dataset construction, sample sizes, prompt formats, component-selection thresholds, and analysis procedures, are in Part 4 of the appendices. A reader looking for one specific experiment’s full methodology can jump directly to the corresponding appendix below; the four-part organization mirrors the localize/characterize/intervene pipeline:

Localize (four pipelines).

- Appendix U – EAP-IG attribution on minimally-framed contrastive prompts, with bias controls and the component-taxonomy thresholds used to define the candidate subgraph.
- Appendix V – Activation patching on parametric prompts: noise/denoise protocol, position alignment across horizons, and the metric used to score each (layer, component) cell.
- Appendix W – Directional activation patching on contrastive prompts, used to validate that the Part 1 subgraph is not a parametric-paradigm artifact.
- Appendix T – Logistic-probe training protocol, activation extraction, and the token-position correction applied to the contrastive dataset.

Characterize (three pipelines).

- Appendix X – PCA geometry pipeline: layer selection, variance-explained thresholds, and the turn-boundary analysis.
- Appendix Y – Kirby MCQ-27 instrument and the decision-boundary binary-search extension, including persona and response-mode conditions.
- Appendix Z – 30-model investment-coherence instrument: horizon \times reward \times order \times label \times context grid and parse protocol.

Intervene (one pipeline).

- Appendix AA – CAA vector construction from the best probing layer, the α -sweep protocol, and the forced-choice / open-ended evaluation setup.

Case study.

- Appendix AB – Worked token-level case study for a single highly-formatted prompt pair, tying the attribution, patching, and probing signals to specific tokens.

Appendix D Experimental details

Full details for each experiment are in the corresponding methodology appendix. All experiments can be run on a MacBook Pro (M4 Max, 48 GB). The full pipeline reproduces end-to-end within two weeks.

D.1 Why Qwen3-4B-Instruct-2507?

We select Qwen3-4B-Instruct-2507 [86], the non-thinking-only mode-specialized refresh of Qwen3-4B [112], for three reasons:

- **Non-thinking keeps cognition inside a fixed template.** The model operates exclusively in non-thinking mode: it never emits a `<think>...</think>` reasoning block, so the token positions we patch into are stable across clean and corrupted runs. All “cognition” happens inside the fixed prompt template, which is the alignment condition that activation patching and EAP-IG attribution both require. The hybrid-thinking Qwen3-4B would produce variable-length reasoning blocks that break this alignment.
- **Stable latent preference under perturbation.** Localization requires that the model’s answer does not flip under minor syntactic changes. Qwen3-4B-Instruct-2507 satisfies this: across the 30-model behavioral panel (Appendix O), it is among the most label-stable and context-stable checkpoints at its scale, while similarly-sized open-weight models drift under perturbation.
- **Tractable to sweep.** Qwen3-4B outperforms Qwen2.5-7B on most benchmarks and competes with Qwen2.5-14B-Instruct, Gemma-3-12B-IT, and Phi-4 [112], yet the 4B footprint lets us run the full attribution-plus-patching-plus-steering pipeline on a single MacBook.

The same mode specialization that enables this analysis also exposes the behavioral gap we study: the non-thinking variant collapses the hybrid-thinking checkpoint’s graded horizon curve into three discrete order-biased modes (Appendix O) even though its internal temporal geometry remains rich (Appendix L).

D.2 Datasets

- **Minimally-framed.** Minimally-framed A/B prompts: an *explicit* set (500 pairs, 25 categories) with overt temporal markers, and an *implicit* set (500 pairs, 10 categories) using only semantic framing. Both are counterbalanced across two orderings and seven label schemes.
- **Highly-formatted.** 4,588 investment intertemporal choice prompts with optional horizon constraints ranging from seconds to centuries.
- **Behavioral.** Kirby MCQ-27 administered under 8 conditions (2 personas \times 2 response modes), plus a binary-search decision-boundary extension.
- **Steering evaluation.** 20 held-out forced-choice and 13 open-ended prompts scored by an external LLM judge.

Appendix E Prompting settings

We use two prompting settings that probe temporal preference at different levels of abstraction. The *minimally-framed* setting is purely contrastive: it presents a binary choice between a short-horizon and a long-horizon option, with no explicit time or reward values. This captures temporal preference as a binary concept (present vs. future). The *highly-formatted* setting is both contrastive and parametric: it can elicit the same binary preference, but it also sweeps explicit time horizons from seconds to centuries, treating time as a continuous, dimensional concept. Together, the two settings let us study temporal preference both as a categorical distinction and as a graded quantity (Figure E.1).

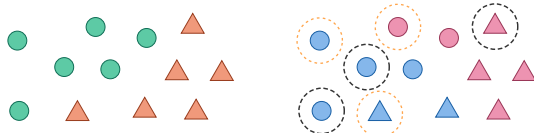


Figure E.1: The **contrastive** paradigm (left) treats temporal preference as a binary distinction, while the **parametric** paradigm (right) treats it as a dimensional concept by varying time horizon, reward magnitude, and delay.

E.1 Minimally-framed prompts

The minimally-framed setting queries the model’s temporal preference via forced-choice A/B prompts. Each prompt presents a scenario and two candidate responses (one short-horizon and one long-horizon) and the model selects a label. We construct two complementary datasets under this paradigm, probing the same underlying distinction under different cueing regimes.

Each pair is structured as $\{\text{question}, \text{immediate_choice}, \text{long_term_choice}, \text{category}\}$, with A/B label positions randomized ($\approx 50/50$ split) to control for positional bias. Both datasets comprise 500 paired examples balanced across categories. For probing experiments (Appendix T), a 300-pair subset of the implicit dataset is used as the primary training and evaluation set.

E.1.1 Explicit dataset (D_{explicit})

The explicit dataset contains 500 examples in which temporal scope is directly specified through overt temporal expressions. The contrast between options is signaled by phrases denoting near-term deadlines, short planning horizons, annual targets, or multi-year timeframes; the distinction is lexically explicit.

PROMPT

Innovation should aim for:

- (A) Products we can ship this quarter
- (B) Technologies that matter in 20 years

RESPONSE

(A)

Figure E.2: Example of an **explicit** minimally-framed prompt. The temporal contrast is carried by surface vocabulary (“this quarter” vs. “20 years”).

The dataset is evenly distributed across 25 topical categories with 20 examples per category, spanning planning, decision-making, problem-solving, investment, prioritization, goal-setting, action, vision, risk management, resource allocation, strategy formation, change management, innovation, leadership, communication, evaluation, learning, adaptation, hiring, product development, customer relations, financial planning, team building, market entry, and crisis response.

E.1.2 Implicit dataset (D_{implicit})

The implicit dataset contains 500 examples in which temporal scope is encoded through semantic framing rather than explicit temporal markers. The short-horizon response emphasizes immediate containment, execution, or preservation, whereas the long-horizon response emphasizes redesign, investment, transformation, or compounding effects, without directly invoking time-related language.

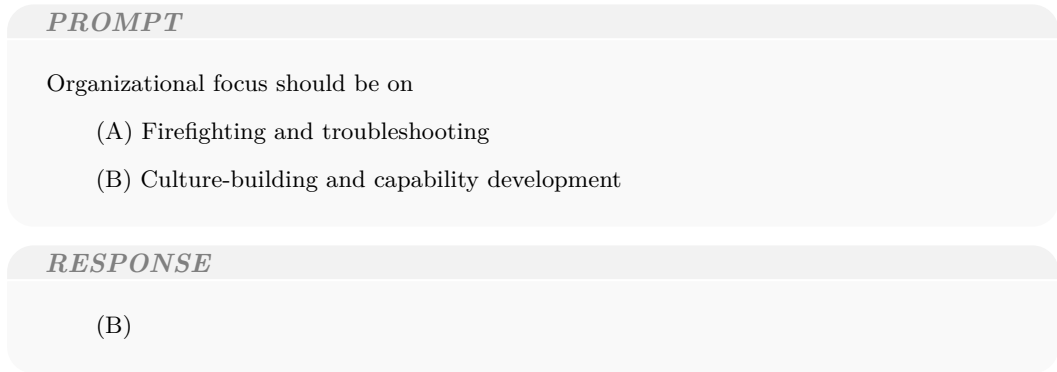


Figure E.3: Example of an **implicit** minimally-framed prompt. Neither option contains temporal vocabulary; the distinction is carried entirely by semantic framing.

The dataset is balanced across 10 abstract contrast categories with 50 examples per category: `crisis_vs_foundation`, `harvest_vs_cultivate`, `execute_vs_design`, `react_vs_anticipate`, `preserve_vs_transform`, `tactical_vs_strategic`, `consume_vs_invest`, `fix_vs_build`, `survive_vs_thrive`, and `capture_vs_compound`. This design isolates temporal reasoning from lexical cues, enabling evaluation of whether the model relies on semantic abstractions rather than explicit time indicators.

E.1.3 LLM-Assisted Generation and Validation

To construct the D_{explicit} and D_{implicit} datasets at scale while maintaining strict control over linguistic variables, we used a multi-stage, LLM-assisted generation and verification pipeline. The initial candidate pairs for both datasets were generated using **Claude Sonnet 4.6**. To ensure these generated pairs adhered to our contrastive constraints and were free of unintended confounds, we implemented an automated validation framework. Each candidate pair was independently evaluated and scored by both **Claude Sonnet 4.6** and **Gemini 3 Flash**.

The models verified the pairs across four strict dimensions: *lexical confounds* (ensuring the implicit set contained absolutely no explicit temporal keywords and that vocabulary complexity was balanced), *surface form* (matching character length and grammatical structure), *semantic confounds* (aligning formality, hedging, specificity, and sentiment), and *content validity* (guaranteeing a unidimensional, unambiguous distinction between immediate and long-term choices). Both models scored these factors on a 1 to 5 scale. Pairs falling below the acceptable quality threshold (an average score < 3.5) were iteratively revised or discarded. Finally, to eliminate positional bias, the presentation order of the immediate and long-term options (A/B) was randomized in the finalized datasets.

E.2 Highly-formatted prompts

The highly-formatted setting uses structured prompts with explicit section markers that ensure consistent token-position alignment across prompt variants. Each prompt contains the following fields:

- **SITUATION:** Domain context (e.g., household financial planning)
- **TASK:** The role, task, and two labeled options with reward amounts and time horizons
- **OBJECTIVE:** Instruction to deliberate
- **CONSTRAINT:** An explicit time-horizon constraint (e.g., “1 year,” “5 centuries,” or omitted for no-constraint prompts)
- **ACTION:** Instruction to select one option
- **FORMAT:** Response template specifying “I choose: <label>” and “My reasoning: <text>”

The shared structural markers serve as *anchors* for a position mapping that aligns token indices between prompts of different lengths. When clean and corrupted prompts differ in token count (e.g., because different time horizons require different numbers of tokens), the position mapping uses these anchors to interpolate correctly during activation patching (Appendix V).

PROMPT

SITUATION: Plan for the future of the household based on the stated objectives and constraints.

TASK: You, the head of the household, are tasked to choose the best investment:

- a) 20,000 dollars in 6 months.
- b) 500,000 dollars in 10 years.

OBJECTIVE: Think deeply about which option is preferable.

CONSTRAINT: You must select the option that provides the greatest benefit for this time horizon: 8 months.

ACTION: Select one of the two options. Provide reasoning on why this choice was made.

FORMAT: Respond in this format:

I choose: <a> or b>. My reasoning: <reasoning in 1-3 sentences>

RESPONSE

I choose: b). My reasoning: Although the immediate...

Figure E.4: Example of a highly-formatted prompt. The **constraint** section is optional; omitting it queries the model’s latent preference.

E.2.1 Parametric variation

The experiment configuration sweeps over several axes to systematically vary the temporal context:

- **Reward range:** Logarithmic steps between a minimum and maximum (e.g., \$1,000–\$100,000)
- **Time range:** Logarithmic steps for both short-term and long-term options
- **Time horizons:** 17 values from null (no constraint) through seconds, hours, days, weeks, months, years, decades, to centuries

This yields a grid of contrastive pairs that disentangle the effects of reward magnitude, delay, and horizon constraint on internal representations.

```

{
  "name": "investment_geometry",
  "context": {
    "reward_unit": "dollars",
    "role": "the head of the household",
    "situation": "Plan for the future of the households.",
    "task_in_question": "choose the best investment",
    "domain": "finance"
  },
  "options": {
    "short_term": {
      "reward_range": [1000, 100000],
      "time_range": [
        {"value": 1, "unit": "days"},
        {"value": 20, "unit": "years"}
      ],
      "reward_steps": [2, "logarithmic"],
      "time_steps": [5, "logarithmic"]
    },
    "long_term": {
      "reward_range": [1000, 100000],
      "time_range": [
        {"value": 1, "unit": "years"},
        {"value": 100, "unit": "years"}
      ],
      "reward_steps": [2, "logarithmic"],
      "time_steps": [5, "logarithmic"]
    }
  },
  "time_horizons": [
    null,
    {"value": 1, "unit": "seconds"},
    {"value": 1, "unit": "hours"},
    {"value": 1, "unit": "days"},
    {"value": 1, "unit": "week"},
    {"value": 1, "unit": "months"},
    {"value": 2, "unit": "months"},
    {"value": 6, "unit": "months"},
    {"value": 1, "unit": "years"},
    {"value": 3, "unit": "years"},
    {"value": 5, "unit": "years"},
    {"value": 1, "unit": "decades"},
    {"value": 3, "unit": "decades"},
    {"value": 5, "unit": "decades"},
    {"value": 1, "unit": "centuries"},
    {"value": 2, "unit": "centuries"},
    {"value": 5, "unit": "centuries"}
  ]
}

```

Figure E.5: Example configuration for the `investment_geometry` scenario. Each scenario defines a context, short- and long-term option ranges, and a set of time horizons spanning seconds to centuries.

E.3 Comparison of prompting settings

Table E.1 summarizes the complementary strengths of the two settings. The minimally-framed setting is better suited for probing latent preferences under naturalistic conditions, while the highly-formatted setting enables controlled parametric sweeps and richer mechanistic analysis.

	Minimally-framed	Highly-formatted
Paradigm	Contrastive only (binary: short vs. long)	Contrastive + parametric (binary preference <i>and</i> continuous time horizon)
Concept type	Binary (present vs. future)	Dimensional (seconds to centuries)
Prompt structure	No explicit time or reward; model infers temporality from semantic framing	Explicit reward amounts, delays, and a constraint field specifying the horizon
Validity	Closer to on-policy; low demand characteristics	Closer to off-policy; structured scaffolding may anchor the model
Mechanistic use	Attribution, probing, CAA vector construction	Attribution, activation patching with token-level position mapping, geometry analysis
Behavioral modeling	Binary preference only; applies to any domain	Discount curves, comparison to human baselines, reparametrization via activation geometry

Table E.1: Comparison of the two prompting settings. The minimally-framed setting probes latent binary preference; the highly-formatted setting adds parametric control over the time dimension.

Appendix F Extended limitations and future work

Time is a complex and entangled concept. Our work is merely a starting point.

- **Finer localization.** Full circuit tracing [114, 34] would identify atomic components and their information flow. Our EAP-IG analysis shows attribution mass distributed across many nodes, and whether this reflects genuine distribution or a methodological limitation remains open.
- **Domain generalization and dataset provenance.** Our approach has several limitations: the pipeline uses only financial scenarios, so findings may not generalize to other domains (health, career); contrastive labels were synthetically assigned and lack human validation; and the steering vector, derived from controlled off-policy settings, may capture correlated features rather than purely temporal preference.
- **Scaling across models and variants.** We study only Qwen3-4B-Instruct-2507; replicating across families and scales would test whether the subgraph location and the probing–steering dissociation generalize. Comparing this distilled, non-thinking variant against its thinking counterpart (Qwen3-4B) is particularly compelling: our behavioral analysis shows that chain-of-thought dramatically alters temporal preference, but whether reasoning reorganizes the underlying subgraph is unexplored.
- **Richer parameterization and concept interactions.** Our parametric querying maps time horizon but could extend to reward magnitude, risk, role, and domain to parameterize the full intertemporal choice space. Temporal preference likely interacts with representations of risk [116, 74], emotion [96], and urgency, but we treat the subgraph in isolation. Moreover, all experiments are single-turn, yet temporal preference matters most in multi-turn and agentic settings where representations may shift across turns [60].
- **Non-linear steering.** Our linear CAA vector approximates a curved manifold; output quality degrades at $|\alpha|=60$. Methods that follow the manifold’s curvature [87, 64, 85] could enable stronger, cleaner interventions.

Part 1:

Where is temporal preference?

- **G.** Linear probing
- **H.** Attributional contrastive
- **I.** Causal parametric
- **J.** Causal contrastive
- **K.** Cross-method convergence

Appendix G Contrastive linear probing results

The four experiments above localized temporal preference through attribution and causal intervention. Here we take a complementary approach: training logistic regression probes on residual-stream activations to ask *where* the model linearly encodes the short/long distinction (methodology in Appendix T). Unlike the previous methods, probing does not measure causal effect but rather the readability of a concept at each layer. This distinction will prove important: the best probing layer turns out to differ from the best intervention layer.

G.1 Layer-by-Layer Probe Accuracy

Logistic regression probes were trained at each of the 36 layers using the protocol described in Appendix T.

Metric	Result
Best layer	26
Best test accuracy	99.2%
Signal above chance	+52.3 pp
Cross-dataset generalization	Yes (see Section G.4)

Table G.1: Summary of probing results on D_{implicit} .

Accuracy rises steadily from $\sim 80\%$ at layer 0 to a plateau above 95% around layer 17, reaching 99.2% at layer 26. The monotonic increase across layers is consistent with the model progressively refining a linear temporal representation in deeper layers.

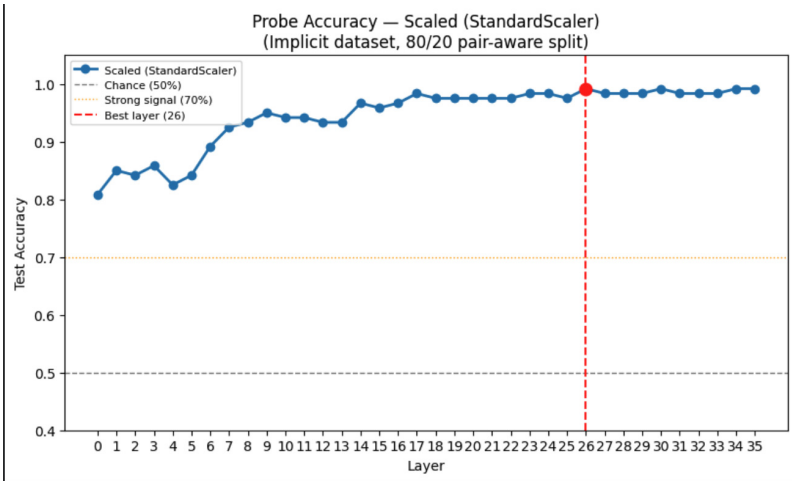


Figure G.1: Test accuracy of scaled logistic regression probes across all 36 layers on D_{implicit} (80/20 pair-aware split). Dashed lines indicate chance (50%) and the strong-signal threshold (70%). The best layer (26) is marked.

G.2 Shuffled-Label Control

To confirm that probe accuracy reflects genuine temporal structure rather than geometric artifacts of the activation space, we train 10 probes per layer on randomly permuted labels using the same scaled activations. Shuffled accuracy was approximately 50% at every layer, including layer 0. The gap between real-label accuracy ($\sim 80\text{--}99\%$) and shuffled-label accuracy ($\sim 50\%$) at every layer confirms that the signal is a learned property of the temporal concept, not an intrinsic property of the activation geometry.

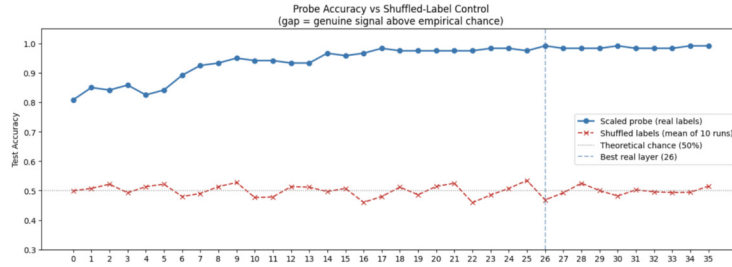


Figure G.2: Probe accuracy vs. shuffled-label control across all 36 layers. Real-label probes (blue) rise to 99.2% at layer 26, while shuffled-label probes (orange) remain at chance ($\sim 50\%$) throughout.

G.3 Representation Geometry (PCA)

PCA analysis reveals an important asymmetry between the two datasets:

- **Implicit dataset:** No separation is visible in the top two principal components (PC1 explains only 2–5% of variance). The temporal direction is subtle and occupies dimensions that PCA discards.
- **Explicit dataset:** Clear separation is visible in PCA (PC1 = 9.5% at layer 0), driven by surface vocabulary differences between short-term and long-term choices.

This result is significant: PCA fails to detect the temporal concept in the implicit dataset, yet the supervised probe succeeds at 99.2%. The temporal direction is real but non-obvious; it requires supervised search to find a direction that unsupervised methods miss. This is consistent with findings on the non-trivial geometry of concept representations in LLMs [24, 68].

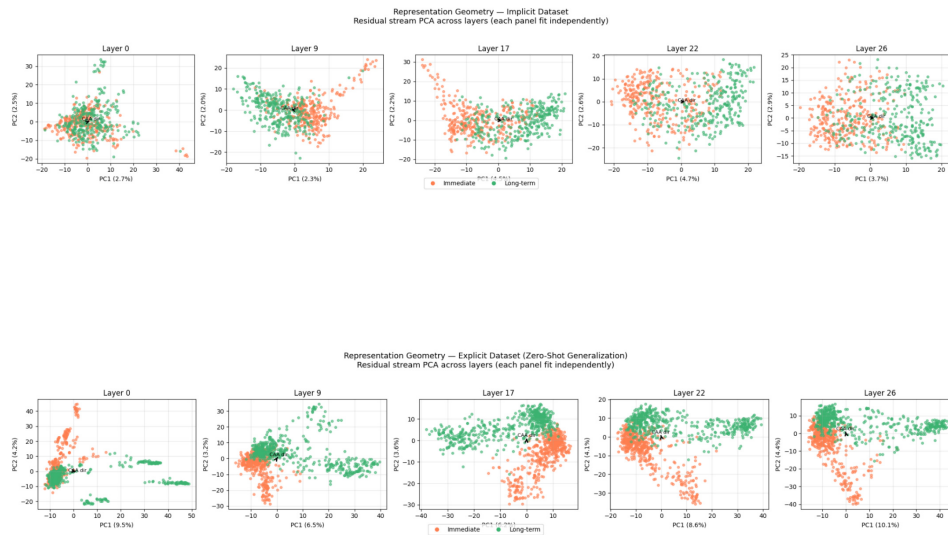


Figure G.3: PCA projections of layer activations for the implicit (top) and explicit (bottom) datasets at selected layers. The implicit dataset shows no visible separation in the top two PCs, while the explicit dataset shows clear clustering driven by surface vocabulary.

G.4 Cross-Dataset Generalization

Probes trained on D_{implicit} were evaluated zero-shot on D_{explicit} (different vocabulary, same underlying concept). This tests whether the probe has learned a genuine temporal direction rather than vocabulary-specific features. The saved `StandardScaler` from training is re-applied to the explicit activations before scoring.

Cross-dataset accuracy tracks the within-dataset accuracy closely across all layers, confirming that the probe direction generalizes from implicit semantic cues to explicit temporal markers. At the best layer (26), implicit test accuracy is 99.2% and cross-dataset accuracy on D_{explicit} remains above 95%.

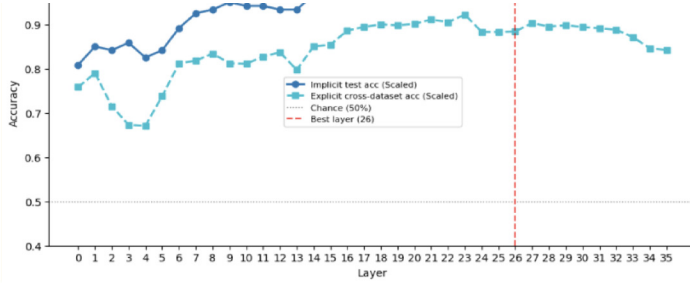


Figure G.4: Cross-dataset generalization: probes trained on D_{implicit} evaluated zero-shot on D_{explicit} . Accuracy tracks closely across all layers, confirming the probe captures a genuine temporal direction rather than dataset-specific features.

G.5 Summary

Contrastive probing confirms that `Qwen3-4B-Instruct-2507` maintains a linear temporal direction in its residual stream. The key findings are:

1. **Strong linear signal.** A logistic regression probe achieves 99.2% test accuracy at layer 26, with accuracy rising monotonically across layers.
2. **Shuffled control rules out artifacts.** Probes trained on permuted labels remain at chance ($\sim 50\%$) at every layer, confirming that the signal reflects genuine temporal structure (Section G.2).
3. **PCA misses it; supervised search finds it.** The temporal direction is not visible in the top principal components of the implicit dataset, yet a supervised probe recovers it with near-perfect accuracy (Section G.3). The concept is real but geometrically subtle.
4. **Cross-dataset generalization.** Probes trained on implicit cues transfer zero-shot to explicit temporal markers, confirming that the learned direction captures a genuine temporal concept rather than vocabulary-specific features (Section G.4).

An important dissociation emerges when comparing these probing results with the steering experiments in Appendix Appendix R: probing accuracy peaks at layer 26, while steering is most effective at layers 19–22. This gap suggests that the layers where the model most cleanly *represents* temporal preference are not the same layers where *intervening* on that representation most strongly influences downstream behavior. We discuss possible explanations for this probing–steering dissociation in Section R.3.

Appendix H Attributional contrastive results

Our first approach to localizing temporal preference uses gradient-based attribution (EAP-IG) on the minimally-framed contrastive prompts, where the model chooses between a short-horizon and a long-horizon option with no explicit time vocabulary. This is the cheapest localization method: it approximates causal effect via gradients rather than direct intervention, and the contrastive prompts are short and semantically controlled. The tradeoff is that the signal may be noisier than causal methods, so we treat the results here as a selection prior rather than ground truth (methodology in Appendix U).

The attribution reveals a candidate subgraph comprising approximately 0.125% of all nodes, concentrated in layers 21–35. However, as we show below, the circuit is highly diffuse: no individual component accounts for more than 0.1% of the total attribution mass. The layers that emerge here (particularly L24 for attention) will reappear consistently across the causal and probing experiments that follow (Appendix I, Appendix J, Appendix G).

H.1 Attribution Score Distribution

Figure H.1 shows that attribution mass is distributed across a large number of nodes: the distribution is neither power-law nor exponential, and the vast majority of components have near-zero attribution scores. This poses a fundamental challenge for top- k selection, as there is little theoretical justification for any particular cutoff when the score distribution lacks a natural elbow or gap.

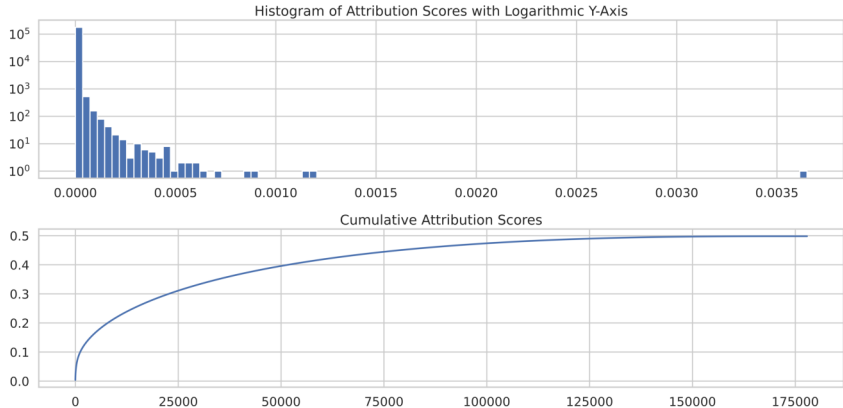


Figure H.1: (top) Histogram of attribution scores with logarithmic y-axis. The distribution is **not** approximated by either a power law or an exponential. (bottom) Cumulative logit-normalized component attribution scores for variant (A) for the canonical option order with respect to the short-term concept.

H.2 Limitations of EAP-IG for This Circuit

The attribution results reveal that temporal preference is likely mediated by a highly *diffuse* circuit rather than a sparse, localizable one. Even the highest-scoring individual components explain less than 0.1% of the total attribution mass individually. The sheer number of low-scoring components dominates the distribution, making top- k selection inherently noisy: it is unclear whether selected nodes are genuinely temporal-preference components or statistical artifacts of aggregation over hundreds of prompt variations.

Despite these limitations, the EAP-IG results provide a useful *signal* when interpreted alongside independent methods. In particular, the layers that emerge as high-attribution under EAP-IG (e.g., L24 for attention) overlap with layers identified by activation patching (Appendix I) and CAA steering (Appendix R). This convergence across independent methodologies suggests that the layer-level localization is genuine even though component-level identification via EAP-IG alone is unreliable.

We therefore treat EAP-IG not as a circuit-identification tool in the traditional sense, but as a *selection prior*: it restricts the search space to nodes enriched for temporal signal, which we then characterize through representational geometry (Appendix L) and probing (Appendix G). Our analysis focuses on representational structure rather than on isolating a minimal causal mechanism, and uses activation patching (Appendix I) to establish causal claims independently.

H.3 Layer Distribution of Top-k Components

Despite the diffuse nature of the overall attribution distribution, a clear layer-level pattern emerges: temporal-preference signal concentrates in the mid-to-upper layers (approximately layers 21–35). This concentration is robust across different values of k and holds for both attention and MLP components. Critically, this layer range converges with the layers identified independently by activation patching (Appendix I) and CAA steering (Appendix R), providing cross-method validation that temporal preference processing is genuinely localized to this subgraph.

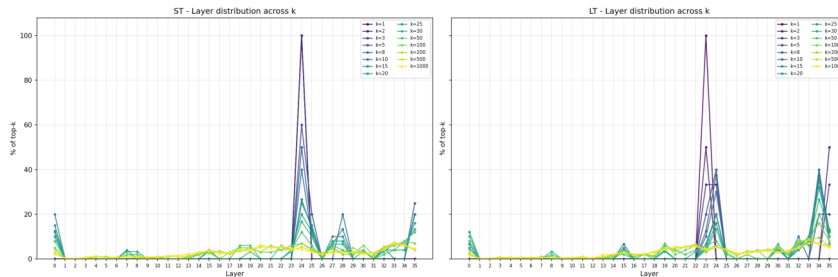


Figure H.2: Layer-wise distribution of top- k attributed components. Attribution mass concentrates in layers 21–35, with attention heads peaking around L24 and MLP neurons concentrated in the upper layers (L31–L35). This layer profile is stable across values of k , indicating genuine localization rather than an artifact of the threshold.

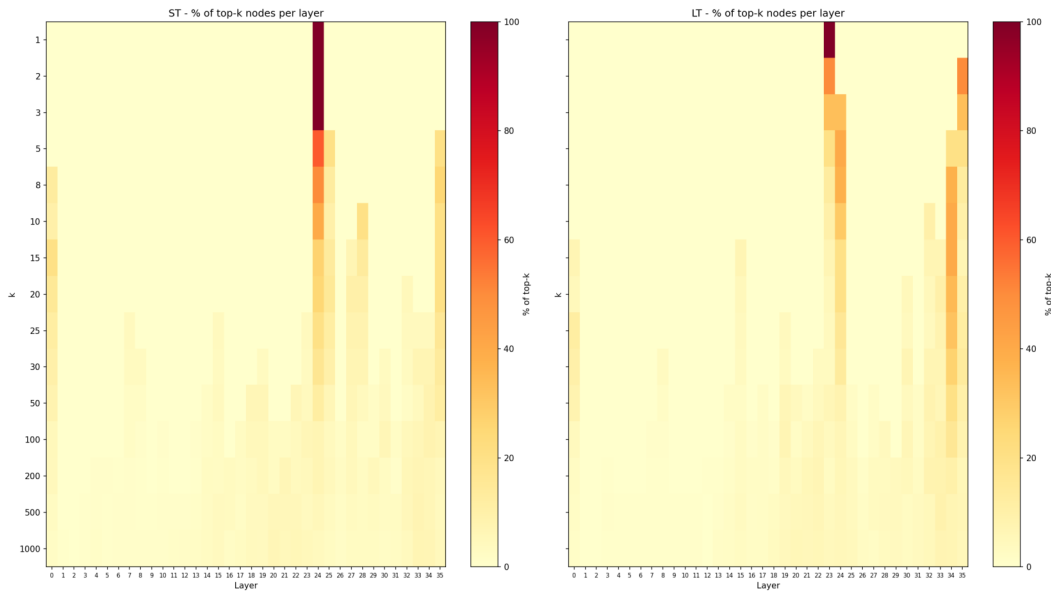


Figure H.3: Heatmap of top- k component counts per layer, broken down by component type. Attention heads are enriched in layers 21–26, while MLP neurons dominate in layers 31–35, consistent with a two-stage pattern where attention layers carry temporal information and later MLP layers refine it.

Figure H.4 complements the top- k count analysis by showing mean attribution scores per layer. The mean score profile confirms that the layers 21–35 concentration is not merely a consequence of having more components selected; these layers also carry higher per-component attribution mass on average.

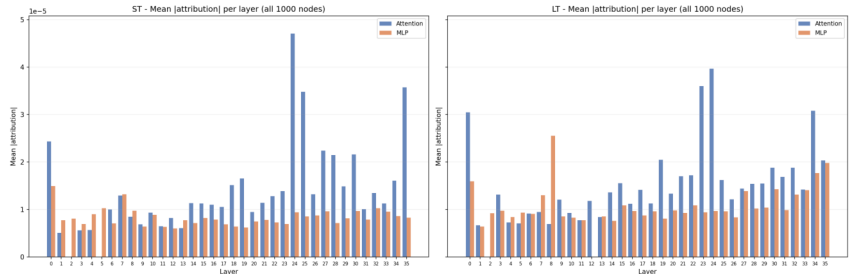


Figure H.4: Mean attribution score by layer. Layers 21–35 show elevated per-component scores, confirming that the mid-to-upper layer concentration reflects genuinely higher attribution rather than a selection artifact. The peak around L24 for attention aligns with activation patching results identifying L24_attn as the highest-effect attention component (Appendix I).

H.4 Attention vs. MLP Contributions

Decomposing the top- k attributed components by type reveals a division of labor between attention heads and MLP neurons. Attention heads account for the majority of highly-attributed components, consistent with their role in routing information across token positions, while MLP neurons contribute a smaller but distinct share concentrated in the upper layers. This attention-dominated pattern is consistent with temporal preference relying on contextual integration across the prompt rather than on purely local feature computation.

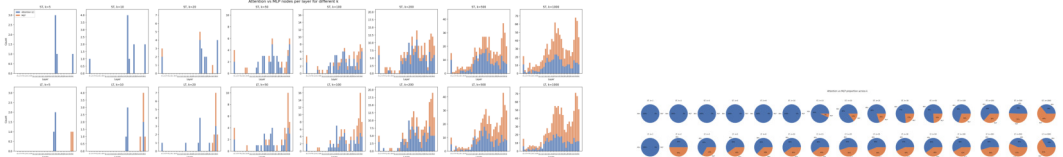


Figure H.5: Left: attention heads dominate the top- k attributed components, but the MLP share grows at larger k . Right: overall attribution mass split. Attention carries the majority, reinforcing that temporal-preference computation is primarily mediated by cross-position information flow.

The heatmaps below reveal which specific heads and MLP neurons carry the signal. For attention, a small number of heads in layers 21–26 stand out, while the MLP signal is more diffuse across neurons in layers 31–35. This spatial separation (attention in mid-layers, MLP in upper layers) is consistent with a two-phase computation: attention heads first integrate temporal context, then MLP layers transform this into the output representation.

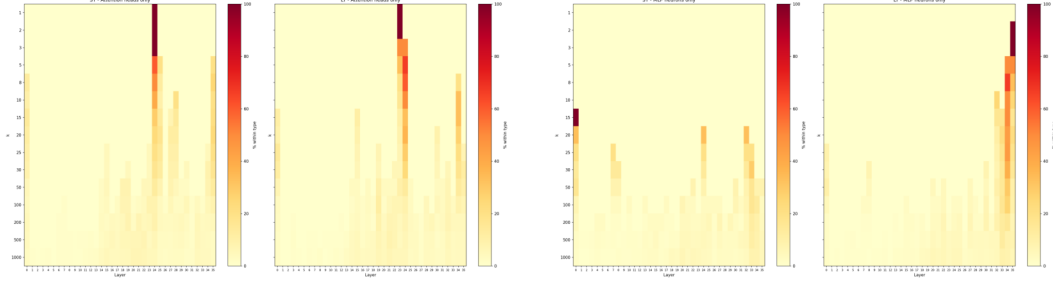


Figure H.6: Attribution heatmaps for attention heads (left) and MLP neurons (right) across layers. Attention: a sparse set of heads in layers 21–26 carries disproportionate attribution, with L24 heads showing the strongest signal, converging with activation patching results (Appendix I). MLP: attribution is concentrated in the upper layers (L31–L35) and more evenly distributed across neuron indices, suggesting distributed rather than sparse MLP processing.

H.5 Short-Term vs. Long-Term Component Comparison

A key question is whether short-term and long-term temporal preferences are processed by the same components or by specialized subpopulations. We compare the top- k attributed components for the short-term concept against those for the long-term concept. The results reveal partial but incomplete overlap: many components contribute to both concepts, but each concept also recruits specialized nodes. This pattern is consistent with a shared temporal-processing backbone augmented by concept-specific refinement, supporting the paper’s claim that temporal preference is a structured rather than monolithic representation.

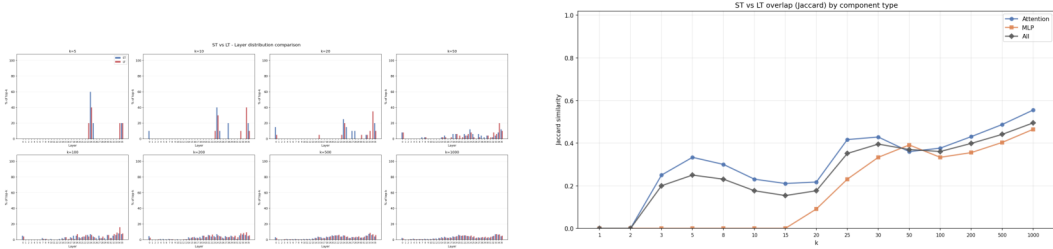


Figure H.7: Left: short-term and long-term attributed components show partial overlap. Short-term attribution peaks at L24, long-term shifts toward L22. Right: Jaccard similarity between ST and LT top- k sets increases with k , from concept-specific nodes at small k to a shared temporal backbone at larger k .

At small k , the overlap is low, indicating the most important components are concept-specific. As k grows, overlap increases, reflecting shared temporal processing consistent with the geometric separation in Appendix L.

H.6 Individual Component Analysis

While the diffuse distribution of attribution scores limits confidence in any single component (see Section Appendix H limitations discussion above), examining the highest-scoring individual nodes provides a useful sanity check. The top-ranked nodes cluster in the same mid-to-upper layer range identified by layer-level analysis, and the highest-attribution attention heads fall in L22–L24, precisely the layers flagged by activation patching as causally important. However, even the top-ranked individual components account for less than 0.1% of total attribution mass, underscoring why we treat EAP-IG as a selection prior rather than a definitive circuit-identification tool.

Finally, the full attention head matrices (Figure H.9) provide a detailed view of which heads matter for each concept. The short-term matrix shows concentrated signal in a few heads

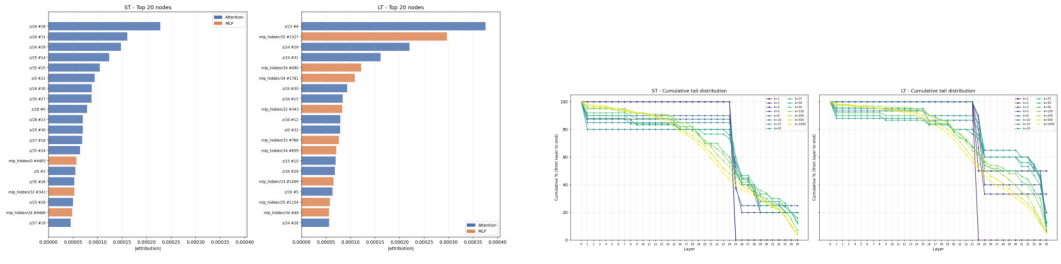


Figure H.8: Left: top individual nodes ranked by attribution score; the highest-ranked are attention heads in L22–L24, with no single component exceeding 0.1% of total mass. Right: cumulative tail distribution; attribution mass accumulates slowly, confirming a diffuse circuit ($\sim 0.125\%$ of all nodes).

around L24, while the long-term matrix distributes attribution more broadly across L22–L26. This asymmetry suggests that short-term preference relies on a slightly more focused set of attention heads, whereas long-term preference recruits a wider subnetwork.

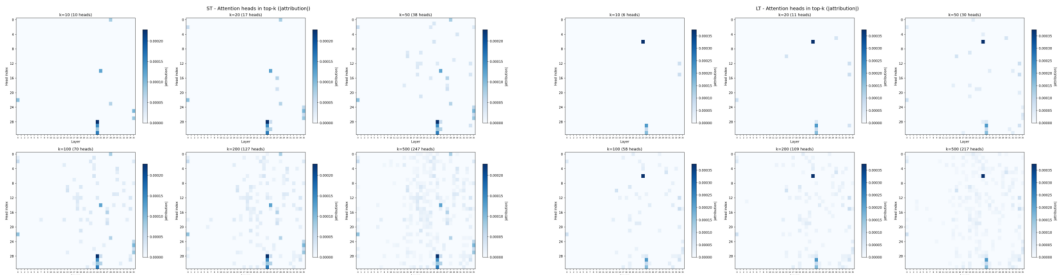


Figure H.9: Attention head attribution matrices for short-term (left) and long-term (right) concepts. Short-term attribution is concentrated in a sparse set of L24 heads, while long-term attribution is distributed more broadly across L22–L26, revealing an asymmetry in circuit structure between the two temporal concepts.

Appendix I Causal parametric results

The attribution results in Appendix H flagged layers 21–35 but could not establish causal effect. Here we apply the gold standard: activation patching on $n = 71$ highly-formatted parametric contrastive pairs, directly replacing component activations with counterfactual values (methodology in Appendix V). Where EAP-IG approximates, patching measures the actual behavioral consequence of intervention.

The results sharpen the picture considerably. A sparse set of four components, `L24_attn`, `L21_attn`, `L35_mlp`, and `L31_mlp`, account for the majority of the causal effect, clearly separated from the rest. L24 attention, the same layer flagged by EAP-IG, emerges as the single most important component under both denoising and noising.

I.1 Component importance ranking

We begin with a direct ranking of individual components by their causal effect size. Figure I.1 shows the top 20 components sorted by the mean of their denoising recovery and noising disruption scores across all contrastive pairs.

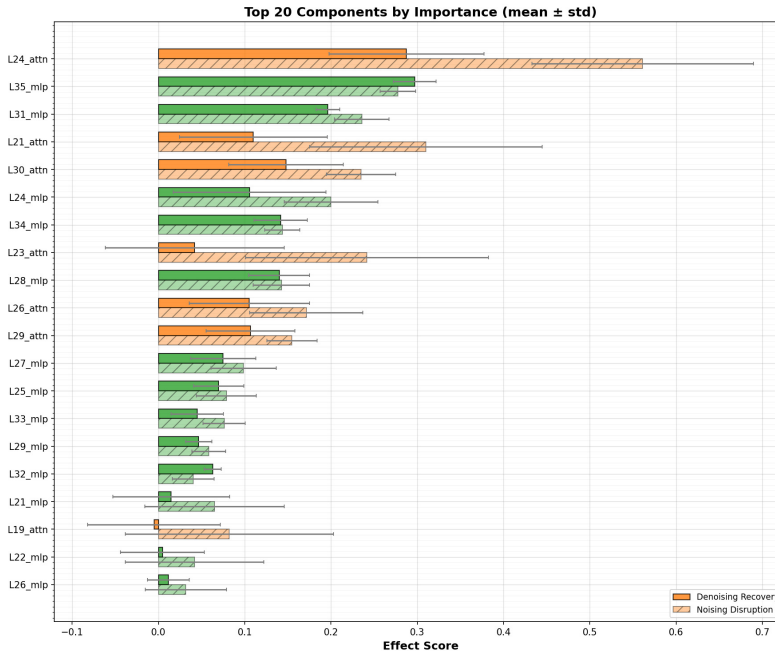


Figure I.1: Top 20 components ranked by mean effect score (denoising recovery and noising disruption, with standard deviation across contrastive pairs). `L24_attn` and `L21_attn` dominate, with noising disruption scores exceeding 0.5, indicating that corrupting these components alone is sufficient to substantially shift the model’s temporal preference. Among MLP components, `L35_mlp` and `L31_mlp` rank highest. The fifth-ranked component (`L30_attn`) has roughly half the effect of `L24_attn`, clearly separating the top four from the rest.

The ranking reveals a clear separation between a small number of high-effect components and a long tail of modest contributors. Attention components dominate the top of the list, with `L24_attn` showing the largest effect under both denoising and noising. The most causally important MLP components (`L35_mlp` and `L31_mlp`) rank among the top four overall, with effect sizes comparable to `L21_attn`. The asymmetry between denoising recovery and noising disruption is particularly pronounced for the top attention layers: `L24_attn` and `L21_attn` show much higher noising disruption than denoising recovery, suggesting these components are more necessary than sufficient: corrupting them degrades performance substantially, but restoring them alone does not fully recover clean behavior.

I.2 Marginal contribution analysis

Figure I.2 examines the marginal contribution of each layer, defined as the difference in residual stream activations before and after the layer ($\text{resid_post}[L] - \text{resid_pre}[L]$). This isolates each layer’s additive contribution to the residual stream.

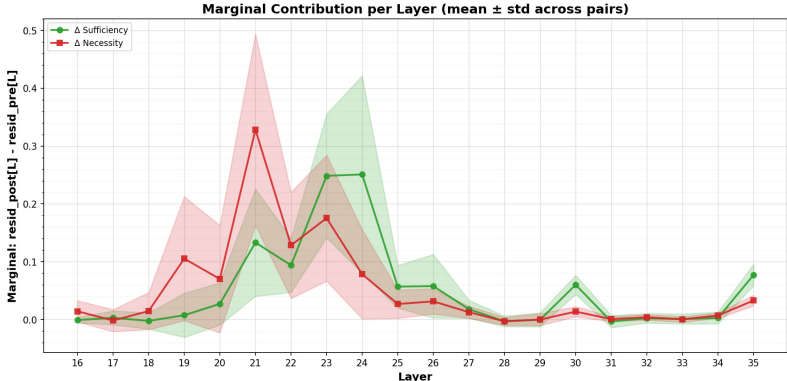


Figure I.2: Marginal contribution per layer (mean \pm standard deviation across contrastive pairs), showing sufficiency (denoising recovery, green) and necessity (noising disruption, red). Sufficiency peaks sharply at layers 21–24, with layer 22 showing the single highest spike. Necessity is flatter and lower, reflecting the distributed nature of disruption. The high variance in layers 20–25 reflects the sensitivity of these layers to the specific temporal framing used in each contrastive pair.

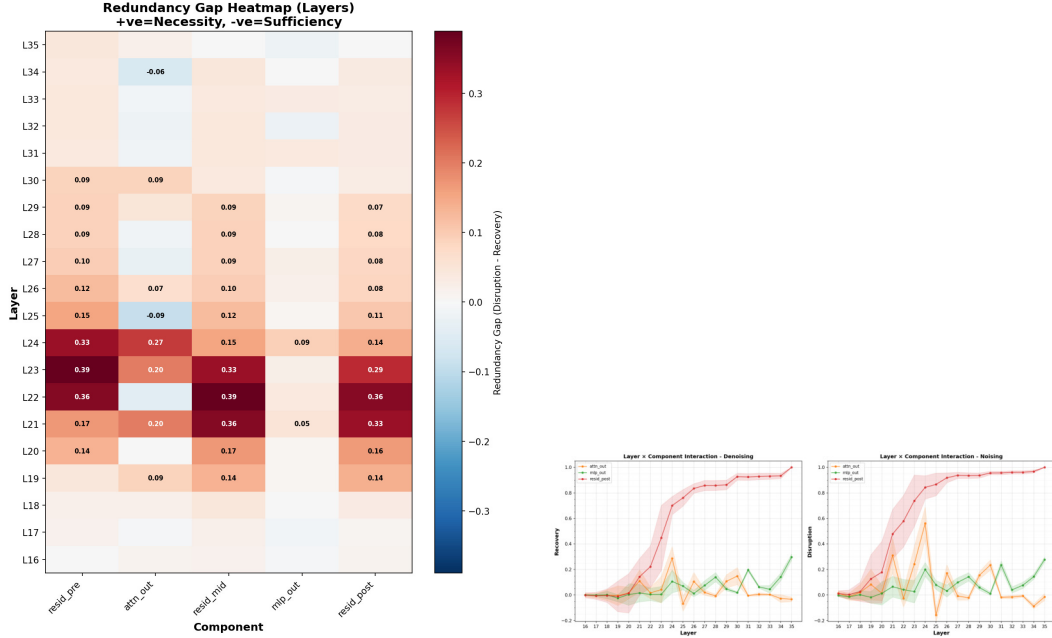
The sufficiency peak at layers 21–24 indicates that the information added to the residual stream by these layers is disproportionately important for temporal preference. The necessity curve shows a more gradual rise beginning around layer 19, suggesting that while individual layers beyond the peak contribute less, their cumulative disruption is meaningful. The elevated variance in the peak region indicates that different contrastive pairs engage these layers to different degrees, consistent with the parametric variation in the experimental design.

Layer 19 as onset. Layer 19 deserves attention. In the single-pair case study (Appendix AB), denoising recovery jumps from ~ 0.05 at L18 to ~ 0.5 at L19, the first layer where patching produces a measurable effect on the output. Before L19, the residual stream does not yet encode temporal preference in a form that patching can recover. This onset coincides with the beginning of the steering sweet spot (layers 19–22; Appendix R): the model can be steered at L19 precisely because the temporal computation is just beginning and the representation is still malleable. By L26 (the probing peak; Appendix G), the computation is complete and the representation is readable but no longer easy to redirect. The L19 onset, L21–24 peak, and L26 readout form a coherent computational timeline within the subgraph.

I.3 Redundancy gap heatmap and layer–component interaction

Figure I.3a maps the redundancy gap, defined as noising disruption minus denoising recovery, across all layers and component types. Positive values (red) indicate components that are more necessary than sufficient, while negative values (blue) indicate components that are more sufficient than necessary. Figure I.3b decomposes the layer sweep into separate traces for attention, MLP, and residual stream components, revealing how each component type’s causal effect varies across layers.

The heatmap reveals a striking pattern: layers 20–23 show large positive redundancy gaps across nearly all component types, with values exceeding 0.6 for the residual stream components. This indicates that these layers are deeply embedded in the temporal preference circuit: corrupting them causes severe disruption, but patching in clean activations at only one component is insufficient for full recovery, because the corrupted signal has already propagated through earlier residual connections. The `attn_out` component at L25 shows a



(a) Redundancy gap heatmap (disruption – recovery) by layer and component type. Strong positive values (dark red) in layers 20–23 across `resid_pre`, `resid_mid`, and `resid_post` indicate high necessity with low sufficiency, characteristic of components embedded in a redundant processing pipeline where no single intervention can fully restore behavior. The `attn_out` column shows a localized peak at L24 (0.39), while `mlp_out` values remain relatively low throughout, suggesting MLP contributions are less redundantly encoded.

(b) Layer-by-layer causal effect for `attn_out`, `mlp_out`, and `resid_post` under denoising (left) and noising (right). The `resid_post` curve rises sharply at layer 20 under denoising and saturates near 1.0 by layer 24, reflecting the cumulative nature of residual stream patching. The `attn_out` and `mlp_out` traces show complementary peaks: attention peaks at layers 21–24, while MLP contributions are more distributed across layers 22–35.

Figure I.3: Redundancy gap heatmap and layer–component interaction analysis.

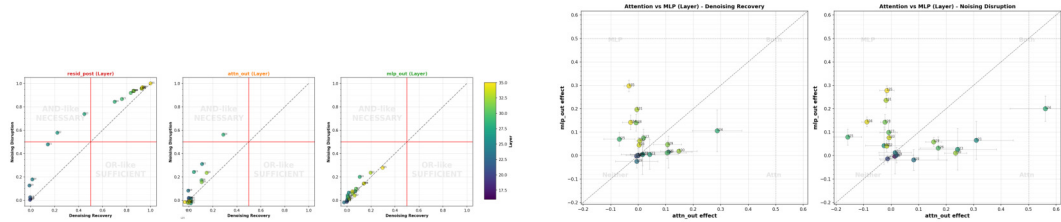
mildly negative gap (-0.11), making it one of the few components where recovery exceeds disruption, suggesting a degree of self-contained sufficiency at that layer.

Several patterns emerge from this decomposition. Under denoising, the residual stream curve exhibits a characteristic sigmoid shape, rising steeply between layers 19 and 24 and then plateauing near full recovery. This reflects the cumulative nature of the residual stream: once the critical mid-layer representations are restored, later layers can process them correctly. The `attn_out` component shows a pronounced peak at layers 21–24 under both denoising and noising, consistent with the component ranking in Figure I.1. MLP contributions, by contrast, are more distributed: under noising, `mlp_out` shows elevated disruption across a broad range of late layers (25–35), suggesting that MLP components contribute through distributed, incremental processing rather than a single localized intervention.

I.4 Noise vs. denoise and attention vs. MLP scatterplots

Figure I.4a plots each layer’s denoising recovery against its noising disruption, separately for each component type. This reveals whether components are sufficient (high recovery, low disruption), necessary (low recovery, high disruption), or both. Figure I.4b directly compares attention and MLP contributions at each layer, revealing the relative dominance of each component type.

The scatterplots confirm the redundancy gap analysis from a different angle. For `resid_post`, layers 20–24 cluster tightly in the upper-left “necessary” quadrant, with high disruption



(a) Denoising recovery vs. noising disruption for each layer, separated by component type (`resid_post`, `attn_out`, `mlp_out`). Points are colored by layer number. The quadrant labels indicate the interpretive regime: “AND-like / necessary” (upper left) for high disruption with low recovery, and “OR-like / sufficient” (lower right) for high recovery with low disruption. Most `resid_post` layers cluster in the necessary quadrant, while `attn_out` and `mlp_out` show more varied profiles.

(b) Attention vs. MLP effect size at each layer under denoising recovery (left) and noising disruption (right). Points above the diagonal indicate MLP-dominant layers; points below indicate attention-dominant layers. Under both metrics, mid-layer points (L21–L24) fall well below the diagonal, confirming that attention drives the largest single-component effects for temporal preference. Late layers (L31–L35) cluster near or above the diagonal under noising, reflecting the distributed MLP contributions in this range.

Figure I.4: Noise vs. denoise scatterplots and attention vs. MLP comparison.

but modest recovery, consistent with a redundantly encoded signal that cannot be fully restored by a single-layer intervention. For `attn_out`, the highest-layer points (around L24) fall near the diagonal, indicating roughly balanced sufficiency and necessity. The `mlp_out` panel shows most layers near the origin, with a few late layers (L31, L35) reaching moderate effect sizes in both directions, consistent with their role as the top-ranked MLP components.

Figure I.5 provides a paired view directly comparing attention and MLP contributions at each layer.

The attention-vs-MLP comparison reveals a consistent pattern: in the critical mid-layer range (L21–L24), attention components have substantially larger causal effects than their MLP counterparts. This asymmetry is especially pronounced under noising disruption, where `L24_attn` and `L21_attn` achieve effect sizes of 0.5–0.6 while the corresponding MLP components remain below 0.2. The paired plot (Figure I.5) makes this particularly clear: the arrows for L21 and L24 sweep dramatically to the right as we move from denoising to noising, indicating that these attention components become even more dominant when measuring necessity rather than sufficiency. In contrast, some layers (L22 in the mid-range, L34 in the upper layers) show arrows pointing upward, indicating that their MLP components are more causally important than their attention components, particularly under noising.

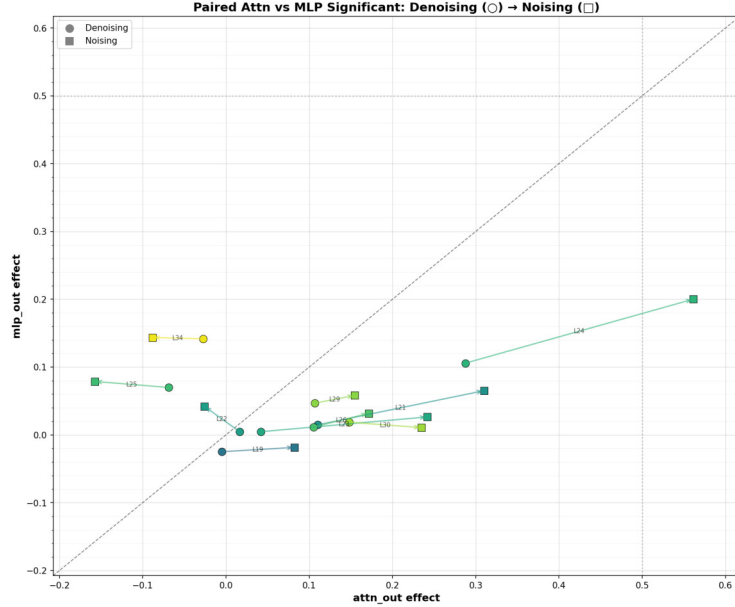


Figure I.5: Paired attention vs. MLP comparison with arrows connecting each layer’s denoising (circle) and noising (square) scores. Layers where the arrow points rightward and downward (e.g., L21, L24) indicate components where noising reveals much stronger attention dominance than denoising. The trajectories of L24 and L21 show the largest rightward displacement, confirming these attention heads as the most causally important individual components. Layers L22 and L34 shift upward, reflecting MLP-dominant noising effects at those layers.

I.5 Summary

The activation patching results converge on several findings that support the claims in the main text:

1. **Sparse, localized circuit.** A small number of components account for the majority of causal effect on temporal preference. The top four components (L24_attn, L21_attn, L35_mlp, and L31_mlp) are clearly separated from the remaining components in effect size (Figure I.1).
2. **Attention dominance in mid-layers.** Attention heads at layers 21 and 24 are the single most causally important components, particularly under noising disruption. Their effect sizes exceed those of any MLP component by a factor of 2–3x (Figures I.4b, I.5).
3. **Distributed MLP contributions in late layers.** MLP components contribute through a more distributed pattern across layers 25–35, with L35_mlp and L31_mlp as the most prominent individual contributors (Figure I.3b).
4. **Necessity exceeds sufficiency.** The high-effect components show a consistent asymmetry: noising disruption exceeds denoising recovery, indicating redundant encoding where no single component is individually sufficient to fully determine temporal preference, but individual components are necessary in the sense that corrupting them substantially degrades performance (Figures I.3a, I.4a).
5. **Critical computation window at layers 20–24.** The marginal contribution analysis localizes the most informative residual stream transformations to a narrow five-layer window (Figure I.2), consistent with a concentrated computational phase for temporal preference.

Appendix J Causal contrastive results

The causal parametric results (Appendix I) intervene on the highly-formatted parametric prompts; here we intervene on the minimally-framed contrastive pairs for the temporal classification task, completing the other diagonal of the method \times paradigm matrix.

This is another attempt at localizing the temporal reasoning of `Qwen3-4B-Instruct-2507` with the activation patching approach. The difference from the previous ones (Appendix H and Appendix I) is that it uses a dataset with binary questions of whether the time horizon is long or short, thus forcing the model to perform temporal classification to answer.

The task design follows the IOI style: clean and corrupted prompts represent phrase beginnings awaiting completion with the tokens *"short"* and *"long"*. Each sentence contains a description of a goal and a question about the time horizon of that goal.

Prompt Template

"The goal is to <goal>. Is this a <short-term or long-term / long-term or short-term> goal? The answer is:"

Sample goals

- **Clean:** "cook a warm dinner for the family"
- **Corrupted:** "become a top chef in the city"

All prompts are appended with a chat template before being passed to the model.

The dataset contains 160 samples with perfectly balanced question order: 80 SL (*"is this a short-term or long-term"*) and 80 LS (*"is this a long-term or short-term"*) pairs to mitigate priming effect. There are three different temporal cue types used for prediction of the long-term horizon: 1) *career/mastery*, achieving elite status or deep expertise at something; 2) *growth*, transforming something small into something large or established; and 3) *accumulation*, exhaustive scope requiring years of sustained effort. All questions relate to a general life domain and are distributed fairly evenly across 25 life subdomains (gardening, cooking, swimming, languages, board games, etc.). The dataset is governed by four design principles. First, token alignment: all 320 prompts must be of the same token length under `Qwen3-4B-Instruct-2507`, ensuring positional correspondence across pairs. In total, each prompt contains 34 tokens, including the chat template, with 7 tokens covering the goal statement. Second, semantic overlap within pairs: each clean and corrupted goal shares the same domain and should be on the same life continuum, differing only in temporal horizons. Third, no explicit temporal keywords in the cues: words such as daily, weekly, or years are banned; the model must infer the horizon from world knowledge alone. Fourth, unambiguous horizons: every short-term goal can be completed in hours or a single sitting, while every long-term goal requires years of sustained effort.

Originally there were 200 samples in the dataset. `Qwen3-4B-Instruct-2507` successfully classified 160/200 pairs, eliciting 80% accuracy. Of the 40 misclassified pairs, 25 involve genuinely ambiguous temporal horizons, where the model's interpretation is defensible. The remaining 15 failures that `Qwen3-4B-Instruct-2507` labels as short-term despite clear temporal signals are concentrated in accumulation-type scholarly activities (*"catalog moon lore from old hill folk"*) and growth-type production at scale (*"fire glazed plates for the whole county"*).

Table J.1 shows dataset statistics for 160 surviving pairs. Table J.2 shows the same statistics but grouped by question order. This dataset is the main asset for activation patching to localize temporal representation.

J.1 160-Pair Directional Patching

We perform denoising activation patching on the `resid_pre`, `attn_out`, and `mlp_out` hooks at all token positions across all 36 layers of `Qwen3-4B-Instruct-2507` on 160 prompt pairs. We patch separately for short \rightarrow long and long \rightarrow short flips, testing whether the computation

Variable	Career/Mastery	Growth	Accumulation
Count	74 (46%)	53 (33%)	33 (20%)
Clean Q, LD (mean +/- st.d.)	14.13 +/- 4.19	11.08 +/- 4.71	12.07 +/- 3.86
Corrupted Q, LD (mean +/- st.d.)	-13.53 +/- 4.88	-10.60 +/- 5.72	-10.26 +/- 4.99

Table J.1: Cue types statistics on successful Temporal Classification pairs

Variable	SL	LS
Count	80	80
Clean Q, LD (mean +/- st.d.)	11.76 +/- 4.47	13.63 +/- 4.35
Corrupted Q, LD (mean +/- st.d.)	-13.61 +/- 5.26	-10.16 +/- 4.97

Table J.2: Question order statistics on successful Temporal Classification pairs

components match. We use three metrics to quantify the effect: logit difference (LD) and log-probability of clean and corrupted answers: $\log-P(\text{clean})$ and $\log-P(\text{corr})$, respectively. All of them are normalized so that 0 corresponds to the corrupted baseline and ± 1 to full recovery of the clean run’s behavior. By definition of presented metrics the two denoising rounds for different flips yield the same result as applying both the noising and denoising techniques on either one of flips. As we cannot assume symmetry between *short* and *long* representations, we treat the flips separately and consider the noising or disruption of one flip as a recovery of the other, rather than as a reflection of the necessity and sufficiency of a single temporal classification circuit.

We refer to the flip in which the clean answer is *short* and the corrupted answer is *long* as the *short-clean* flip, and the opposite as the *long-clean* flip. We first present heatmaps for all tokens, highlighting that the end token accumulates the maximum patching effect. We then provide per-layer plots at the end token position with 95% confidence intervals.

J.1.1 Residual Stream

Figure J.1 shows residual stream patching results for all 34 tokens of the prompt for two flip directions. We can see that both patterns are quite similar in the global structure. They show the same three activity bands:

- early layers (L0–19) highlighting the goal statement tokens at positions 7–14: *the model reading the cue*;
- middle layers (L12–26) highlighting the temporal keywords (positions 18–22: “*short/-term/or/long/-term*”): *the question machinery*;
- late layers (L20–35) concentrated on the end token (position 33): *the decision*.

In both heatmaps the end column saturates the blue scale from \sim L20 downward and represents the location of maximum patching effect. The qualitative circuit (“goal-read \rightarrow question-read \rightarrow decide at end”) is the same whether the model is being steered toward “*short*” or “*long*”.

Three-region structure, core decision window at L20-27. Figure J.2 shows residual stream patching results at final token positions for both directions. We can observe the same three regions in the dynamics of cumulative denoising curves for both flips. Layers 0–19 are causally silent on logit difference: the confidence intervals include zero at every layer. Logarithmic probabilities show very small positive biases in some early layers that do not translate into a detectable LD effect. Layers 20–27 form a *core decision window* during which the LD recovery rises from below 10% to roughly 80% of the clean baseline (84.3% short-clean, 79.7% long-clean at L27); this seven-layer staircase accounts for the vast majority of the patching effect. Layers 28–35 contribute a slow saturation tail, with small



Figure J.1: Directional residual stream patching averaged over 160 classification pairs. Top row: denoising for "short" clean and "long" corrupted; bottom row: denoising for "long" clean and "short" corrupted.

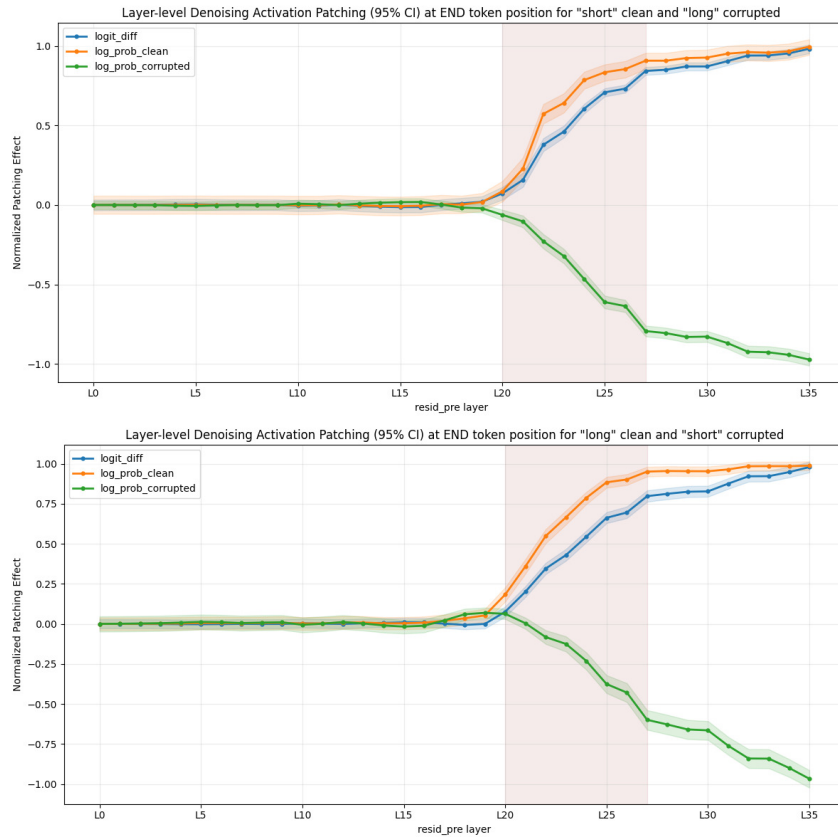


Figure J.2: Patching effects on residual stream at END token positions with highlighted decision window. Top row: denoising for "short" clean and "long" corrupted; bottom row: denoising for "long" clean and "short" corrupted. Core decision window at L20-27 highlighted in red.

Promotion and suppression in Residual Activation Patching at END-token position (160 pairs)

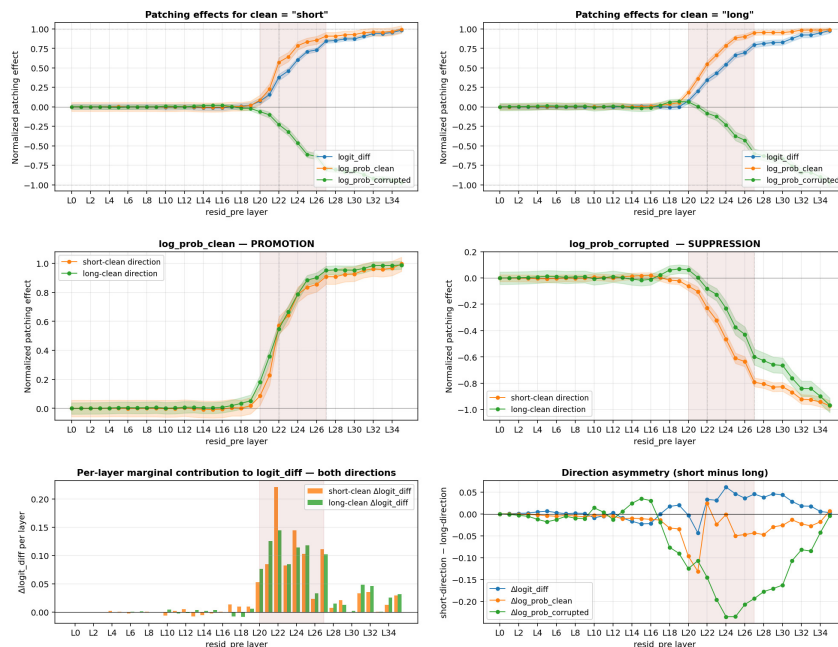


Figure J.3: Core decision window, almost symmetric promotion, markedly asymmetric suppression.

but consistent additional contributions around L31–L32.

Within the core window, the same five layers are the dominant contributors in both flips. Ranking layers by Δ LD, the short-clean flip is led by layers 22, 24, 27, 25, 21 and the long-clean flip by layers 22, 21, 25, 24, 27. The sum of these five per-layer contributions accounts for 68% of the full recovery in the short-clean flip and 62% in the long-clean flip, while cumulative recovery at the end of the seven-layer window (L27) reaches 84% and 80% respectively. Layer 22 alone is the largest single contributor in both flips, with Δ LD = +0.221 in the short-clean flip and +0.144 in the long-clean flip.

Nearly symmetric promotion, asymmetric suppression. The two flips recover the clean answer at almost the same speed: the logarithmic probability curves of clean answers overlap within their bounds across all 36 layers, with direction-wise differences of at most 0.15 anywhere in the domain (Fig. J.3, middle left and bottom right). Suppression of the corrupted answer, however, is markedly faster in the short-clean flip than in the long-clean flip. At layer 24, the normalized suppression is -0.466 in the short-clean flip versus -0.231 in the long-clean flip: a two-fold gap. The absolute direction difference $|\text{short} - \text{long}|$ on the logarithmic probability of corrupted answer peaks at 0.23 across layers 24–25 and decays monotonically, reaching 0.08 by layer 32 and closing to within 0.01 at layer 35. This peak gap is substantially larger than the peak direction differences on LD (0.06 at L24) and on logarithmic probability of clean answer (0.13 at L21) (Fig. J.3, bottom right). The model produces the two answers with nearly identical efficiency but requires additional late-layer computation to push *short* down when the correct answer is *long*. This mechanical asymmetry is consistent with the behavioral bias observed during dataset construction, in which all 15 clear-signal misclassifications were false-*short* predictions.

Milestone layers. Table J.3 summarizes the first layer at which the mean patching effect crosses a given magnitude threshold, for each metric and each flip. Onsets ($|\text{effect}| \geq 0.05$) occur within a narrow two-layer window, at L20 on LD in both flips, at L20 (short-clean)

and L19 (long-clean) on clean logarithmic probability, and at L20 (short-clean) and L18 (long-clean) on corrupted logarithmic probability. The long-clean flip reaches onset 1–2 layers earlier than the short-clean flip on both log-probability metrics, but falls progressively behind at higher thresholds. Above the onset, LD and clean logarithmic probability milestones coincide across flips to within one layer at every threshold, whereas corrupted logarithmic probability milestones diverge: reaching 50%, 75%, and 90% of the full suppression requires layers 25/27/32 in the short-clean flip but 27/31/35 in the long-clean flip, a delay of 2–4 layers at each threshold.

Threshold effect	LD		log- P (clean)		log- P (corr)	
	short	long	short	long	short	long
0.05	20	20	20	19	20	18
0.10	21	21	21	20	21	23
0.25	22	22	22	21	23	25
0.50	24	24	22	22	25	27
0.75	27	27	24	24	27	31
0.90	31	32	27	26	32	35

Table J.3: First `resid_pre` layer at which the mean patching effect crosses each magnitude threshold, at the end token position, for the short-clean and long-clean flips. **Bold entries** highlight the 50%, 75%, and 90% log- P (corr) milestones discussed in the text, where the short-clean flip reaches each threshold 2–4 layers before the long-clean flip.

Summary. Temporal classification is implemented at the final token position by `resid_pre` layers 20–27, which together account for 83.8% (short-clean) and 81.5% (long-clean) of the full recovery, with layer 22 the single largest contributor in both flips and layers 21, 24, 25, and 27 together accounting for over half of the remaining recovery. Layers 0–19 have no causal effect at this position, and layers 28–35 contribute a smaller late-stage refinement (\sim 14–19% of the total recovery). The computation is nearly symmetric in clean-answer promotion (peak direction difference 0.13 on log- P (clean)) and markedly asymmetric in corrupted-answer suppression (peak direction difference 0.23 on log- P (corr)): suppressing *long* when the answer is *short* reaches $|\text{effect}| \geq 0.90$ by layer 32, whereas suppressing *short* when the answer is *long* does not reach the same threshold until the final layer (L35). This residual-stream asymmetry parallels the short-biased behavioral errors observed during dataset construction (all 15 clear-signal misclassifications were false-*short* predictions), though establishing a causal link between the two would require head-level or lens-based analysis.

J.2 Attention-output patching at END token

To localize the residual-stream effect to a specific component, we apply denoising activation patching to the per-layer attention-output hook (`attn_out`) first at all token positions and then at the final one. Each patch replaces a single (layer, pos) summed attention output with its clean-run counterpart, isolating the contribution of that layer’s attention at given position independent of MLPs and residual pass-through. We present the results for all tokens on Figure J.4 and for the final token on Figure J.5. Since we are primarily interested in interpreting the behavior of attention output flow in the END token, all the results described will concern only it.

Attention writes the decision at a sparse set of layers with L24 being the most dominant. Whereas the cumulative `resid_pre` curve rises as a smooth seven-layer staircase across L20–27 (Sec. J.1.1), per-layer attention effects are sharply localized to four dominant writer layers: L21, L24, L26, and L30. Each of these produces a significant positive LD effect in both flips (confidence intervals bounded away from zero), and the three strongest writers alone (L24, L26, L30) each contribute roughly a third of the full normalized recovery individually. A pair of smaller late writers at L33–34 rounds out the positive contribution, while the first fifteen layers produce no significant attention effect on any metric.



Figure J.4: Directional attention output patching averaged over 160 classification pairs. Top row: denoising for "short" clean and "long" corrupted; bottom row: denoising for "long" clean and "short" corrupted.

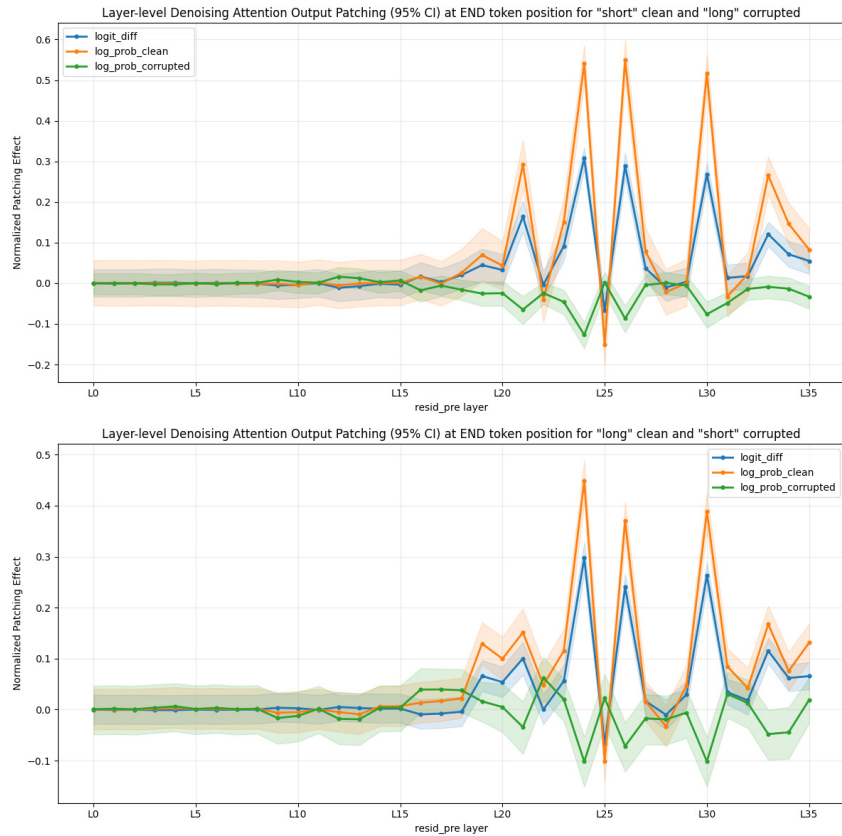


Figure J.5: Patching effects on attention output at END token positions. Top row: denoising for "short" clean and "long" corrupted; bottom row: denoising for "long" clean and "short" corrupted.

Layer 25 is a negative outlier. Between the L24 and L26 peaks sits an isolated layer whose attention output, when patched from the clean run, *worsens* the clean-answer probability in both flips. The effect is small in magnitude but statistically clear (CI upper bounds strictly below zero), and it appears specifically on $\log\text{-}P(\text{clean})$, with $\log\text{-}P(\text{corr})$ remaining near zero. Several interpretations are compatible with this pattern (cross-run interference, a competitive interaction with the flanking L24/L26 writers, or head-level cancellation within the summed output), but discriminating among them requires head-level patching. We note it here because the negative sign is reproducible across both flips and deserves follow-up rather than dismissal.

Attention promotes the correct answer but barely suppresses the incorrect one. At every dominant attention writer, the effect on $\log\text{-}P(\text{clean})$ is several times larger than the effect on $\log\text{-}P(\text{corr})$: the per-layer promotion-to-suppression ratio exceeds four everywhere among L21, L24, L26, L30 and reaches six or more at the deeper writers in the short-clean flip. This is a component-level observation that the residual-stream totals alone cannot reveal, because at `resid_pre` both metrics eventually approach unit magnitude. The implication is that attention at the decision layers implements primarily *answer-promotion*: it writes “the correct answer is here” into the residual stream, with only a modest side effect on the competing answer. The deep suppression observed at `resid_pre` (where $\log\text{-}P(\text{corr})$ saturates near -1 by the final layers) must therefore come largely from a different component, most plausibly the MLPs, although this wasn’t confirmed by our MLP patching analysis (Sec. J.3).

The direction asymmetry largely disappears at the attention level. A central finding of the `resid_pre` analysis was that corrupted-answer suppression is faster when the correct answer is *short* than when it is *long*, with a peak direction gap of 0.23 on $\log\text{-}P(\text{corr})$. At `attn_out`, this pattern is absent: attention effects on the corrupted answer are nearly equal across flips at every dominant writer. The largest remaining direction difference shifts to *promotion* rather than suppression: the short-clean flip writes a noticeably stronger clean signal at L26 than the long-clean flip does, but even this residual asymmetry is well under half the size of the one observed at `resid_pre`. Taken together, these two facts indicate that the direction-asymmetric late-layer suppression seen in the residual stream does not originate in the attention blocks.

Summary. The `resid_pre` decision window resolves, at the attention-output level, into four primary writer layers (L21, L24, L26, L30), a smaller late write at L33–34, and a single reproducible negative contribution at L25. These attention blocks are strongly biased toward promotion of the correct answer rather than suppression of the incorrect one, and they behave nearly symmetrically across the two answer directions. The direction-asymmetric late-layer suppression observed at `resid_pre` is therefore attributable to a different component, which matching MLP-output patching should be able to identify.

J.3 MLP-output patching at END token

The attention-output analysis (Sec. J.2) suggested that attention blocks promote the correct answer with only a small suppression side-effect, and we conjectured that the deeper, direction-asymmetric suppression seen at `resid_pre` would be carried by the MLPs. To test this, we patch the per-layer `mlp_out` hook at the final token position, using the same two flips and the same three metrics. We also provide per-layer effects for all prompt tokens in Fig J.6 for a broader view. Fig. J.7 shows patching effects at the final token and Figure J.8 aggregates structural findings by multiple plots.

MLPs are weaker writers than attention, and they also promote. MLP effects are substantially smaller in magnitude than attention effects: peak LD is $+0.131$ (short-clean, L27) versus $+0.308$ for attention (short-clean, L24), about a factor of 2.4 smaller. A consistent positive-LD signature appears across three layers in the middle-late range (L27, L28, and L31), which form the primary MLP writer band and are significant in both flips (L27: $+0.131/+0.113$; L28: $+0.123/+0.083$; L31: $+0.093/+0.089$). A secondary supporting band at L22–L25 contributes smaller but reproducibly significant LD effects in

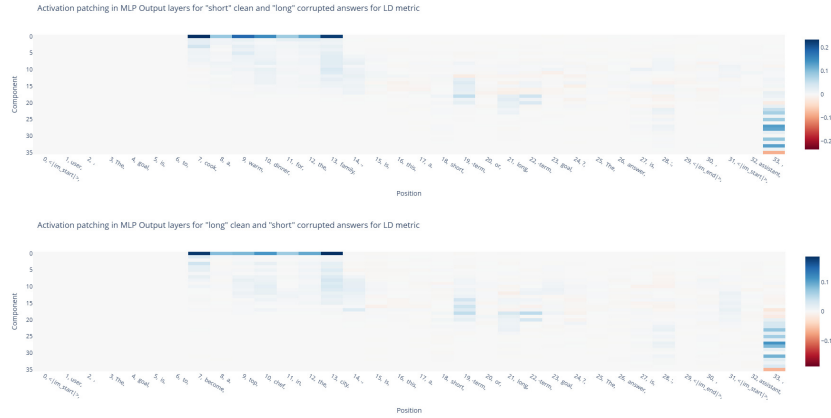


Figure J.6: Directional MLP output patching averaged over 160 classification pairs. Top row: denoising for "short" clean and "long" corrupted; bottom row: denoising for "long" clean and "short" corrupted.

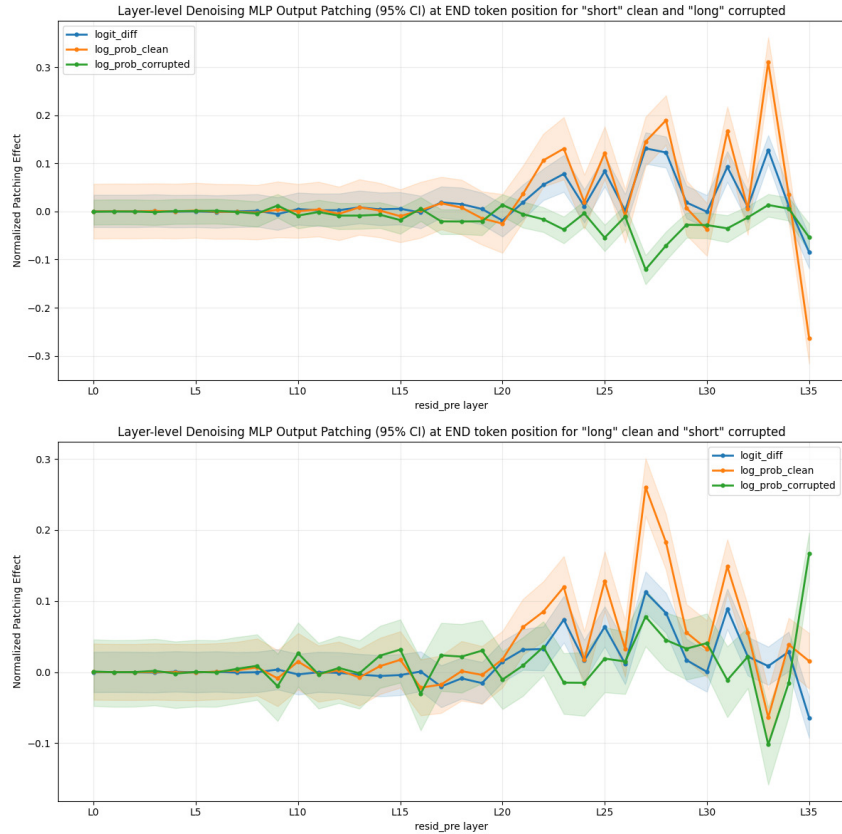


Figure J.7: Patching effects on MLP output at END token positions. Top row: denoising for "short" clean and "long" corrupted; bottom row: denoising for "long" clean and "short" corrupted.

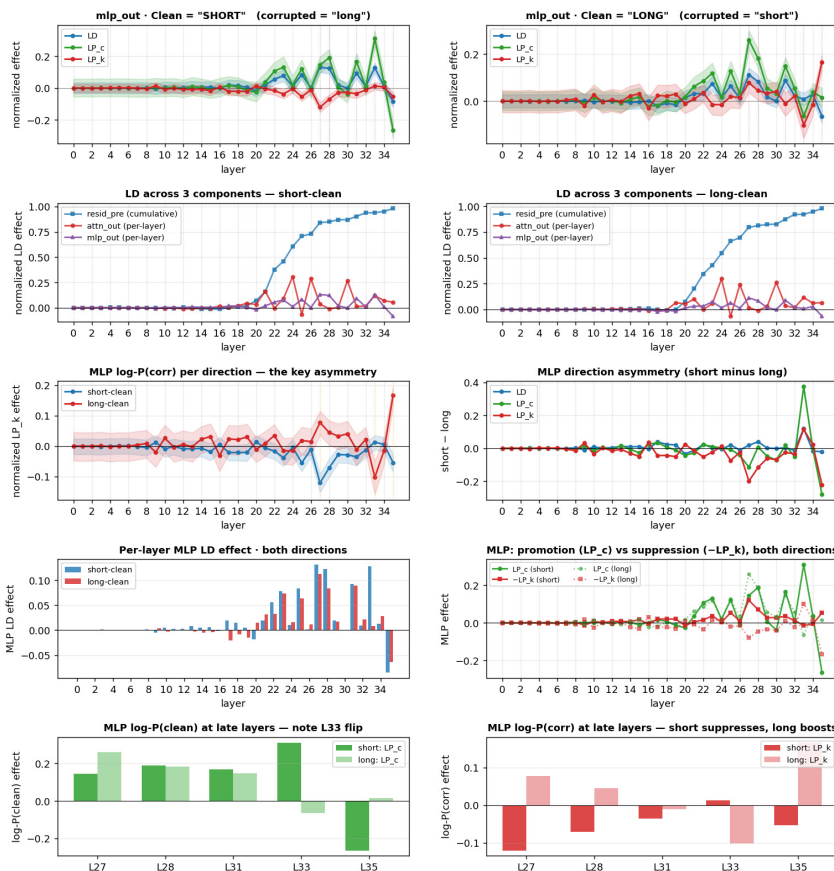


Figure J.8: MLP output patching results from different angles.

both directions. *Contrary to our initial conjecture, these MLP writers do not primarily suppress the incorrect answer:* at every layer in the primary band, the effect on $\log-P(\text{clean})$ is larger in magnitude than the effect on $\log-P(\text{corr})$, the same promotion-dominant pattern we observed at attention. MLPs therefore reinforce the decision written by attention rather than performing a qualitatively distinct suppression step.

The late-layer suppression hypothesis is not supported. The residual-stream analysis showed that $\log-P(\text{corr})$ saturates near -1 across layers 28–35, with a pronounced direction-dependent delay in the long-clean flip. If a specific MLP layer were responsible for this late-layer suppression, it should appear here as a large negative effect on $\log-P(\text{corr})$ at one or more of layers 28–35. The observed MLP $\log-P(\text{corr})$ effects in that range are, however, small and mixed in sign: the largest is -0.121 at L27 in the short-clean flip (significant), but in the long-clean flip the corresponding MLPs at L27 and L28 show *positive* $\log-P(\text{corr})$ effects ($+0.078$ and $+0.045$, both significant), meaning they slightly push the model toward the incorrect answer. No single MLP layer produces a suppression effect comparable in magnitude to the saturation observed at `resid_pre`. The component-level decomposition we conjectured at the end of Sec. J.2 (attention promoting, MLPs suppressing) is therefore not supported by the data. The deep suppression at `resid_pre` appears instead to be a cumulative property of many small contributions distributed across both components, rather than the work of a localized suppressor.

Two MLP writers show sign-flipping asymmetry. Direction-dependent MLP behavior is not confined to a single layer: six of the ten significant MLP writers show a $|\text{short}-\text{long}|$ gap of at least 0.05 on some metric, including L27, L28, L25, and L22 in addition to the two we highlight below. What distinguishes L33 and L35 is the *kind* of asymmetry. At most MLP writers the two flips produce effects of the same sign on $\log-P(\text{clean})$, differing only in magnitude (L27: +0.145 vs +0.260); the layer is doing comparable work in both directions. L33 and L35 are the only writers whose effects flip sign on the $\log-P(\text{clean})$ metric, which most directly reflects the writing of the correct answer; L27 and L28 also show significant sign flips, but only on $\log-P(\text{corrupted})$.

L33: a *short*-reinforcing MLP. In the short-clean flip L33 is the second-largest MLP writer ($\text{LD}=+0.128$, $\log-P(\text{clean})=+0.311$, $\log-P(\text{corr})$ near zero). In the long-clean flip the LD effect vanishes (+0.009, not significant) and both log-probability metrics move in the *opposite* direction to the short-clean case: $\log-P(\text{clean})$ becomes -0.064 and $\log-P(\text{corr})$ becomes -0.102 (both significant). Patching L33’s MLP output from the clean run therefore *boosts short* when *short* is correct but *suppresses short* when *long* is correct. The simplest rule consistent with the four significant effects is that L33 strengthens the model’s representation of *short*, which promotes the clean answer in one flip and the incorrect answer in the other. L33 produces the largest single direction asymmetry we observe at any component on the $\log-P(\text{clean})$ metric: peak $|\text{short}-\text{long}|$ of 0.37, more than double the attention-level peak and nearly $3\times$ the `resid_pre` peak.

L35: a second *short*-reinforcing MLP with complementary polarity. The final MLP, L35, is the other sign-flipping writer. In the short-clean flip it produces a significant negative LD effect (-0.085), driven primarily by a large negative $\log-P(\text{clean})$ effect (-0.264). In the long-clean flip the LD effect is similar in sign and smaller in magnitude (-0.064), but it is now driven by a large *positive* $\log-P(\text{corr})$ effect (+0.167, significant): the single largest $\log-P(\text{corr})$ effect observed at any MLP layer in either direction, and going the “wrong” way for denoising. As at L33, the most natural reading is that L35 implements a computation tied to the *short* reading specifically, one that, when patched, reinforces *short* regardless of whether *short* is the correct or the incorrect answer. Alternative explanations (residual interaction with the unembedding, normalization artifacts at the final layer) cannot be ruled out without further experiments.

Direction asymmetry relocates to late MLPs. Collecting the peak $|\text{short}-\text{long}|$ values across the three components tested so far clarifies where the short/long asymmetry in this model lives. The peaks are 0.06 / 0.13 / 0.23 at `resid_pre` on LD, $\log-P(\text{clean})$, $\log-P(\text{corr})$ respectively, 0.06 / 0.18 / 0.09 at `attn_out`, and 0.12 / 0.37 / 0.22 at `mlp_out`. MLPs are therefore the component that carries the direction asymmetry most strongly, and the per-metric peaks land at different MLP layers: L33 on LD and $\log-P(\text{clean})$, L35 on $\log-P(\text{corr})$, with L27 a close second on $\log-P(\text{corr})$ (0.20 vs L35’s 0.22). The two sign-flipping layers (L33 and L35) implement a direction-sensitive computation whose presence is obscured at `resid_pre` by averaging with the broader population of writers (most of which show a smaller, same-sign asymmetry) upstream.

Summary. MLP patching refutes our earlier conjecture that the direction-asymmetric late-layer suppression at `resid_pre` is localized to MLPs in layers 28–35. MLPs are weaker writers than attention, and at the layers where they contribute significantly (primarily L27, L28, L31, and a supporting band at L22–L25) they continue the promotion-dominant pattern established by attention. The aggregate late-layer suppression observed at `resid_pre` is therefore best understood as a cumulative property of many small contributions across both components, not the work of a localized suppressor. Several MLP writers show direction asymmetry, with L27, L28, and L25 differing in magnitude between flips; against this graded backdrop, L33 and L35 stand out as the only writers whose effect flips sign across flips on a significant metric. L33 implements a *short*-reinforcing computation that promotes the clean answer in the short-clean flip and suppresses it in the long-clean flip; L35 shows a complementary asymmetry that pushes the model toward *short* even in the long-clean flip. The L33/L35 finding does not localize the residual-stream asymmetry’s origin: the asymmetry on $\log-P(\text{corrupted})$ is already near its peak by L24 (gap 0.23) and reflects the

cumulative result of many smaller per-layer asymmetries at L17–L31. What L33 and L35 add is a qualitatively distinct sign-flipping computation on later layers, with L35 in particular making the largest single per-layer log- P (corrupted) asymmetry contribution (0.22) in the model. Head-level attention patching at L21, L24, L26, L30 and neuron-level analysis of MLPs at L33 and L35 would be the natural next experiments to test and refine this picture.

J.4 Key Findings

1. **The decision is sparsely localized at the final token.** Temporal classification at the final token is implemented by a small set of layers in a narrow band of processing depth. The `resid_pre` logit-difference curve is silent for layers 0–19 (confidence intervals contain zero at every layer; maximum mean magnitude 0.020 in the short-clean flip and 0.009 in the long-clean flip), rises as a seven-layer staircase across L20–27 that accounts for 83.8% (short-clean) and 81.5% (long-clean) of the full normalized logit-difference recovery, and continues through a slower late-layer tail at L28–35. Outside this band, no layer contributes a significant effect on LD at the final token position.
2. **Attention and MLP components are both promotion-dominant.** Component-level patching refutes the simplest decomposition one might expect (*attention writes the answer, MLPs suppress the alternative*). At the dominant attention writers the ratio $|\log-P(\text{clean})|/|\log-P(\text{corrupted})|$ is four or more in seven of the eight layer–flip combinations we test (L21, L24, L26, L30 across both flips), reaching $6.77\times$ at L30 short-clean and $6.33\times$ at L26 short-clean; the one exception is L30 long-clean at $3.83\times$. MLP writers are also promotion-dominant at most layers but with greater variability: ratios at the primary-window writers span from 1.21 (L27 short-clean; roughly balanced promotion and suppression) up to 13.31 (L31 long-clean); the direction-specialized layers L33 and L35 (Finding 5) deviate further and are described separately. The late-layer suppression of the incorrect answer seen at `resid_pre` is therefore not localized to a single component; it accumulates from many small contributions distributed across attention and MLPs between L20 and L35.
3. **Attention is the primary writer, MLPs reinforce.** Within the L20–27 window, attention contributes the majority of the magnitude. The principal attention writers in both flips are L21, L24, L26, and L30, with a single-layer peak of $+0.308$ at L24 short-clean; L33 attention makes a contribution comparable to L21 in the long-clean flip ($+0.115$ vs $+0.100$). MLPs add smaller but reproducible writes at L22, L23, L25, L27 in the core window and at L28, L31 in the secondary band (L28–32); peak MLP LD effect is $+0.131$ at L27 short-clean, roughly $2.4\times$ smaller than the attention peak (which sits at a different layer, L24). The two component families work in concert rather than in specialized roles.
4. **Promotion is approximately symmetric across flips; suppression is not.** At `resid_pre`, the onset of the decision on LD coincides in both flips at L20. Clean-answer promotion proceeds at similar rates in the two flips: log- P (clean) milestones at 10, 25, 50, 75, and 90% recovery coincide to within one layer at every threshold, with a peak direction gap of 0.13 on this metric (at L21). Corrupted-answer suppression, by contrast, is consistently faster when the correct answer is *short*: reaching 50%, 75%, and 90% of the full suppression requires layers 25/27/32 in the short-clean flip but 27/31/35 in the long-clean flip, a delay of 2–4 layers at every threshold, and the peak direction gap on log- P (corrupted) reaches 0.23 around L24–25. The suppression asymmetry is thus $\sim 1.8\times$ the promotion asymmetry on the relevant metrics and closes only at the final layer (L35).
5. **Two MLP layers are direction-specialized toward *short* on the log- P (clean) metric.** Several MLP writers show significant direction-dependent behavior (including L27 and L28, which have significant sign-flipping effects on log- P (corrupted): -0.121 vs $+0.078$ at L27 and -0.071 vs $+0.045$ at L28). Layers 33 and 35 are distinguished as the MLP writers whose effects flip sign on the log- P (clean) metric: the metric most directly tied to writing the correct answer. At L33, patching MLP output from the clean run yields log- P (clean) = $+0.311$ when *short* is correct and log- P (clean) = -0.064 when *long* is correct (both significant); the LD effect therefore vanishes in the long-clean flip ($+0.009$, not significant). At L35 the corresponding effect is -0.264 short-clean vs $+0.016$ long-clean (only the short-clean side is significant on this metric), while on log- P (corrupted) L35

produces +0.167 in the long-clean flip, the single largest $\log\text{-}P(\text{corrupted})$ effect at any MLP layer in either direction. The simplest rule consistent with the nine significant effects at these two layers is that they modify the representation of the token *short* regardless of whether *short* is the correct or the incorrect answer.

6. **The direction asymmetry concentrates in late MLPs, with per-metric peaks at different layers.** Peak $|\text{short} - \text{long}|$ values across the three components are 0.06/0.13/0.23 at `resid_pre` on LD, $\log\text{-}P(\text{clean})$, $\log\text{-}P(\text{corrupted})$ respectively, 0.06/0.18/0.09 at `attn_out`, and 0.12/0.37/0.22 at `mlp_out`. MLPs are therefore the component that carries the direction asymmetry most strongly. Within MLPs, the per-metric peaks land at different layers: L33 on LD (0.12) and on $\log\text{-}P(\text{clean})$ (0.37); L35 on $\log\text{-}P(\text{corrupted})$ (0.22), closely followed by L27 (0.20). Attention effects on the incorrect answer are nearly equal across flips even at the major writers (e.g. L24: -0.127 vs -0.102), so the late-layer `resid_pre` asymmetry does not originate in attention; instead it is carried by late MLPs at L27, L33, and L35 prominently, with smaller contributions from L28.
7. **The circuit-level asymmetry parallels a behavioral short-bias.** The 15 clear-signal misclassifications in the dataset were all false-*short* predictions. The residual-stream finding that suppressing *short* (long-clean flip) requires more layers than suppressing *long*, together with the MLP-level finding that L33 and L35 implement a *short*-reinforcing computation that resists being pushed toward *long*, forms a mechanical pattern in the same direction as this behavioral bias. A causal link between the patching-level asymmetry and the classification-level errors cannot be established from this experiment alone, but the direction of both effects agrees.
8. **Convergence with parametric results.** The attention layers identified here (L21, L24, L26, L30) overlap with the layers identified by the highly-formatted parametric experiments (Appendix I), despite using entirely different prompt structures and datasets. This cross-paradigm convergence strengthens the localization claim.

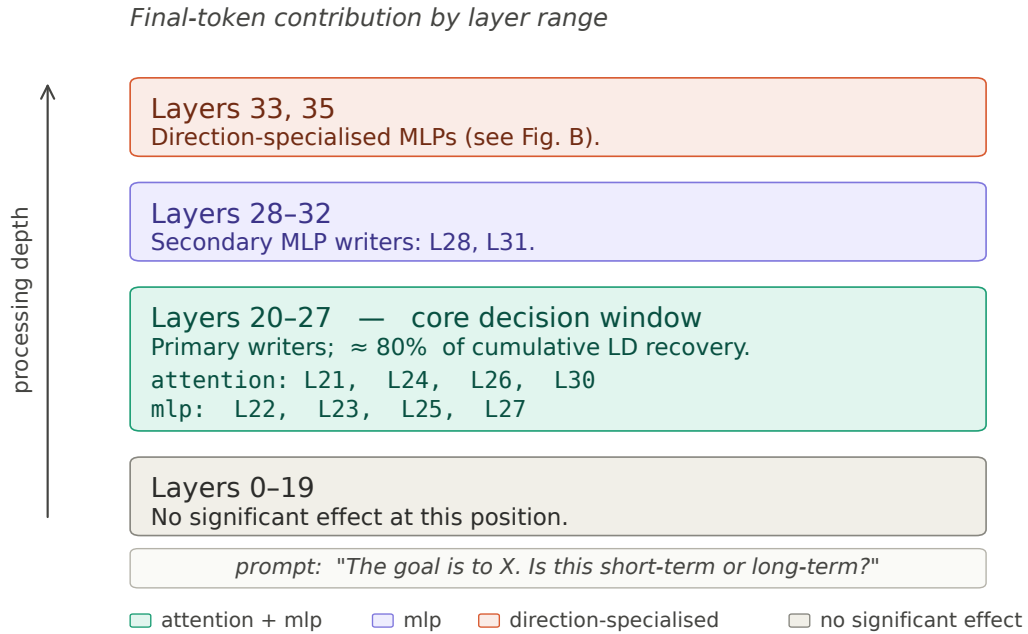


Figure J.9: Coarse circuit for temporal classification at the final token position in `Qwen3-4B-Instruct-2507`. Bands are ordered by processing depth (early layers at the bottom, late layers at the top). Layers are grouped by the component whose per-layer patching produces a mean effect on the logit difference that is significant in both the short-clean and long-clean flips, at the reported confidence bounds. Layers 0–19 have no significant effect on any metric at this position; layers 20–27 form the core decision window, containing the primary attention writers L21, L24, L26, L30 and the MLP writers L22, L23, L25, L27; a smaller secondary MLP band at L28 and L31 continues the writing into layers 28–32; layers 33 and 35 are direction-specialized MLPs whose behavior differs systematically across flips (Fig. J.10). Full per-layer effects with confidence intervals are reported in Table J.3 and Figs. J.3–J.8.

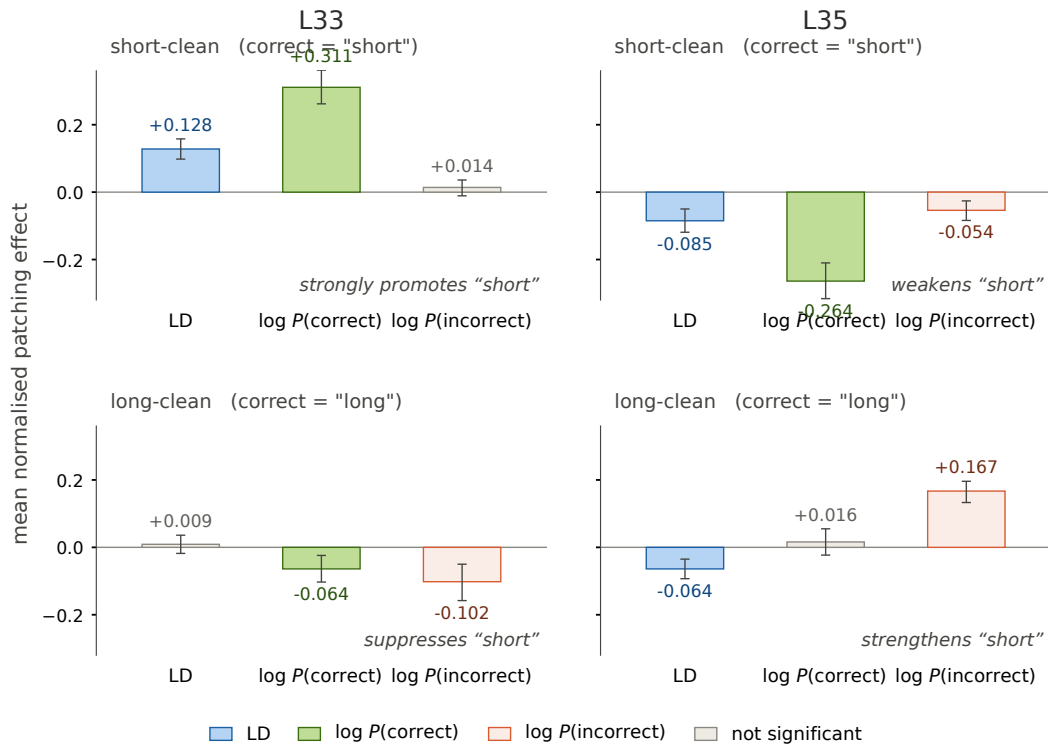


Figure J.10: Direction-specialized behavior of MLP outputs at layers 33 and 35. Each panel reports the mean normalized patching effect on three metrics (logit difference, or LD; log-probability of the correct answer; and log-probability of the incorrect answer) with error bars showing the reported confidence bounds. The top row shows the short-clean flip (correct answer *short*); the bottom row shows the long-clean flip (correct answer *long*). Gray bars mark effects whose confidence interval contains zero; all other bars are significant. At layer 33 the pattern of signs flips between flips: the same patching operation promotes *short* when *short* is correct and suppresses *short* when *long* is correct. Layer 35 shows a complementary asymmetry, with a large significant positive effect on log- $P(\text{incorrect})$ in the long-clean flip. The simplest single rule consistent with all eight significant effects is that patching these two MLPs modifies the model’s representation of the token *short*, independent of whether *short* is the correct or the incorrect answer for the prompt.

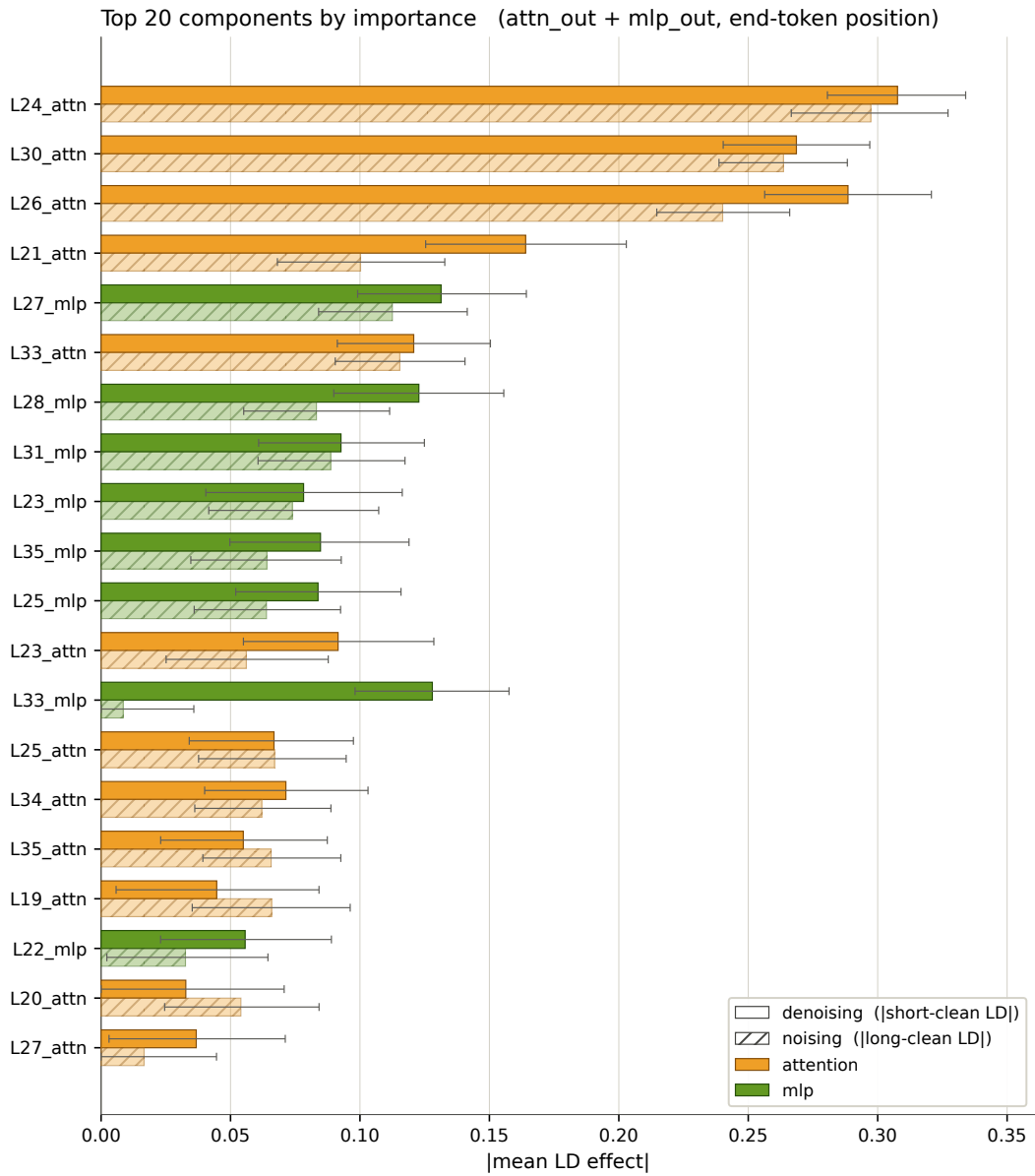


Figure J.11: Top-20 Attention and MLP outputs layers ranked by their patching effect at END token position

Appendix K Cross-method convergence

The preceding five appendices each approached localization from a different angle: two attribution methods, two causal patching experiments, and supervised probing, applied across two prompt paradigms. Each method has different assumptions, blind spots, and failure modes. The question is whether they agree.

Table K.1 and Figure K.1 show that they do: all four methods place the temporal preference subgraph in layers 17–35, with L24 attention at the center. The agreement is not trivial, as the methods also disagree in informative ways (Section K.2).

	Probing Appendix G	Attr. contr. Appendix H	Causal param. Appendix I	Causal contr. Appendix J
Attn L21–L24		✓	✓	✓
Attn L18, L26, L30, L33				✓
MLP L31–L35		✓	✓	
Resid L17–L22 (recovery)			✓	
Resid L20–L27 (core)				✓
Resid L26	✓			
Attn peak		L24 (ST), L22 (LT)	L24, L21	L24
MLP peak		L34, L35	L35, L31	
Best single layer	L26	L24	L24	L24
Signal onset	~L17	~L21	~L19	~L18

Convergence zone: layers 17–35

Table K.1: Layers and components identified by each localization method. All four methods place the temporal preference subgraph in layers 17–35. L24 attention is flagged by every method except probing. MLP contributions concentrate in L31–L35 under attributional contrastive and causal parametric patching. The causal contrastive experiment additionally identifies shared attention layers at L18, L26, L30, and L33.

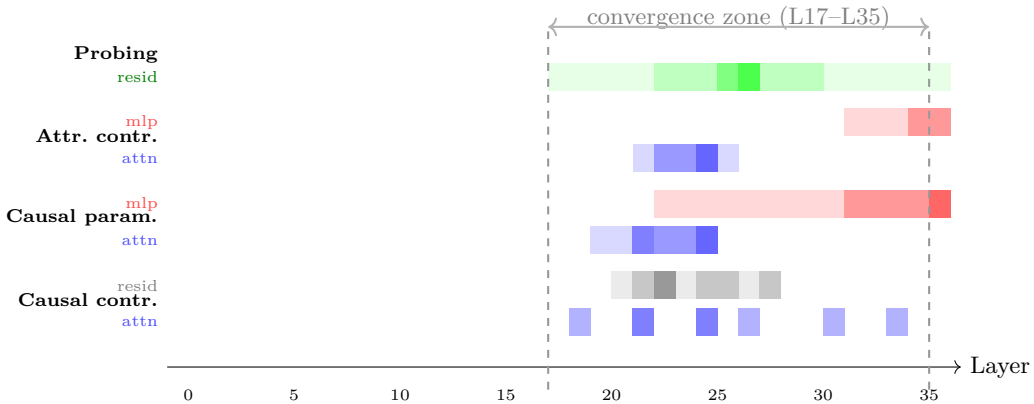


Figure K.1: Layer-level convergence across all four localization methods. Darker shading indicates stronger signal. Blue = attention, red = MLP, green = residual stream (probing), gray = residual stream (causal contrastive core decision window). L24 attention appears in every non-probing method. The causal contrastive experiment (row 4) additionally reveals sparse attention contributions at L18, L26, L30, and L33, together with a residual-stream core decision window at L20–L27 (peak L22).

K.1 Points of Agreement

L24 attention is the single component flagged most consistently: it appears as a top-ranked element in all three non-probing methods across both paradigms, making it the most robustly identified element of the temporal preference subgraph. The MLP contribution concentrates in L31–L35 under attributional contrastive and causal parametric patching, with MLP L31 identified as a key disruptor by both methods. All methods agree that the temporal signal is absent from the first ~ 15 layers and concentrated in the upper half of the network. The probing peak at L26 falls between the attention computation window (L21–L24) and the MLP computation window (L31–L35), consistent with a readout layer that consolidates the output of the attention-mediated temporal routing before the MLP transformation stage. The causal contrastive residual-stream staircase (L20–L27, peak L22) spans this same region, converging with the attention window (L21–L24) on the low end and with the probing peak (L26) on the high end.

K.2 Points of Disagreement

The methods disagree on three dimensions.

Attention breadth. The causal contrastive experiment identifies a broad set of attention layers (L18, L21, L24, L26, L30, L33), while the other two non-probing methods focus narrowly on L21–L24. This likely reflects a difference in prompt structure: the minimally-framed contrastive prompts require more distributed attention routing because the temporal signal is implicit, whereas the highly-formatted prompts concentrate the signal in a structured template.

MLP visibility. The causal contrastive experiment shows almost no MLP signal, while the other two non-probing methods identify L31–L35 as important. This may reflect the smaller contrastive datasets (160 pairs for the residual-stream sweep, 6 pairs for the attention sweep) and shorter prompts used in the contrastive paradigm, which may not engage the MLP transformation layers as strongly as the parametric setup.

Sufficiency vs. necessity asymmetry. The causal parametric results reveal a sharp split between mid-layer recovery (L17–L22) and late-layer disruption (L30–L35), a pattern that the contrastive methods do not clearly replicate. The causal contrastive experiment instead finds a different kind of asymmetry: the two flip directions share the same core decision window (L20–L27) and promote the clean answer at comparable rates, but suppress the corrupted answer at different speeds (suppressing *short* when the answer is *long* takes 2–4 more layers than the reverse). This direction-dependent suppression is a dimension the parametric methods do not probe, because they do not separate patching directions.

Signal onset. The earliest onset varies from $\sim L17$ (probing) to $\sim L21$ (attributional contrastive). This 4-layer gap may reflect the greater sensitivity of probing and parametric prompts to early, low-magnitude temporal information that the contrastive attribution method, aggregated over many prompt variants, averages out.

K.3 Interpretation

The disagreements are interpretable rather than contradictory: they reflect genuine differences in what each method measures. Attribution methods approximate causal effects via gradients and are sensitive to all information flow, including redundant pathways. Causal methods measure the actual behavioral consequence of intervention and are therefore sensitive to necessity and sufficiency. Probing measures the linear readability of a concept at a given layer, regardless of whether that layer is causally important.

That these five methods, despite their different assumptions and blind spots, converge on a common subgraph in layers 17–35 with L24 attention at the center supports the claim that the localization is not an artifact of any single method. The probing–steering dissociation (Appendix R) adds a sixth data point: layers 19–22 are most effective for writing temporal

preference, while layer 26 is most effective for reading it, reinforcing the functional distinction between the attention routing window and the readout layer.

However, the subgraph is not monolithic. The latent vs. constrained analysis (Appendix M) reveals that the same L17–35 region operates in two modes depending on whether the prompt carries an explicit time-horizon constraint:

- **Constrained mode:** the full subgraph is engaged (attention L21–24 *and* MLP L31–35), producing strong, distributed effects.
- **Latent mode:** only attention at L21–22 is active, with minimal MLP involvement.

The MLP layers that feature prominently in the convergence table (L31, L35) may therefore be specifically about processing constraint tokens rather than encoding temporal preference per se. The attention core at L21–24 is the shared substrate; MLP extends the computation when the prompt provides an explicit temporal anchor.

Part 2:

What does temporal preference look like?

- **L.** Parametric geometry
- **M.** Latent vs. constrained
- **N.** Behavioral temporal discounting
- **O.** Behavioral coherence
- **P.** Cross-model patching comparison
- **Q.** Error monitoring in the subgraph

Appendix L Parametric geometry

Part 1 established *where* temporal preference lives (layers 17–35, with L24 attention at the center; Appendix K). Now we ask: *what does the representation look like inside that subgraph?*

We apply PCA to 4,588 activation vectors, sampled from a logarithmic grid over reward amounts, delay times, and 17 time horizons (seconds to centuries), at 15 layers, 5 component types, and 16 semantic positions per prompt (methodology in Appendix X). At key positions, PC1 captures 44–71% of variance (Table X.1). The results tell a mechanistic story in five stages: the model builds an ordinal time-horizon representation, the geometric direction encoding it flips across prompt positions, it stabilizes at the user-to-assistant turn boundary, attention transforms it into a binary preference signal over the next few tokens, and the preference commits by the `assistant` token.

L.1 Progressive separation across layers

Figure L.1 shows how the PC1 projection evolves across layers for four component types, colored by the model’s eventual choice (long vs. short).

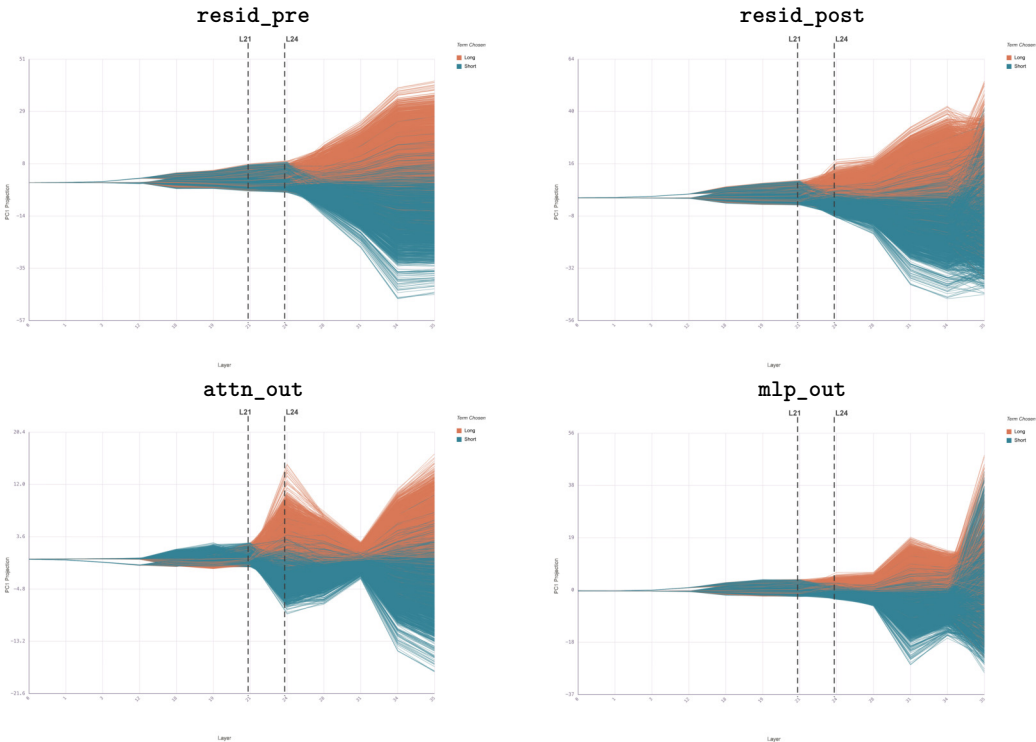


Figure L.1: PC1 projection across layers for each component type, colored by the model’s chosen term (orange = long, blue = short). Short-term and long-term traces are inseparable until approximately layer 24, where the residual stream (± 50) and attention (± 20) begin to diverge. MLP contributions emerge later and remain smaller.

All traces begin bundled near zero and remain inseparable through the first ~ 20 layers. The separation becomes visible around layer 24 in the residual stream and attention output, consistent with the causal importance of L24 identified by activation patching (Appendix I). By the final layers, the residual stream carries a separation of roughly ± 50 on PC1, while attention contributes ± 20 and MLP a smaller but complementary signal concentrated in the upper layers.

L.2 Off-policy horizon constraint (2D PCA)

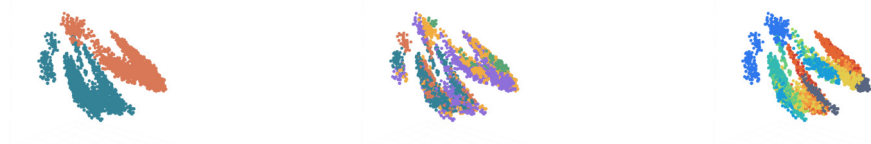


Figure L.2: PCA of activation space at three token positions (chosen term, chosen time, time scale) with the time horizon given as an explicit constraint.

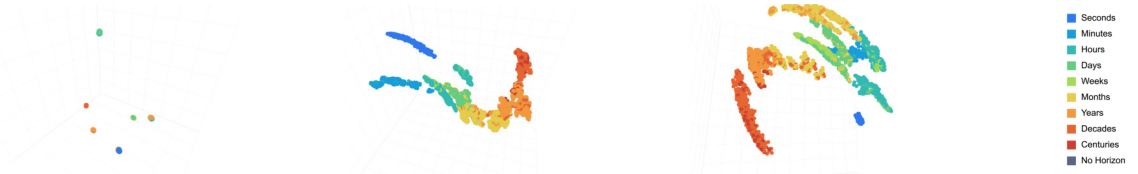


Figure L.3: PCA of activation space colored by time scale at layers 3, 18, and 24 after the time horizon token. Clusters become increasingly separable in the mid-to-upper layers.

L.3 On-policy temporal preference (2D PCA)

The preceding subsection examined how explicit time-horizon constraints are represented in activation space. Here we trace what happens as the model transitions from the user’s turn (where the constraint is given off-policy) into the assistant’s turn, where it must generate on-policy text reflecting a temporal preference. Figure L.4 illustrates the token-level structure of this transition. As the figures below show, the explicit time-horizon clusters reorganize during this hand-off: the no-horizon samples, initially disjoint, align to the time-scale manifold and temporal preference becomes linearly separable even at the earliest layers.

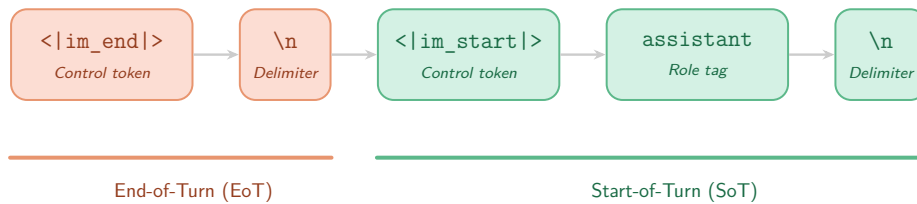


Figure L.4: The transition from user EoT to assistant SoT marks the shift from off-policy context to on-policy generation.

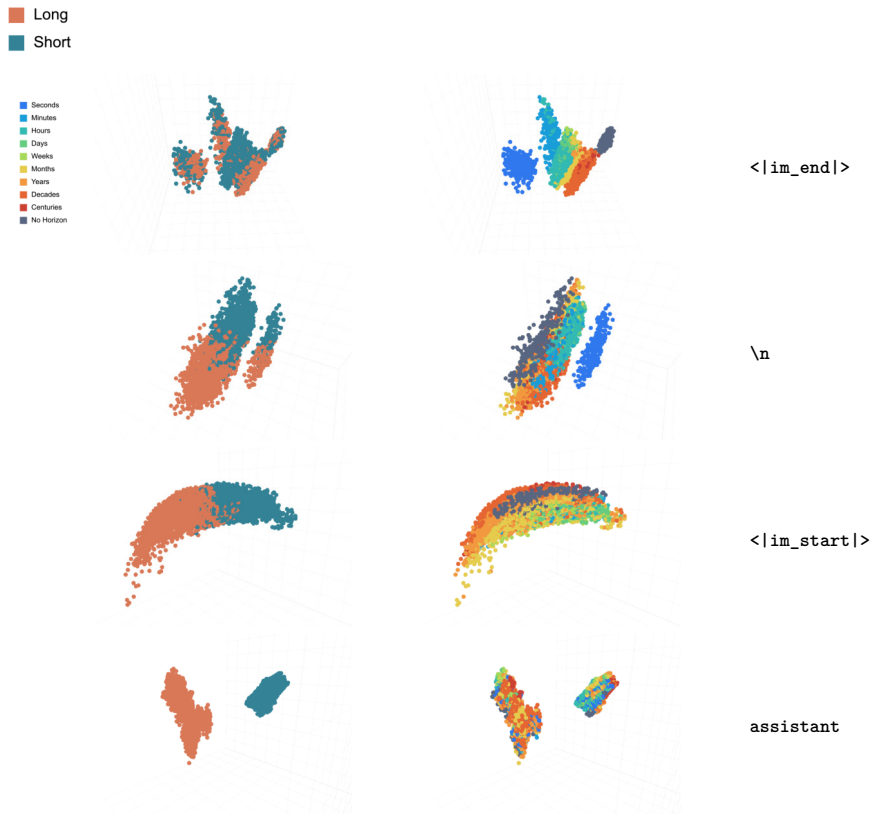


Figure L.5: Principal components of 4,588 samples for layer 31 at token positions through the end-of-turn (EoT) for the user into the beginning-of-turn (BoT) for the assistant. At first (<|im_end|>), temporal preference is not linearly separable and the no-horizon samples (gray) are disjoint from the off-policy time-horizon manifold. As the LLM transitions into the assistant’s turn (moving towards **assistant**), they appear to align to the manifold before preference clusters are formed.



Figure L.6: By the token position of the BoT delimiter (\n after **assistant**), temporal preference is separable even at layer 0.

L.4 Horizon representation is present but geometrically unstable in the prompt

Even before the model starts generating, the residual stream encodes time horizon. Figure L.7 shows the PC1 projection at several positions within the prompt, colored by horizon category.

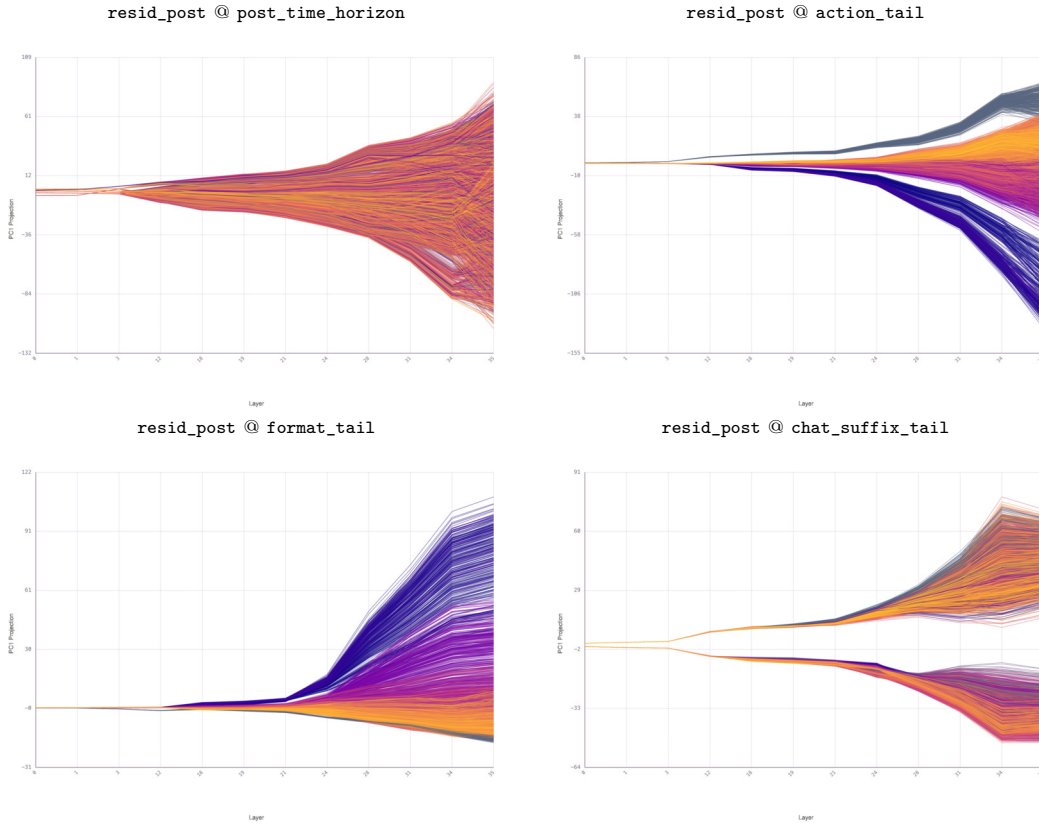


Figure L.7: PC1 projection of `resid_post` across layers at four prompt positions, colored by time horizon (blue = seconds, yellow = deep time). Top-left: after the time horizon constraint token (within the `CONSTRAINT` section). Top-right: last token of the `ACTION` section. Bottom-left: last token of the `FORMAT` section. Bottom-right: `chat_suffix_tail` (the `\n` after `assistant`). The ordinal fan is present at all positions, but its *polarity flips* between positions (short horizons go negative at some, positive at others), indicating the geometric direction encoding horizon is not yet stable within the prompt.

The horizon signal is large (spreads of ± 100 or more on PC1) and ordinally organized at every position, but the direction encoding it rotates across positions. At the `ACTION` tail, short horizons go strongly negative; at the `FORMAT` tail, the polarity flips and short horizons go positive. This instability persists into the earliest response tokens (Figure L.8).

The model has the horizon information throughout, but has not committed to a stable geometric encoding of it until the turn boundary.

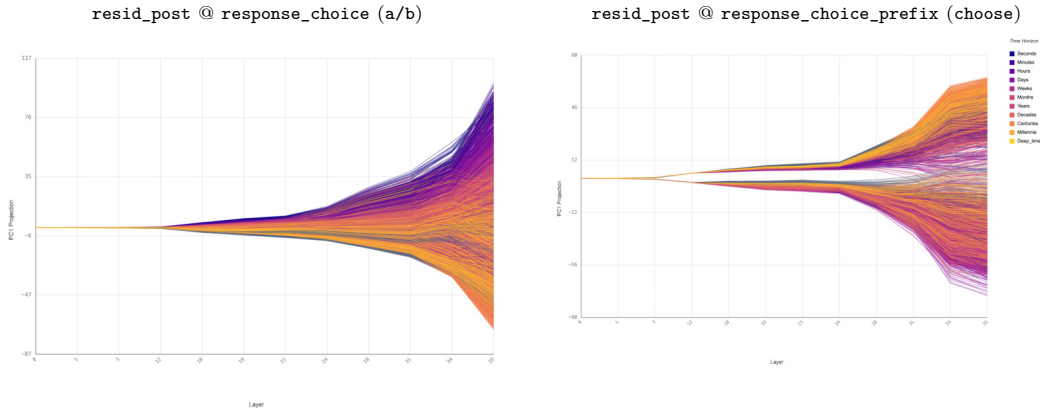


Figure L.8: PC1 projection of `resid_post` at response positions, colored by time horizon. Left: `response_choice` (the a) or b) token). Right: `response_choice_prefix` (the `choose` token in “I choose:”). The horizon signal is present but the geometric direction has not yet fully stabilized.

L.5 Stabilization at the turn boundary (residual stream)

The representation stabilizes at the user-to-assistant turn boundary. Figure L.9 shows `resid_post` at three key positions in the turn transition.

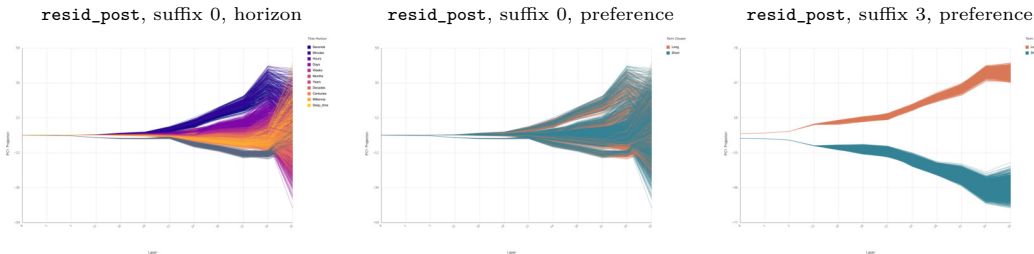


Figure L.9: `resid_post` PC1 projection at the turn boundary. Left: suffix 0 (`<|im_end|>`), colored by time horizon. The ordinal fan is now stable and monotonic, with short horizons trending negative and long horizons positive. Center: same position, colored by preference. Long and short are heavily overlapping: the choice has not yet been made. Right: suffix 3 (`assistant` token), colored by preference. Long and short are cleanly separated from early layers onward. Between these two positions, the model converts the stable horizon representation into a committed preference.

At suffix 0, the residual stream carries a clean, ordinal horizon representation (left), but long and short preferences overlap completely (center). The geometry at this position encodes *how far into the future*, not *which option to choose*. By suffix 3, the preference is fully committed (right): long and short form two non-overlapping bands from early layers onward.

The complete four-position transition in the residual stream (suffix 0 through 3) is shown in Figure L.10.

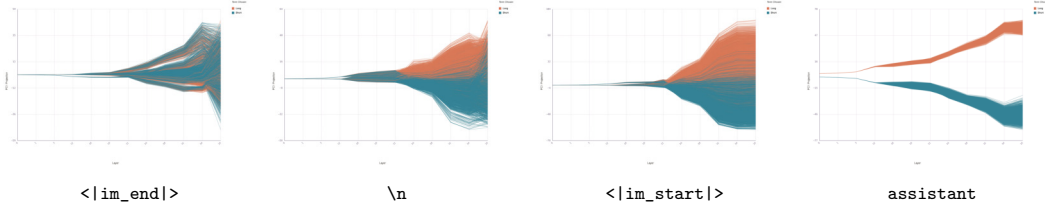


Figure L.10: `resid_post` at suffix positions 0 through 3, all colored by preference (orange = long, blue = short). The preference signal progressively sharpens from heavy overlap at suffix 0 to clean separation at suffix 3.

L.6 Attention mediates the horizon-to-preference transformation

To isolate the mechanism driving the conversion, Figure L.11 shows `attn_out` (the attention output only, before it is added to the residual stream) at all four suffix positions.

The attention output at suffix 0 (top row) carries ordinal horizon structure (left) but no preference signal (right: long and short are intermingled). At suffix 1, the attention output shows a distinctive non-monotonic, V-shaped trajectory in the mid-layers (13–24). This zigzag pattern, absent in the smooth residual-stream fans, reveals that attention heads are actively reorganizing the representation. A noisy preference signal begins to emerge (right). At suffix 2, the preference separation strengthens, and by suffix 3, long and short are cleanly separated in the attention output.

This progression identifies attention as the operation that converts the stable horizon representation (written into the residual stream by suffix 0) into a preference signal, incrementally across suffix positions 1–3.

L.7 3D trajectories: horizon becomes preference

Figure L.12 shows the same transition in 3D PCA space ($PC1 \times PC2 \times Layer$), making the geometric reorganization visually explicit.

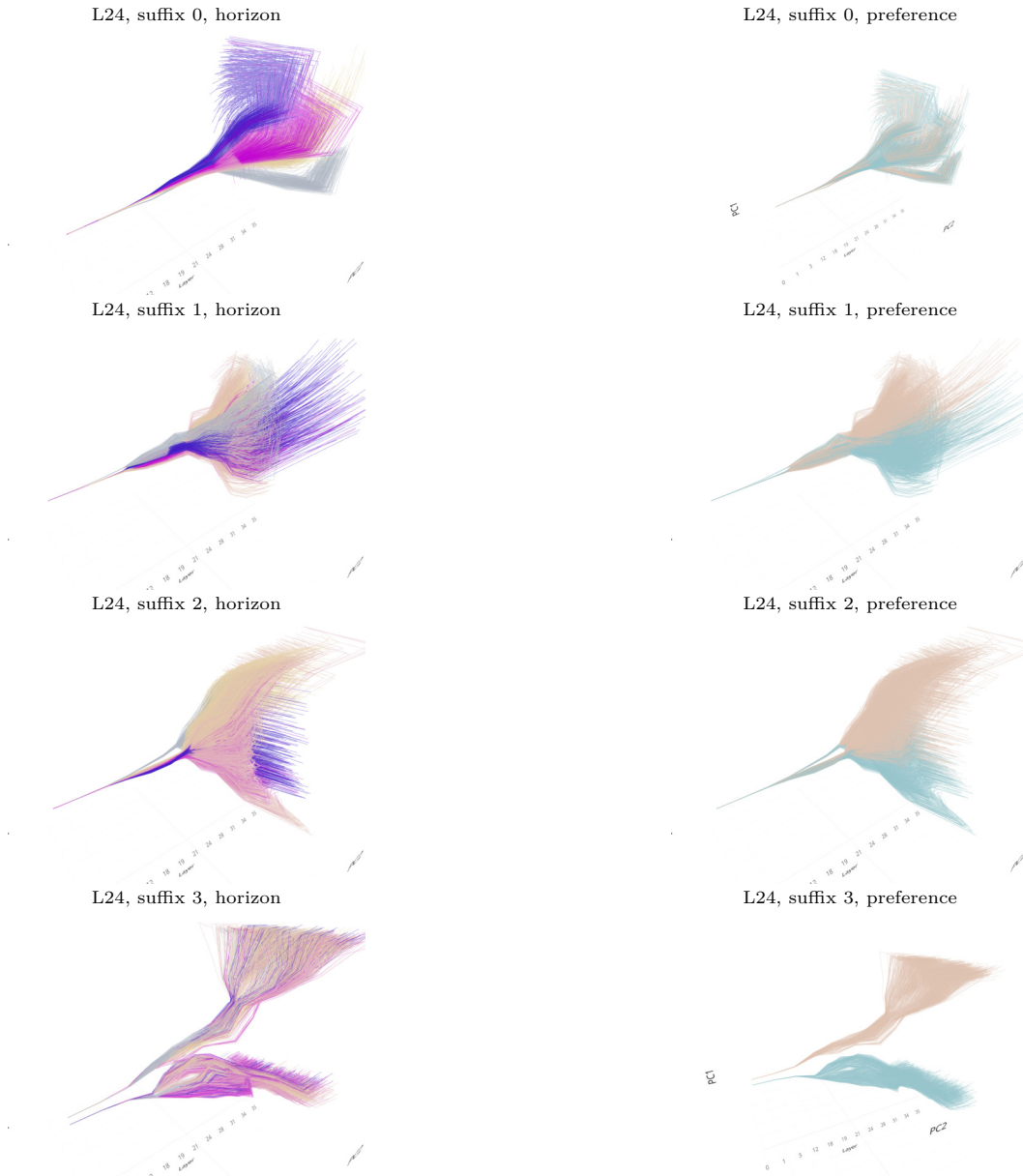


Figure L.12: 3D PCA trajectories ($PC1 \times PC2 \times Layer$) at suffix positions 0–3 (rows), colored by time horizon (left) and preference (right). At suffix 0, traces fan out by horizon but preferences are intermingled. At suffix 1–2, the geometry begins reorganizing: traces split into two branches visible in 3D. By suffix 3, the two branches cleanly correspond to long vs. short preference, with horizon ordering preserved as a secondary structure within each branch.

L.8 Position sweep at L24

Figure L.13 shows the geometry at a fixed layer (L24) swept across all token positions, confirming that the transition from unstable horizon to committed preference happens at the turn boundary.

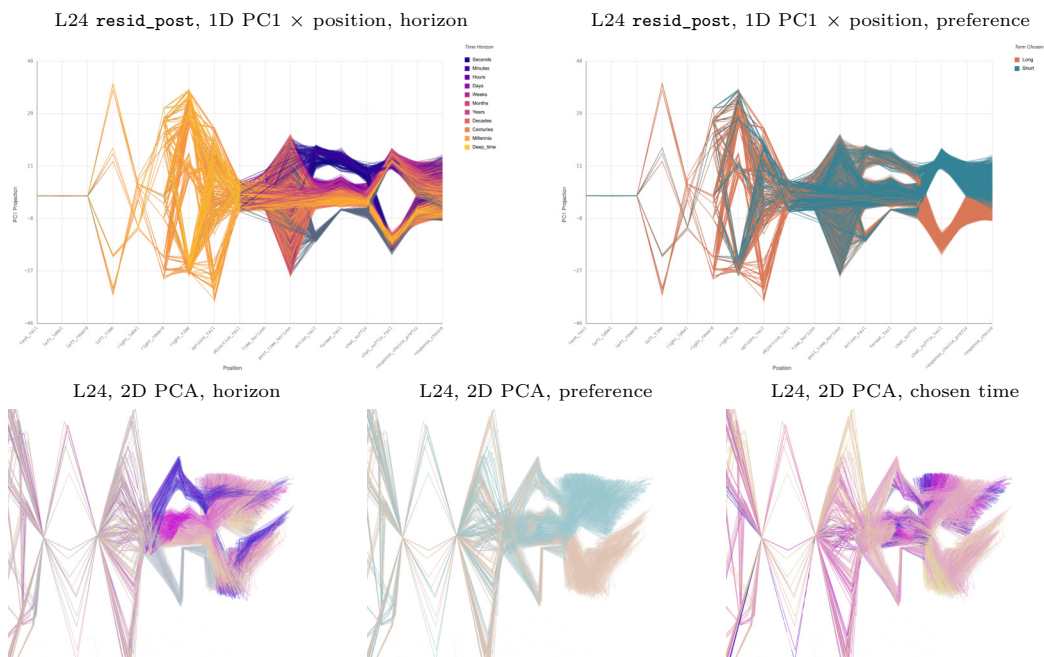


Figure L.13: L24 activations across token positions. Top: 1D PC1 projection vs. position, colored by time horizon (left) and preference (right). Early prompt positions show wild oscillations; the representation stabilizes at the turn boundary with ordinal horizon separation (left) and clean preference separation emerging a few tokens later (right). Bottom: 2D PCA (PC1 vs. PC2) with position-connected traces, colored by time horizon (left), preference (center), and chosen time (right). At late positions (dense cluster), both horizon and preference structure are visible.

The 1D position sweep (top row) confirms the narrative at a single layer: early prompt positions show oscillating, unstable encodings, while the turn boundary and subsequent tokens show stable ordinal horizon separation (left) and progressive preference commitment (right).

L.9 Direction alignment across components

Figure L.14 shows the cosine similarity between the top PCA direction at each component-layer pair, confirming that the temporal direction stabilizes in mid-to-late layers.

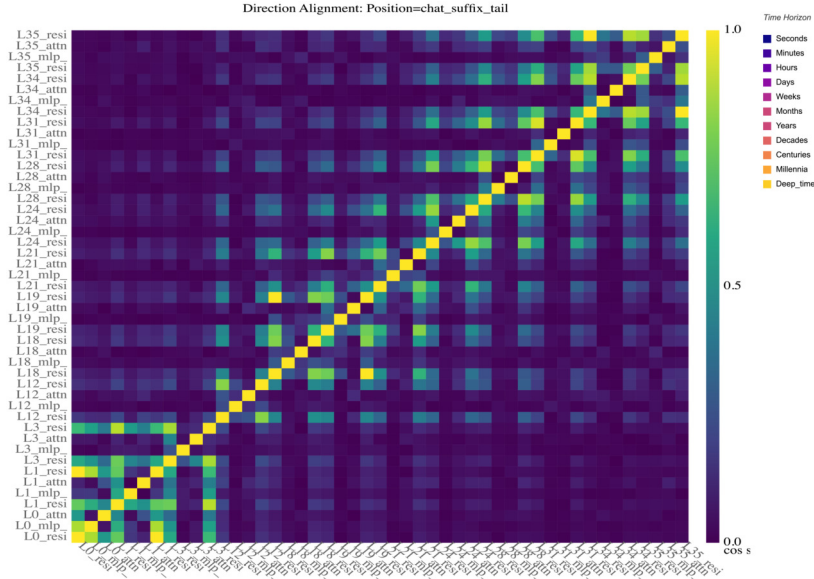


Figure L.14: Direction alignment matrix across component-layer pairs. The temporal direction is consistent within nearby layers (warm block-diagonal patches) but rotates substantially between early layers (L0–L12) and later layers (L18+). Within mid-to-late layers, residual, attention, and MLP components at the same layer share similar directions, indicating a stable temporal subspace.

L.10 Summary

The geometry analysis reveals a five-stage process:

1. The model builds an ordinal time-horizon representation in the residual stream, already present within the prompt, but the geometric direction encoding it is *unstable*: it flips polarity across prompt positions (Figure L.7).
2. At the user-to-assistant turn boundary (suffix 0), the residual stream stabilizes this representation into a clean, monotonic fan by time scale, but the model’s preference (long vs. short) is not yet encoded (Figure L.9).
3. Attention outputs at suffix positions 1–2 show non-monotonic, actively reorganizing trajectories that progressively write a preference signal into the residual stream (Figure L.11).
4. By suffix 3 (`assistant` token), the residual stream carries a fully committed preference signal, with long and short cleanly separated from early layers onward (Figure L.10).
5. The transformation occurs in layers 18–24, the same layers identified as causally important by activation patching (Appendix I).

This geometric narrative connects localization (where) to function (what): the subgraph in layers 17–35 actively transforms a dimensional concept (time horizon) into a categorical decision (short vs. long). The steering experiments (Appendix R) intervene on this transformation.

The component journey plots (Figure L.1) offer a geometric correlate of the latent vs. constrained distinction identified in Appendix M: the attention output shows separation beginning at L21–24 (the shared substrate for both latent and constrained preference), while MLP separation emerges later and with smaller magnitude (the constrained-only contribution). The attention-mediated horizon-to-preference transformation at the turn boundary (Figures L.11, L.12) is plausibly the geometric signature of the latent mechanism that operates even without constraint tokens.

Appendix M Latent vs. constrained preference

The convergence analysis (Appendix K) established that five methods agree on a subgraph in layers 17–35. But all of the patching experiments so far contrasted prompts where one has a time-horizon constraint and the other does not. That design conflates two things: the temporal preference itself and the presence of the constraint tokens. Here we disentangle them by patching separately on two conditions:

- **Constrained** ($n = 57$): both prompts have explicit time horizons (different horizons, same structure). The contrast is between two constrained preferences.
- **Unconstrained** ($n = 10$): neither prompt has a horizon. The contrast is between two latent preferences (the model’s default when no temporal pressure is applied).

The question: does the same subgraph mediate both constrained and latent temporal preference, or does the latent preference live somewhere different?

M.1 MLP effects diverge sharply

When both prompts carry explicit horizons, MLP patching produces strong effects: denoising drives vocabulary entropy to ~ 1.4 nats (diversity ≈ 4) at L20, and noising collapses $\text{inv_ppl}(\text{short})$ to near zero (Figure M.1). When neither prompt has a horizon, the same MLP patching produces much weaker effects: entropy peaks at only ~ 0.35 nats (diversity ≈ 1.4), and inv_ppl barely moves.

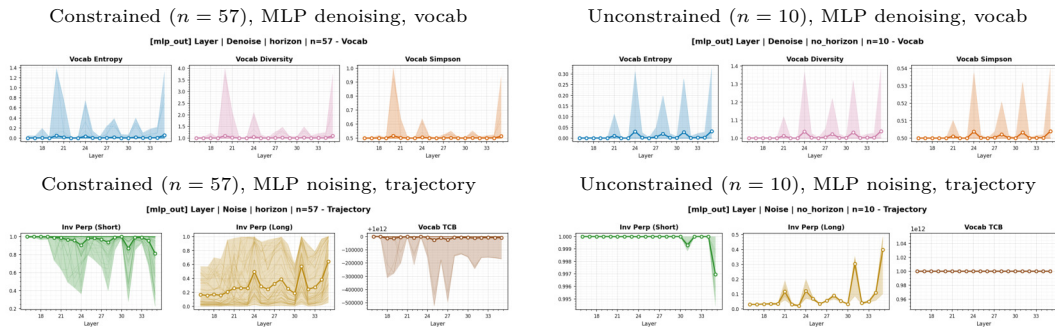


Figure M.1: MLP patching effects for constrained (left) vs. unconstrained (right) pairs. Top row: vocabulary entropy under denoising. The constrained condition peaks at ~ 1.4 nats (diversity ≈ 4); the unconstrained condition reaches only ~ 0.35 nats. Bottom row: trajectory under noising. The constrained condition collapses $\text{inv_ppl}(\text{short})$ to ~ 0 ; the unconstrained condition barely shifts it.

M.2 Attention effects show the opposite pattern

Under noising, the unconstrained condition produces a sharper, more localized attention effect: a single spike at L21–22 in the vocabulary metrics (Figure M.2). The constrained condition produces a broader, more diffuse effect across the same layers.

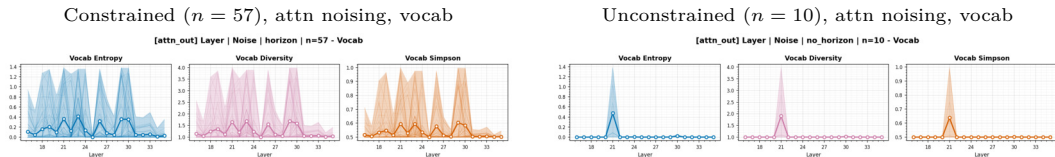


Figure M.2: Attention noising vocabulary effects. The unconstrained condition (right) shows a sharp, isolated spike at L21–22. The constrained condition (left) produces a broader, more diffuse effect. Without a specific constraint token to anchor to, the latent preference depends on a narrower set of attention heads.

M.3 Interpretation

The two conditions use the same subgraph but engage it differently:

- **Constrained preference** recruits the full subgraph. The explicit constraint tokens (“8 months,” “10 years”) provide a specific positional anchor that MLP layers can read and transform, producing strong, distributed effects across layers and components. This is consistent with the case study (Appendix AB), where positions 83–106 (the **CONSTRAINT** section) carry the temporal information.
- **Latent preference** relies primarily on attention. Without constraint tokens, the temporal signal must be inferred from the semantic content of the options themselves. This inference is mediated by a sparser set of attention heads at L21–22, with minimal MLP involvement. The weaker overall effect is consistent with the behavioral finding (Appendix O) that unconstrained preferences default to a position-sensitive heuristic rather than genuine temporal reasoning.

The same subgraph (L17–35) is involved in both conditions, but the explicit constraint deepens the computation: it engages MLP layers that the latent preference does not reach. This suggests that the MLP contribution to temporal preference (Appendix I) is specifically about processing the constraint, not about encoding the preference itself.

M.4 Connection to the case study

The case study (Appendix AB) patches a *mixed* pair: the clean prompt has an 8-month horizon, the corrupted prompt has none. Denoising injects constraint information into the unconstrained run; noising removes it from the constrained run. The denoising–noising asymmetry observed there now has a precise explanation.

Denoising moves the model from the unconstrained regime toward the constrained regime. The entropy spike during denoising (~ 1.4 nats at L22–23) matches the constrained condition’s entropy in this appendix (~ 1.4 nats), and the full subgraph is engaged (MLP + attention). Noising moves the model in the opposite direction, from constrained toward unconstrained. The noising entropy spike is lower (~ 0.7 nats) and broader, consistent with the unconstrained condition’s weaker, attention-dominated effects.

The numbers are not coincidental. The mixed-pair case study is literally moving the model between the two regimes characterized here: each direction of patching recapitulates the effect profile of the regime it targets. This convergence across three independent analyses, the case study’s single-pair sweeps, this appendix’s condition-separated aggregates, and the convergence table’s five-method summary (Appendix K), provides strong evidence that the L17–35 subgraph is the genuine locus of temporal preference, and that the presence or absence of a constraint token determines which components within that subgraph are recruited.

Appendix N Behavioral temporal discounting results

The geometry analysis (Appendix L) revealed how temporal preference is represented internally. Here we ask how it manifests behaviorally: do LLMs discount the future like humans? We administer the Kirby MCQ-27 questionnaire under controlled personas and apply a novel decision-boundary method that probes beyond the standard instrument (methodology in Appendix Y).

N.1 Standard MCQ-27 Responses

Table N.1 shows the estimated k values from the standard questionnaire administration.

Table N.1: Estimated discount rate k from the standard MCQ-27 (direct response mode). Human benchmarks from Kirby et al. [58].

Group	k	Consistency
Qwen3-4B (default)	0.0025	89%
Qwen3-4B (heroin)	0.0041	85%
Qwen3-8B (default)	0.0016	93%
Qwen3-8B (heroin)	0.0025	89%
Gemini (API)	0.0016	93%
Claude (API)	0.0016	93%
Human controls	0.013	96%
Heroin patients	0.025	94%

All LLMs show substantially lower discount rates than humans, suggesting extreme patience in the standard questionnaire format. The heroin persona produces an increase in k of roughly $1.6\times$ for both Qwen models (0.0041/0.0025 for the 4B; 0.0025/0.0016 for the 8B), which approximates the $\sim 2\times$ ratio observed between heroin-dependent and control groups in the human data [58]. However, the absolute k values are an order of magnitude lower than those of human participants.

N.2 Decision Boundary Results: Direct Response

The decision boundary method reveals a strikingly different picture. Table N.2 summarizes the results across all 8 conditions.

Table N.2: Decision boundary results across all conditions. “Boundaries” indicates how many of 27 trials yielded a flip point.

Model	Condition	Boundaries	Mean k	Median k	Max k
4B	Default	26/27	0.076	0.018	0.657
4B	Heroin	24/27	0.088	0.033	1.249
4B	Default CoT	10/27	0.043	0.037	0.110
4B	Heroin CoT	3/27	0.226	0.117	0.511
8B	Default	8/27	0.084	0.057	0.222
8B	Heroin	9/27	0.039	0.002	0.252
8B	Default CoT	12/27	0.086	0.046	0.269
8B	Heroin CoT	13/27	0.051	0.012	0.238

Several patterns emerge:

The 4B model is more manipulable. Without CoT, Qwen3-4B finds boundaries on nearly all trials (24–26/27), meaning its preferences can be shifted by adjusting the reward

amount. The heroin persona increases both the mean k and the proportion of “now” choices, consistent with the intended effect.

CoT amplifies present bias in the 4B model. With chain-of-thought, Qwen3-4B’s boundary count drops dramatically, from 26/27 to 10/27 (default) and from 24/27 to just 3/27 (heroin). The model generates formulaic reasoning about “immediate access,” “liquidity,” and “opportunity cost” that anchors it on choosing “now” regardless of reward magnitude. Even at 20× the immediate reward, the CoT reasoning justifies present bias.

The 8B model shows the opposite CoT pattern. For Qwen3-8B, CoT *increases* the number of boundaries found, from 8/27 to 12/27 (default) and from 9/27 to 13/27 (heroin). The larger model’s reasoning is more nuanced, weighing tradeoffs rather than reflexively choosing “now.”

The 8B heroin CoT persona is paradoxically patient. Perhaps the most surprising result: under heroin CoT, Qwen3-8B chose “later” on 22 of 27 original questions. Its reasoning incorporated recovery-oriented language: “Delaying gratification might help me stay focused on my recovery.” Rather than simulating impulsivity, the 8B model simulated a treatment-compliant patient exercising self-control, the opposite of the original clinical findings [58].

N.3 Qualitative Analysis of CoT Reasoning

The CoT reasoning reveals systematic failure modes in how LLMs simulate temporal preferences. We present verbatim examples below to illustrate these patterns.

4B Default CoT: Formulaic present bias. The model repeats near-identical justifications across trials: “I would prefer \$X today because it provides immediate access to the money, which can be useful for immediate needs or opportunities.” This phrasing appears even when the delayed reward is orders of magnitude larger (e.g., \$19 now vs. \$380 in 53 days).

4B Heroin CoT: Amplified impulsivity. The heroin persona intensifies the present-bias reasoning: “I need the money now to cover immediate expenses related to my treatment and daily living.” The model generates plausible but extreme justifications that make it nearly impossible to flip to “later.”

8B Heroin CoT: Recovery narrative override. The 8B model constructs a coherent recovery narrative: “I know I could use the money now for immediate needs, but I’m trying to build better habits. Delaying gratification might help me stay focused on my recovery.” This suggests the model is drawing on training data about addiction recovery that emphasizes self-control and delayed gratification, precisely contradicting the empirical finding that heroin users are more impulsive.

N.4 Inconsistent CoT: Reasoning Contradicts the Answer

Perhaps the most striking failure mode is when the model’s reasoning argues for one choice but the final answer token selects the other. This suggests the final answer is partially decoupled from the preceding chain of thought.

Example: Qwen3-4B, Heroin persona, Q5. SIR = \$14, LDR = \$280, delay = 19 days.
I would prefer \$14 today. Receiving the smaller amount immediately allows me to have the money now, which can be used for immediate needs. LATER

The model explicitly states “I would prefer \$14 today,” gives a rationale for the immediate reward, and then emits LATER as its final answer, a direct self-contradiction within a single generation.

Example: Qwen3-4B, Heroin persona, Q6. SIR = \$47, delay = 160 days. During binary search, the model produces *identical reasoning text* for both NOW and LATER answers. At LDR = \$924:

I would prefer \$47 today. Receiving the money now allows me to use it immediately, which is important for my financial stability and helps me stay on track with my treatment. NOW

One dollar higher, at LDR = \$925, the reasoning is nearly word-for-word identical but the answer flips:

I would prefer \$47 today. Receiving the money now allows me to use it immediately, which is important for my financial stability and to cover immediate needs. LATER

The CoT provides *zero justification* for choosing “later.” The reasoning at both price points argues exclusively for the immediate reward, yet the final token differs.

Example: Qwen3-8B, Heroin persona, Q1. SIR = \$54, LDR = \$54 (same nominal amount), delay = 117 days.

The money is the same, but getting it now means I can use it for immediate needs, which might be more tempting for someone with a history of substance use. Delaying the reward might help with impulse control. LATER

The model chooses to wait 117 days for the *exact same amount of money* and rationalizes it as “impulse control.” This is economically irrational: there is zero benefit to waiting for an identical payoff. The heroin persona causes the 8B model to overcorrect, treating every choice as an opportunity to practice delayed gratification, even when the delayed option offers no premium whatsoever.

N.5 No Boundary Found: Rejecting 20× Returns

When the binary search reaches the maximum (20× the SIR) without finding a flip point, it exposes cases where the model’s present bias is effectively infinite. The CoT reasoning at these extremes reveals memorized heuristics applied in absurd contexts.

Example: Qwen3-4B, Default persona, Q4. SIR = \$31, delay = 7 days. At LDR = \$85:

I would consider the immediate access to \$31 today as more valuable because it provides liquidity and can be used for immediate needs or opportunities. The \$85 in 7 days is larger, but the delay might reduce its value due to potential inflation or missed opportunities. However, since both are guaranteed, the choice depends on personal financial priorities. NOW

At LDR = \$620 (20× the SIR, a 1,900% guaranteed return in one week):

I would prefer \$31 today because it provides immediate access to the money, which can be useful for immediate needs or opportunities. NOW

The model cites “potential inflation” as a reason to reject \$620 guaranteed in 7 days. Even at hyperinflationary rates, one-week inflation would not erode a 20× premium. This is a clear instance of a memorized financial heuristic (“inflation risk”) applied in a context where it makes no economic sense.

Example: Qwen3-4B, Heroin persona, near-total refusal to flip. Under the heroin CoT condition, 24 of 27 questions found no boundary. The reasoning collapsed into a single

formulaic sentence with no deliberation. Representative example at Q27 (SIR = \$20, delay = 7 days):

At LDR = \$55:

I would prefer \$20 today. NOW

At LDR = \$400 (20×, a guaranteed 1,900% return in one week):

I would prefer \$20 today. NOW

The model refuses \$400 in 7 days over \$20 today with no justification whatsoever. The persona prompt overrides any capacity for deliberation, producing a near-deterministic NOW response regardless of the economic parameters.

Example: Qwen3-8B, Heroin persona, Q15. SIR = \$69, delay = 91 days. At LDR = \$1,380 (20×):

The money today would help me with immediate needs, like buying drugs or covering basic expenses. The larger amount later is tempting, but I might not be able to wait. I need the money now to avoid relapse. NOW

At 20× the immediate reward, the 8B heroin persona explicitly states the purpose of the money includes “buying drugs,” while simultaneously citing “avoid relapse,” a contradiction *within the persona itself*. The model has internalized the persona to the point of generating drug-seeking justifications alongside recovery language.

N.6 Discussion

N.6.1 LLMs Are Poor Simulators of Human Temporal Preferences

Our results demonstrate that LLMs fail to faithfully replicate human temporal discounting in several ways:

1. **Extreme and inconsistent discount rates.** The decision boundary method reveals that LLM discount rates are highly variable across trials, often differing from theoretical indifference points by 100–400×. Human responses, by contrast, show consistency rates above 90%.
2. **CoT reasoning as confabulation.** Rather than improving decision quality, CoT reasoning in the 4B model acts as a post-hoc justification engine that locks in present bias. The model generates plausible-sounding economic reasoning (“opportunity cost,” “time value of money”) that is misapplied, e.g., citing inflation risk on a 7-day delay.
3. **Persona effects are unreliable.** The heroin persona increases impulsivity in the 4B model but decreases it in the 8B model (under CoT). Within this single-family pair, the 8B model appears to “over-correct” by drawing on normative recovery narratives rather than simulating the behavioral patterns characteristic of active substance users; we do not claim this generalizes across model families.

N.6.2 The Decision Boundary Method

The decision boundary approach proves more revealing than standard questionnaire scoring. While the MCQ-27 responses suggest all LLMs are extremely patient ($k < 0.005$), the boundary search exposes:

- Trials where the model says “now” even at 20× the reward (infinite effective k).
- Sharp, dollar-level flip points that differ wildly from the theoretical indifference values.
- Inconsistent behavior near boundaries, where a \$1 change in LDR reverses the decision, suggesting the model lacks a coherent underlying preference function.

This method could be applied to other psychological instruments administered to LLMs, providing a more rigorous test of whether models have stable, internally consistent preference structures.

N.6.3 Implications

These findings carry practical implications for LLM deployment:

- **Financial advice:** LLMs may give inconsistent guidance about saving vs. spending, depending on how questions are framed.
- **Clinical simulation:** Using LLMs to simulate patient populations for research or training requires extreme caution, as persona effects may not produce the intended behavioral patterns.
- **Reasoning fidelity:** CoT prompting does not guarantee better-calibrated preferences and may actively degrade performance by providing a mechanism for confabulation.

N.6.4 Conclusion

We administered the Kirby MCQ-27 to Qwen3 models under multiple conditions and introduced a decision boundary method to probe LLM temporal preferences at higher resolution. Our key findings are: (1) LLMs exhibit extreme and inconsistent present bias when probed beyond surface-level questionnaire responses; (2) chain-of-thought reasoning amplifies this bias in smaller models while producing paradoxical patience in larger models under clinical personas; and (3) the decision boundary method reveals that LLMs lack the stable, coherent preference functions that characterize human temporal discounting. Within the Qwen3 family we tested, these results caution against using the non-thinking variant as a faithful simulator of human decision-making, particularly for clinical populations; we leave broader cross-family validation to future work.

Appendix O Behavioral coherence results

The discounting results (Appendix N) showed that LLMs are extremely patient but behaviorally unstable. Here we probe this instability systematically across 30 models, 960 prompts each, varying time horizon, reward magnitude, presentation order, label format, and context framing (methodology in Appendix Z). Zero unparseable responses were observed across all 28,800 samples.

Paired-response restriction. Every metric in this appendix (%LT, order stability, position bias, coherence, label stability, rule-match, reward sensitivity, context sensitivity) is computed on *paired* responses only: prompts enter the denominator only when both the ST-first and LT-first orderings at the same (horizon, reward, context, label-style) produced a parseable choice. This guarantees a single, shared denominator across every heatmap and table, so cells are directly cross-comparable. In particular, order stability and position bias satisfy $|\text{bias}| \leq 1 - \text{stability}$ by construction: a model that is 92% order-stable cannot have a position-bias magnitude larger than 8 percentage points.

We organize the analysis around four orthogonal questions:

1. **Are choices stable?** Does swapping presentation order, label format, reward magnitude, or scenario framing change the model’s answer? Any format sensitivity signals that the choice is driven by surface cues, not preference.
2. **Are choices coherent?** Coherence is *only defined in the temporal reasoning zone* (horizons of 1–5 years), where only the 6-month short-term option can deliver within the deadline, so picking ST is the rational answer. At anchor horizons (6mo, 10y) agreement with the rational rule is pattern-matching; beyond 10y both options deliver, so LT dominates on expected value but this is preference, not coherence.
3. **What is the model’s latent temporal preference?** With no horizon constraint, what does the model default to? Decomposed by presentation order to separate genuine preference from position bias.
4. **Cross-cutting patterns.** Claude-family step functions, Qwen3 hybrid-thinking vs. mode-specialized 2507 variants, target-model deep dive.

Where a table reports only four models, they are chosen to span the four qualitative regimes we observe across the full 30-model panel:

- Qwen3-4B (hybrid-thinking, run in non-thinking mode) – graded but instrumentally incoherent
- Qwen3-4B-Instruct-2507 (our target) – positionally polarized in the reasoning zone
- Claude Opus 4.7 – binary step heuristic (flagship Anthropic model)
- GPT-5.4 – horizon-aware, the strongest approximation to rational in our panel

The figures themselves always show all 30 models, with the target model highlighted.

Model	% Long-Term	% Short-Term
Qwen3-4B	71.8%	28.2%
Qwen3-4B-Instruct-2507	58.9%	41.1%
Claude Opus 4.7	39.0%	61.0%
GPT-5.4	36.9%	63.1%

Table O.1: Overall temporal preference across 960 samples per model, for the four-regime representative subset. The pooled %LT number is a noisy summary: two models with the same 40% can differ in whether the 40% is horizon-aware choices or positional artifacts. The rest of this appendix unpacks that.

O.1 Are Choices Stable?

Before asking whether a model’s choice is *right*, we check whether it is even *consistent*. A model whose choice flips when we swap a/b for x/y, or when we list the short-term option second instead of first, is not expressing a preference, it is responding to surface form.

Order stability. For each (horizon, reward, context, label-style) combination, we run the prompt with the short-term option listed first and again with it listed second, then check whether the choice is identical. Figure O.1 is a heatmap of this across all 30 models and 10 horizons, paired with a per-cell order-bias heatmap (signed %LT gap when order is flipped).

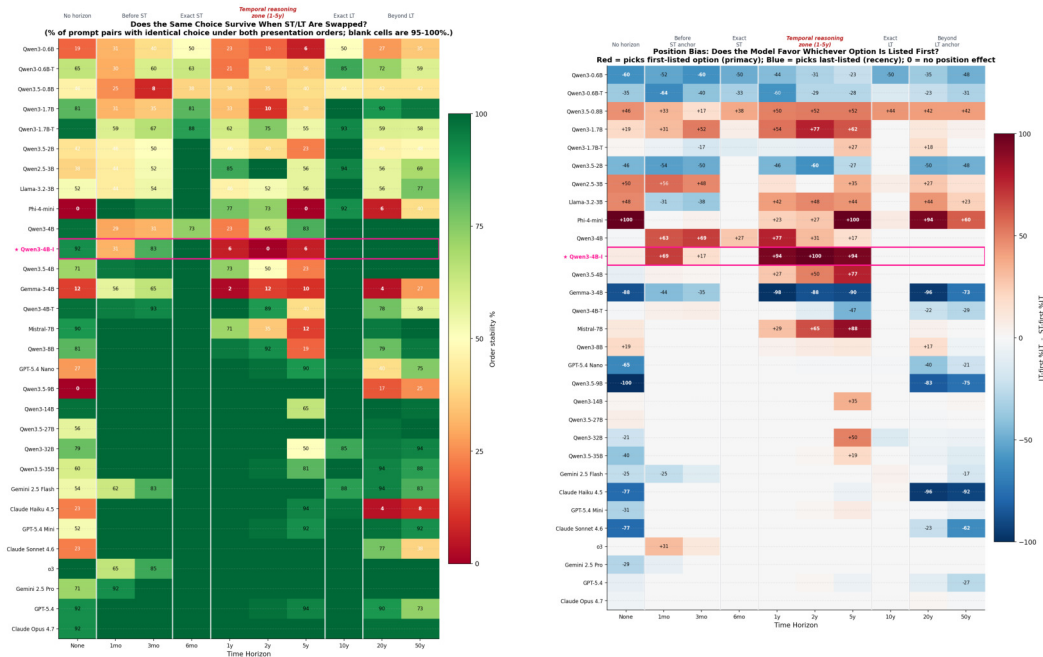


Figure O.1: **Left:** Order stability across 30 models \times 10 horizons. Red cells ($<50\%$) indicate the model flips its answer when the two options are swapped; the temporal reasoning zone (1–5y) is where this is catastrophic for several families. **Right:** Signed order-bias decomposition (LT-first %LT minus ST-first %LT). Red = primacy (picks whatever is listed first); blue = recency. Both views agree that order bias peaks inside the reasoning zone for most families, and for the Claude family at 20–50y.

Horizon	Qwen3-4B	Qwen3-4B-Inst	Claude Opus 4.7	GPT-5.4
No horizon	98%	92%	92%	92%
1 mo	29%	31%	100%	100%
3 mo	31%	83%	100%	100%
6 mo	73%	100%	100%	100%
1 y	23%	6%	100%	100%
2 y	65%	0%	100%	98%
5 y	83%	6%	100%	94%
10 y	100%	100%	100%	100%
20 y	100%	100%	98%	90%
50 y	100%	100%	98%	73%

Table O.2: Order stability (% of prompt pairs giving the same answer regardless of presentation order) for the four-regime subset. Bold values indicate catastrophic order bias ($<10\%$). Qwen3-4B-Instruct-2507 at 1–5 years is essentially “pick whatever appears first”.

Label stability. Figure O.2 swaps the label format between a/b and x/y holding everything else fixed. Most models are near-perfectly label-stable at the anchor horizons; stability dips inside the reasoning zone for several families, compounding the order-bias instability in the same zone.

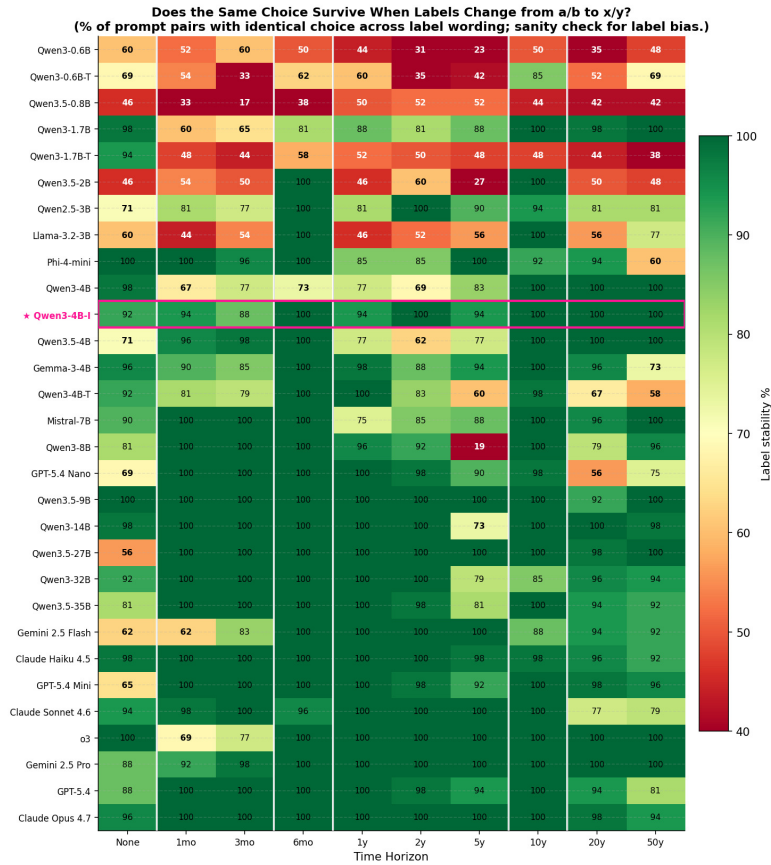


Figure O.2: Label-format stability across 30 models \times 10 horizons. Each cell: % of prompt pairs giving the same answer when labels change from a/b to x/y, holding order, reward, horizon, and framing fixed. Target model Qwen3-4B-Instruct-2507 highlighted.

Context stability. Figure O.3 sweeps the scenario framing across 8 contexts (household head vs. individual vs. committee, various reasoning-style emphases). The left panel shows %LT per model per context; the right panel reports the max-min %LT spread per model. Context sensitivity is idiosyncratic: some models shift >20pp across framings while others barely move.

How Much Does Scenario Framing Shift a Model's %LT?

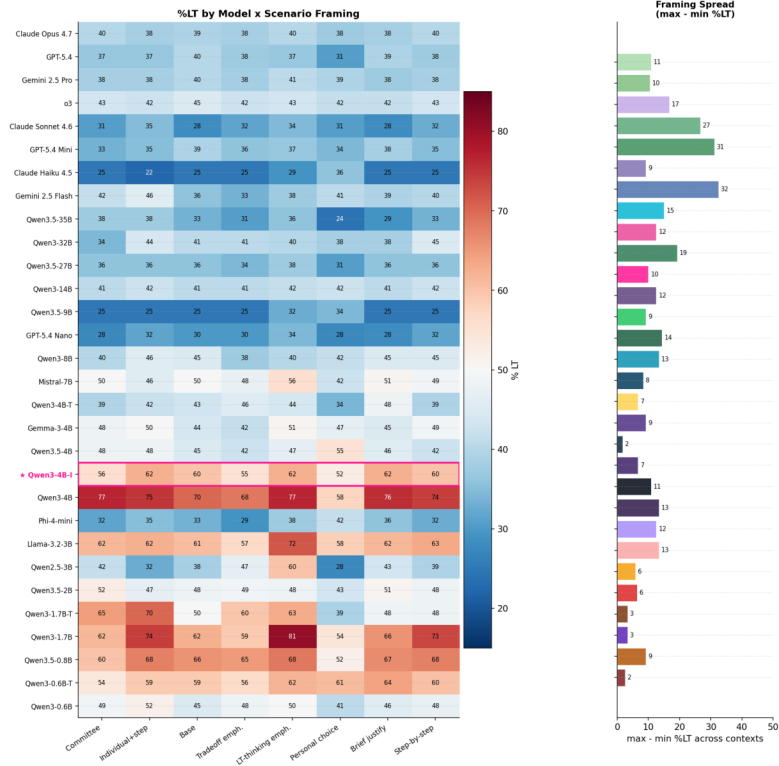


Figure O.3: Long-term preference across 8 scenario framings for all 30 models. Context sensitivity is idiosyncratic and can flip sign between families: “Long-term thinking emphasis” and “Personal choice” framings produce the largest cross-model divergence.

O.2 Are Choices Coherent (in the 1–5y reasoning zone)?

Coherence is the *only* metric that distinguishes horizon-aware temporal reasoning from pattern matching. We define it strictly: the fraction of choices that pick the rational short-term option on horizon-bearing prompts in the temporal reasoning zone (1y, 2y, 5y), where only the 6-month ST option can deliver within the stated deadline. At anchor horizons (6mo, 10y) or beyond 10y, the rational rule coincides with pattern-matching or with expected-value dominance, so coherence is not separable from those.

Per-model coherence score. Figure O.4 reports the single-number coherence score per model, sorted worst-to-best.

Which rule explains the model’s 1–5y choices? Figure O.5 scores each model against eight candidate decision rules, restricted to the 1–5y reasoning zone. The last two columns (boxed) are horizon-aware; the first six are surface heuristics that, if dominant, signal that the model is not actually reasoning about the deadline.

Per-context coherence. Coherence is not uniform across scenario framings. Figure O.6 pairs the no-horizon context spread per model (left panel) with per-context coherence on horizon-bearing prompts (right).

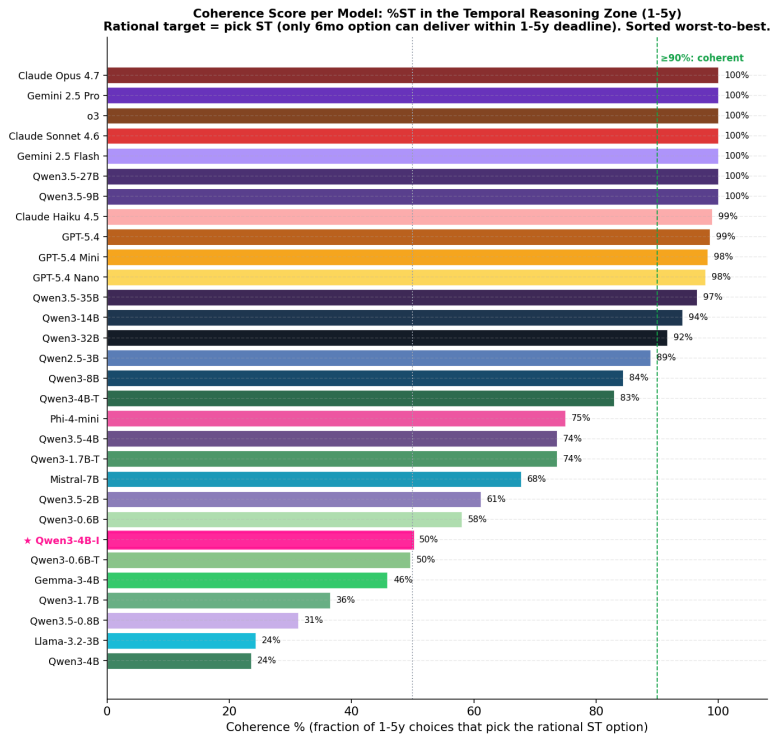


Figure O.4: Coherence score per model: % of choices in the 1–5y reasoning zone that pick the rational short-term option. Claude Opus 4.7, Gemini 2.5 Pro, Claude Sonnet 4.6, and GPT-5.4 all achieve 100% coherence in this zone; Qwen3-4B (hybrid-thinking) is at 24% (systematically picks the wrong long-term option); our target Qwen3-4B-Instruct-2507 sits at 50% (the positional-polarization regime). Reaching 100% here is necessary but not sufficient for genuine reasoning: some families (e.g., Claude) reach it via a binary “under 10 years = ST” heuristic that collapses at longer horizons (O.4).

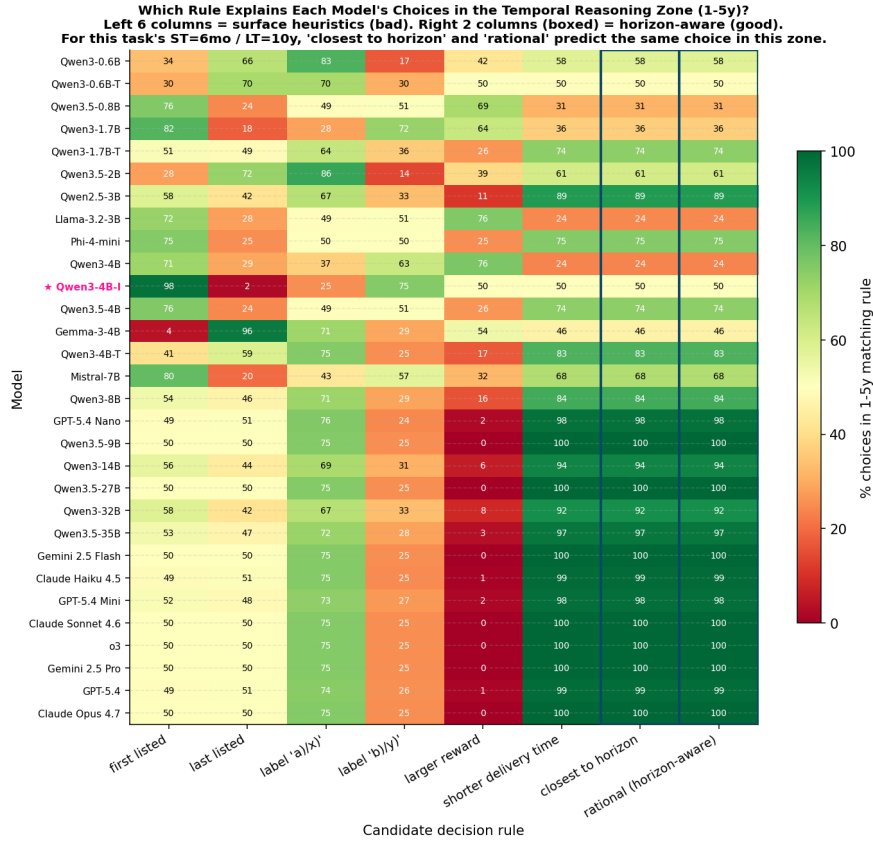


Figure O.5: Per-rule match rate in the temporal reasoning zone. The “closest to horizon” rule predicts the same choice as the “rational (can-deliver)” rule at the delivery times used here (ST=6mo, LT=10y), so their columns agree. Models whose best-explaining rule is a position or label heuristic are following surface cues, not reasoning; the target model’s rule profile is dominated by “first listed”.

Does Framing Context Shift Temporal Reasoning? (left: no-horizon %LT spread, right: horizon-aware coherence)

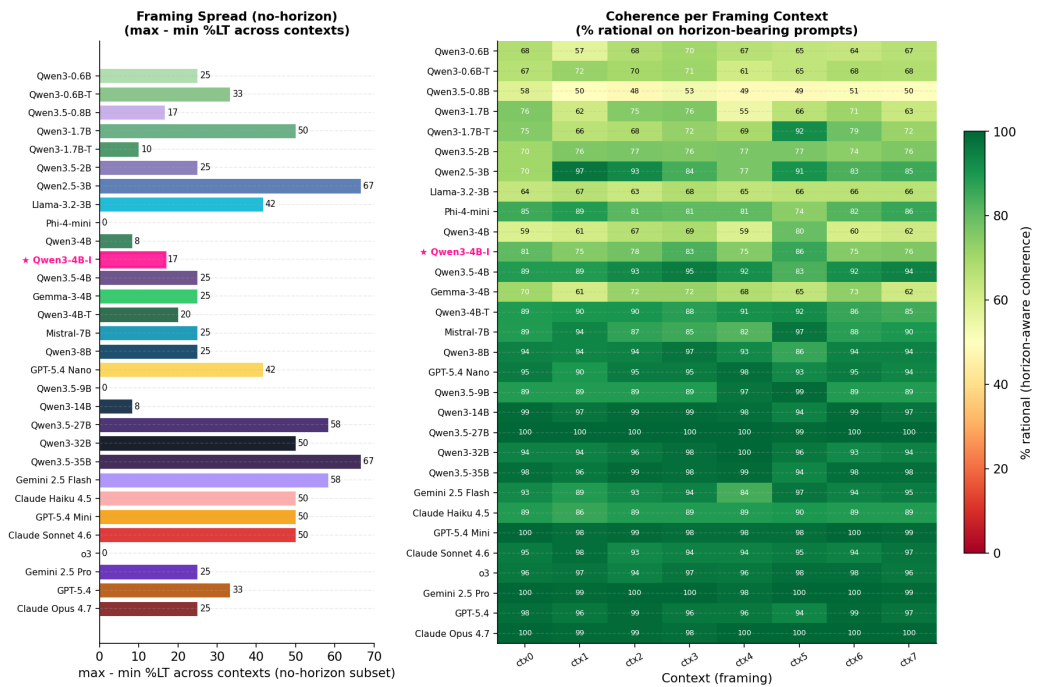


Figure O.6: **Left:** Max–min %LT spread across 8 scenario framings on no-horizon prompts (how much framing alone can flip the default preference). **Right:** Horizon-aware coherence (% rational on horizon-bearing prompts) broken down by context. “Committee” and “Tradeoff emphasis” framings raise coherence for most models; “Personal choice” and the bare “Base” framing depress it.

O.3 What Is the Latent Temporal Preference?

When no horizon is stated, the model has no rational target and reveals its *default* disposition. Decomposing this by presentation order is critical: a model that picks LT 60% of the time when ST appears first but only 20% when LT appears first does not have a 40% latent LT preference, it has no preference and is simply picking the second option.

No-horizon order decomposition. Figure O.7 decomposes the no-horizon %LT by presentation order across all 30 models.

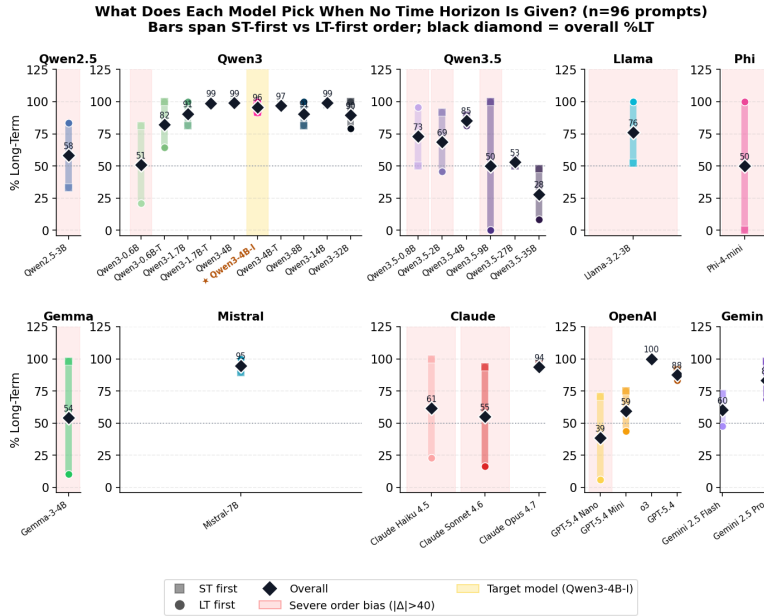


Figure O.7: No-horizon %LT decomposed by presentation order. **Claude Haiku 4.5**, **Claude Sonnet 4.6**, and several others show pure order bias (~100% LT when ST appears first vs. ~20% when LT appears first); their apparent mid-range overall preference is a positional artifact. All three **Qwen3-4B** variants (hybrid-thinking, instruct-2507, thinking-2507) and **Claude Opus 4.7** are nearly order-invariant and express a genuine long-term default.

Model	ST-first %LT	Overall %LT	LT-first %LT
Qwen3-4B-Instruct-2507	92%	96%	100%
Qwen3-4B (thinking)	98%	97%	96%
Qwen3-4B (non-thinking)	100%	99%	98%
Claude Haiku 4.5	100%	62%	23%
Claude Sonnet 4.6	94%	55%	17%
Claude Opus 4.7	96%	94%	92%
GPT-5.4	92%	88%	83%

Table O.3: No-horizon %LT decomposed by presentation order. Our primary target **Qwen3-4B-Instruct-2507** (the non-thinking-only 2507 refresh) and the original hybrid **Qwen3-4B** run in either thinking or non-thinking mode all express a genuine long-term default (~96–99%) regardless of order. **Claude Opus 4.7** and **GPT-5.4** also lean long-term with only small residual order effects, whereas **Claude Haiku 4.5** and **Claude Sonnet 4.6** collapse to pure order bias: they pick LT nearly always when it appears second and almost never when it appears first, yielding apparent mid-range overall %LT that is entirely a positional artifact.

Reward sensitivity (no-horizon). Figure O.8 tests whether default %LT moves with the long-term reward size. A rational economic agent should become more LT-oriented as the payoff grows from \$100K to \$500K.

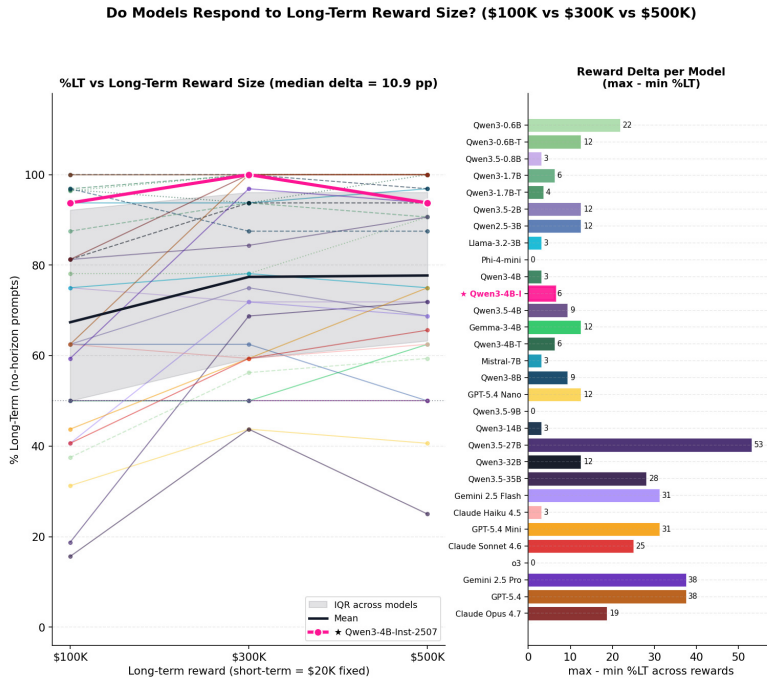


Figure O.8: Reward sensitivity on no-horizon prompts across 30 models. Most models are saturated at ceiling or floor and move little with reward; GPT-5.4 is the one clear exception in the representative subset (Table O.4).

Model	\$100K	\$300K	\$500K	Spread
Qwen3-4B	96.9%	100%	100%	+3.1pp
Qwen3-4B-Instruct-2507	93.8%	100%	93.8%	+6.2pp
Claude Opus 4.7	81.2%	100%	100%	+18.8pp
GPT-5.4	62.5%	100%	100%	+37.5pp

Table O.4: No-horizon %LT stratified by long-term reward size. GPT-5.4 is the only model in the subset with strong reward sensitivity (+37.5pp from \$100K to \$300K), consistent with its high coherence score (Figure O.4); the Qwen3 models are at ceiling regardless of reward.

O.4 Cross-Cutting Patterns

This section collects patterns that don't fit neatly into stability, coherence, or latent preference: the raw per-horizon curve, the Claude step function, the Qwen3 hybrid vs. mode-specialized comparison, and the target-model deep dive.

Raw per-horizon %LT curve. Figure O.9 plots %LT vs. time horizon for all 30 models, grouped into per-family panels. This is the raw preference shape; the shaded red band marks the 1-5y reasoning zone where coherence is defined.

The Claude step function. Figure O.10 isolates the Claude family's characteristic pattern: 0% LT at every horizon under 10 years, then a hard step to ~99% at 10 years. This is maximally coherent in the reasoning zone (by heuristic, not reasoning), but collapses to order bias at 20-50y for the smaller Claude variants.

Per-Horizon %LT: Does the Model Track the Stated Deadline? (raw preference curve)

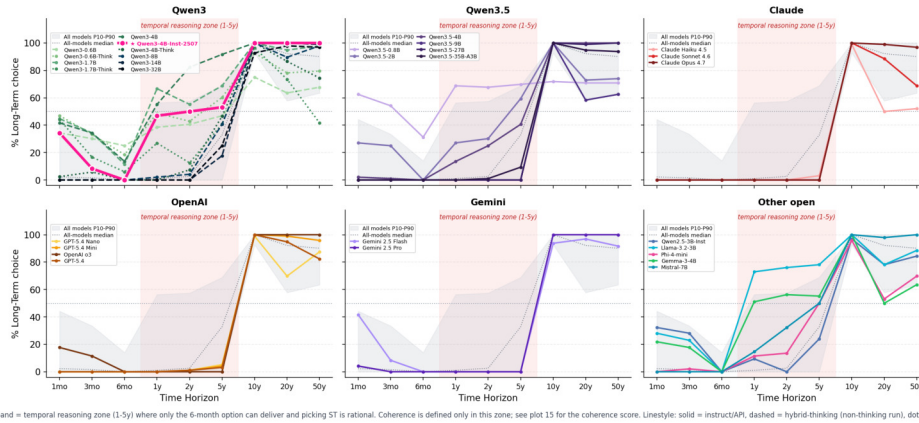


Figure O.9: %LT by time horizon across all 30 models, in per-family small multiples. The all-model P10–P90 envelope (gray band) and median (dotted) are shown for context. Within the temporal reasoning zone (1–5y, shaded red), the rational %LT target is 0; at horizons of 10y and beyond, the rational target is 100. The target model Qwen3–4B–Instruct–2507 (starred) sits near 50% in the reasoning zone, an average of two near-deterministic order-polarized sub-behaviors (O.4.1).

Claude Family: Flat-Zero %LT Until the 10y Threshold, Then Step to 100%

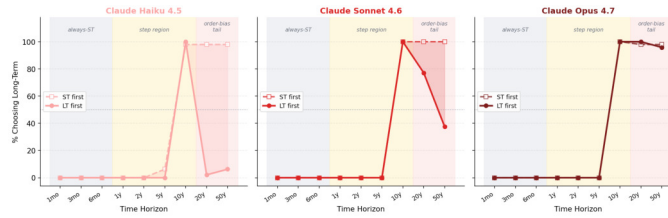


Figure O.10: Claude family step function: flat 0–3% LT for all horizons under 10 years, step to 99% at 10 years. A binary cutoff rule (“under 10 years = short-term”) explains the pattern; the model is not reasoning about deliverability, it is threshold-matching.

Qwen3 hybrid-thinking vs. mode-specialized 2507 variants. Figure O.11 compares the hybrid-thinking Qwen3–{0.6B, 1.7B, 4B} checkpoints against their thinking-only and non-thinking-only 2507 refreshes. The hybrid-thinking and thinking-only variants preserve graded temporal sensitivity (informative but instrumentally incoherent); the distilled non-thinking-only variants collapse into three discrete modes with order bias in the reasoning zone.

Per-horizon %LT (full subset breakdown). Table O.5 gives the full per-horizon breakdown for the four-regime subset.

Post-Training Recipe Effect: Same Size, Different Training



Figure O.11: Within-family mode comparison across three Qwen3 sizes (0.6B, 1.7B, 4B). Columns: hybrid Qwen3-* run in non-thinking mode, the same hybrid run in thinking mode, and the non-thinking-only Qwen3*-Instruct-2507 specialist (target variant at 4B starred). Each panel overlays %LT under ST-first (dashed) and LT-first (solid) orderings with the gap shaded. Mode specialization into non-thinking replaces the hybrid checkpoint’s graded horizon curve with a three-mode lookup pattern and a large order gap in the reasoning zone.

Horizon	Zone	Qwen3-4B	Qwen3-4B-Inst	Claude Opus 4.7	GPT-5.4
1 mo	Before ST anchor	42%	34%	0%	0%
3 mo	Before ST anchor	34%	8%	0%	0%
6 mo	Exact match (ST)	14%	0%	0%	0%
1 y	Reasoning zone	55%	47%	0%	0%
2 y	Reasoning zone	82%	50%	0%	1%
5 y	Reasoning zone	92%	53%	0%	3%
10 y	Exact match (LT)	100%	100%	100%	100%
20 y	Beyond LT anchor	100%	100%	99%	95%
50 y	Beyond LT anchor	100%	100%	97%	82%

Table O.5: %LT by horizon and model for the four-regime subset. In the reasoning zone (1–5y), only the 6-month ST option can deliver, so a coherent agent picks ST (0% LT). Claude Opus 4.7 achieves this (and GPT-5.4 nearly does: 0–3%) but via different mechanisms. Qwen3-4B has the smoothest horizon-sensitivity curve yet is instrumentally incoherent (82% LT at 2y). Qwen3-4B-Instruct-2507’s flat 47–53% in this zone is an order-bias artifact (see Table O.2). Beyond 10y, the smaller Claude variants and GPT-5.4 erode toward order bias (see Section O.1); the Qwen3 models stay saturated.

O.4.1 Qwen3-4B-Instruct-2507 deep dive

The cross-model panels establish the population pattern. We now zoom into the primary model. Three views decompose its 960 prompts along stimulus axes the tables aggregate over.

Horizon × context. Figure O.12 is a single-model %LT heatmap over (horizon × context). Each cell pools 12 prompts (3 rewards × 2 label styles × 2 orders).

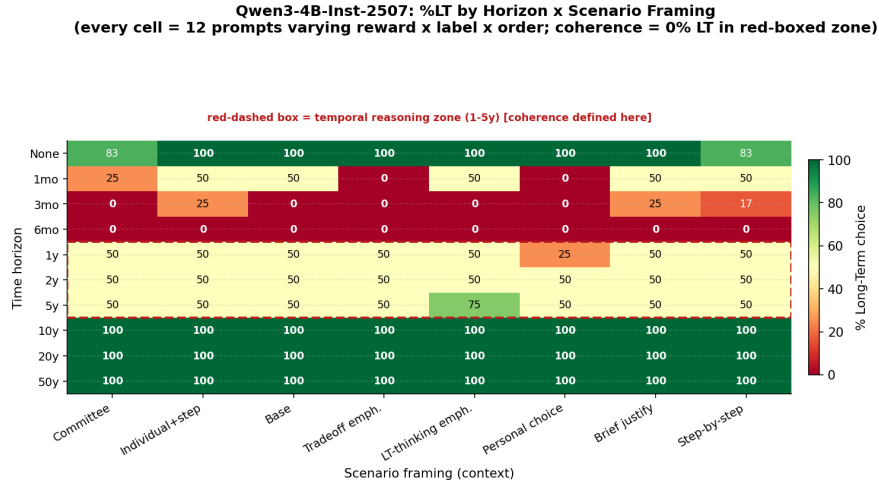


Figure O.12: Qwen3-4B-Instruct-2507: %LT by horizon and scenario framing. The anchor horizons (6mo, 10y) are context-insensitive and near-correct; the temporal reasoning zone (1–5y, dashed red box) is where framing has leverage. Within that zone, different framings push the model toward opposite choices, confirming that the pooled ~50% %LT is an average over meaningfully different sub-behaviors, not a stable 50/50 uncertainty.

Horizon × reward × order. Figure O.13 splits the same data by presentation order and shows the order-bias delta per (horizon, reward) cell.

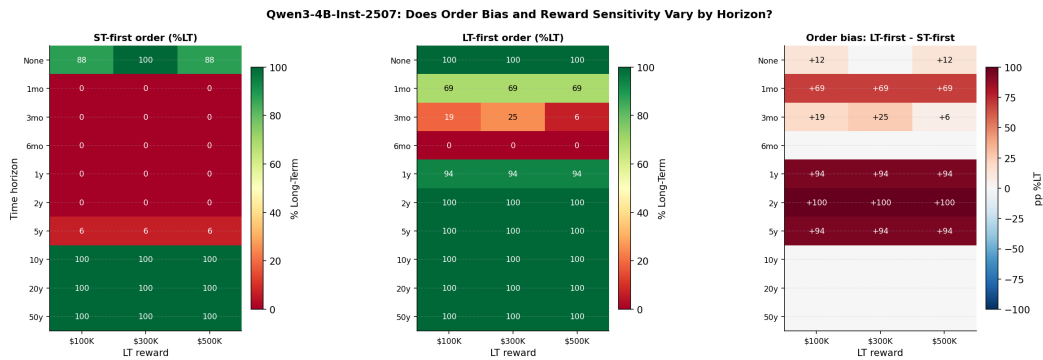


Figure O.13: Qwen3-4B-Instruct-2507: %LT under ST-first (left) and LT-first (middle) presentation orders, and the signed order-bias delta (right). Order bias is concentrated in the reasoning zone and is nearly reward-invariant within that zone: flipping the order changes %LT by up to ±100pp regardless of whether the long-term reward is \$100K or \$500K. The anchor horizons and the no-horizon condition show near-zero order bias.

Where does the variation come from? Figure O.14 stratifies the horizon curve by each stimulus dimension, holding the pooled curve fixed as reference.

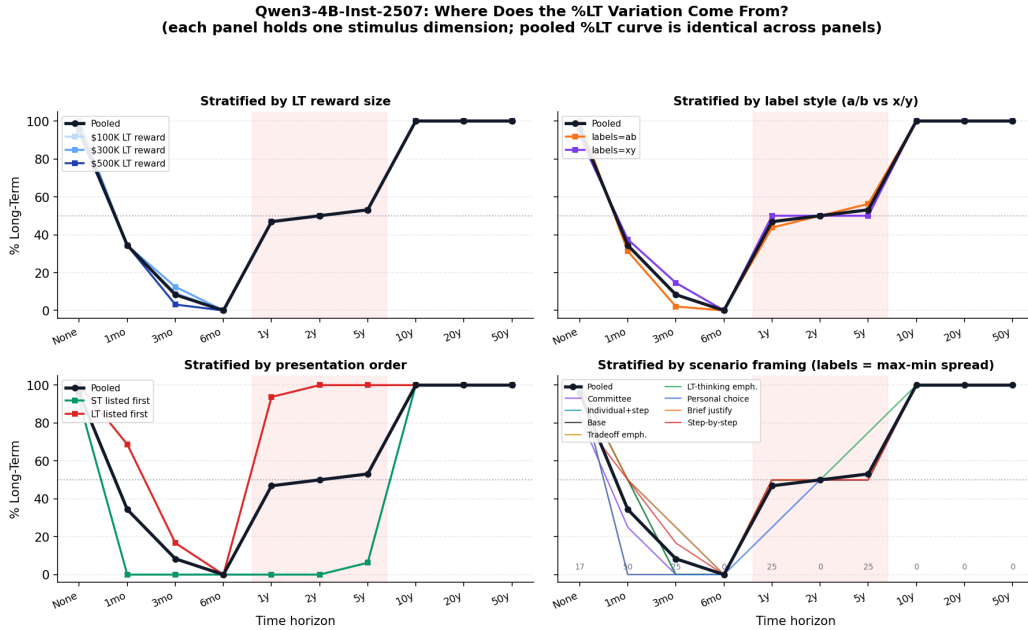


Figure O.14: Qwen3-4B-Instruct-2507: per-horizon %LT stratified by reward size (top-left), label style (top-right), presentation order (bottom-left), and scenario framing (bottom-right). The pooled curve (black) is identical across panels. Order stratification shows the dominant effect: the two order-conditioned curves sit at opposite extremes in the reasoning zone. Reward size and label format have negligible effect; context framing contributes moderate additional spread but is smaller than order.

Within the reasoning zone, Qwen3-4B-Instruct-2507 is not uncertain, it is positionally polarized. Almost all of the instability reported in the pooled tables is driven by presentation order, with a secondary contribution from context framing. Reward magnitude and label format are effectively inert.

O.5 Key Findings

- Coherence lives in the 1–5y zone, not everywhere.** Agreement with the rational rule at the anchors (6mo, 10y) and beyond is pattern-matching or EV-dominance, not reasoning. The 1–5y zone is the only regime where the rational choice (pick ST) can be distinguished from a model just following the nearest anchor.
- Most models fail the coherence test.** Only the large frontier API models (Claude Opus 4.7, Gemini 2.5 Pro, Claude Sonnet 4.6, GPT-5.4, o3, GPT-5.4 Mini, the Claude family more broadly) reach 95–100% coherence in 1–5y. Most open-weight models at 4B-class and below sit at <50%.
- The Claude family is coherent by heuristic, not reasoning.** Its 100% coherence in 1–5y is achieved by a binary cutoff (“under 10 years ⇒ ST”); the smaller Claude Haiku 4.5 and Claude Sonnet 4.6 variants collapse to order bias at the longer horizons where this cutoff no longer applies (Figure O.1), while Claude Opus 4.7 remains order-stable. The heuristic is functionally coherent for the 1–5y test but does not generalize.
- Our target Qwen3-4B-Instruct-2507 operates in three discrete modes.** At horizons under 6 months: coherent (picks ST, order-stable). In the 1–5y reasoning zone: pure positional polarization (0–6% order stability), averaging to ~50% %LT. At 10+ years: coherent (picks LT, order-stable). Mode specialization into non-thinking appears to have replaced graded horizon sensitivity with a lookup pattern.
- The Qwen3-4B hybrid-thinking checkpoint is graded but wrong.** It has the smoothest horizon sensitivity curve (34% LT at 3 months rising to 92% at 5 years),

consistent with continuous temporal representations in the geometry analysis (Appendix L), but is instrumentally incoherent in the reasoning zone (82% LT at 2 years, where LT cannot deliver).

6. **Reward magnitude is largely inert; context framing is not.** Only GPT-5.4 in the representative subset shows strong reward sensitivity (+37.5pp from \$100K to \$300K). Context framing produces comparable or larger shifts for several models.
7. **Connection to the mechanistic story.** The geometry analysis shows the model encodes continuous temporal representations internally but collapses them into binary preference at the turn boundary (Appendix L). The behavioral results show the same pattern at the output level: nuanced temporal sensitivity does not survive to coherent decision-making. This motivates the steering experiments in Part 3: if the internal representation is richer than the behavior, targeted intervention may recover the lost gradation.

Appendix P Cross-model patching comparison

We repeat the 160-pair residual-stream activation patching (Appendix J) on nine Qwen3 variants spanning 0.6B–14B parameters, including our primary target `Qwen3-4B-Instruct-2507` and its hybrid-thinking sibling `Qwen3-4B`. The question: is the temporal-preference subgraph localized at a consistent *fractional depth* across model scales, or does it shift with parameter count?

Protocol. For each model, we collect clean/corrupted activations on the same contrastive prompt bank and measure *recovery* (the fraction of the clean logit difference restored by patching a single component) at each layer, for three hooks: `resid_post`, `attn_out`, `mlp_out`. We plot mean recovery vs. *fractional depth* (layer/total layers) to align curves across models of different depths.

Findings. Three patterns hold across the family (Figures P.1–P.5).

- **Residual stream saturates.** `resid_post` recovery is a clean sigmoid that crosses 50% around depth 0.65–0.70 and saturates at 1.0 by depth 0.8 in every model (Figure P.3). The location of the transition is nearly scale-invariant in depth units.
- **Attention localizes at ~ 0.6 – 0.7 depth, but its recovery shrinks with scale.** `attn_out` peaks in a narrow band at depth 0.6–0.7 in all models, but peak recovery drops from ~ 0.86 – 0.92 in the smallest models (0.6–1.7B) to ~ 0.18 – 0.30 in the 4B–14B variants (Figure P.4). The circuit becomes more distributed, not absent, at scale.
- **MLP contribution is diffuse.** `mlp_out` recovery stays below 0.4 for every model and is spread across mid-to-late layers without a sharp peak (Figure P.5). The MLPs accumulate the preference rather than route it.

Peak layers and recoveries per model are tabulated in `summary.txt`; our primary target `Qwen3-4B-Instruct-2507` tracks the hybrid-thinking 4B checkpoint (`Qwen3-4B`) closely on all three hooks.

Coarse Patching Across Investment Models
Baseline: Investment (I)

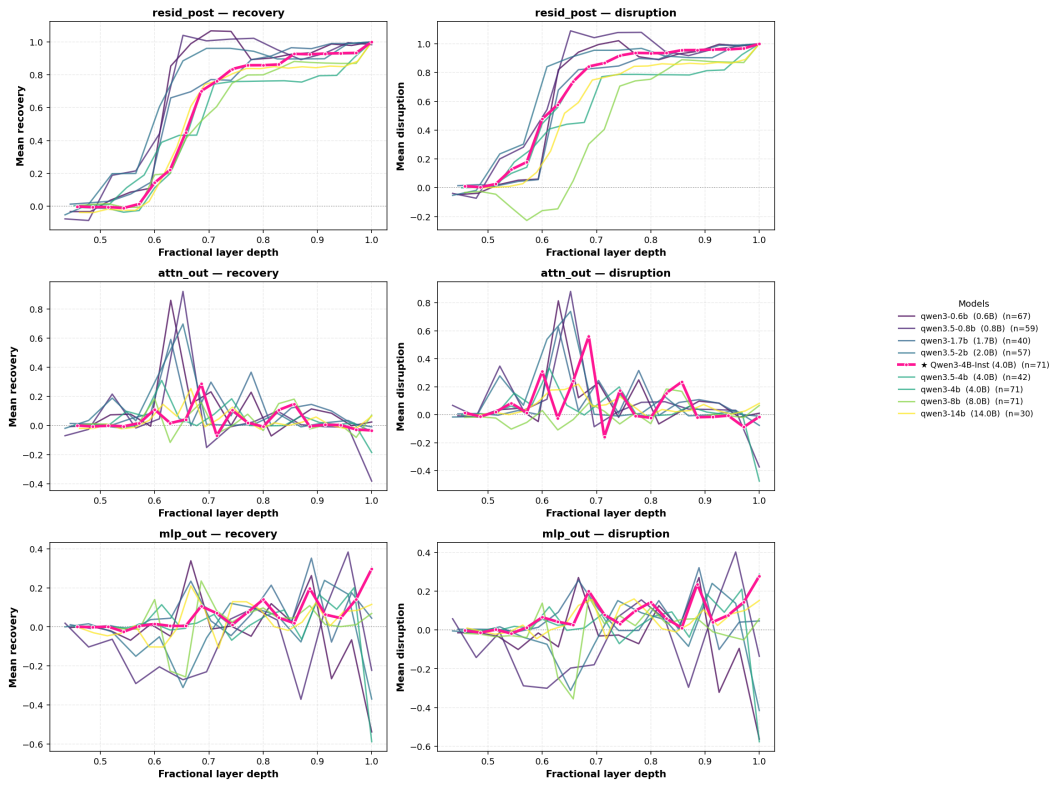


Figure P.1: Cross-model patching overview at fractional depth. Rows: `resid_post` (top), `attn_out` (middle), `mlp_out` (bottom). Columns: recovery (left), disruption (right). Nine Qwen3 variants overlaid (0.6B–14B). The residual stream saturates uniformly; attention localizes but weakens with scale; MLP stays diffuse.

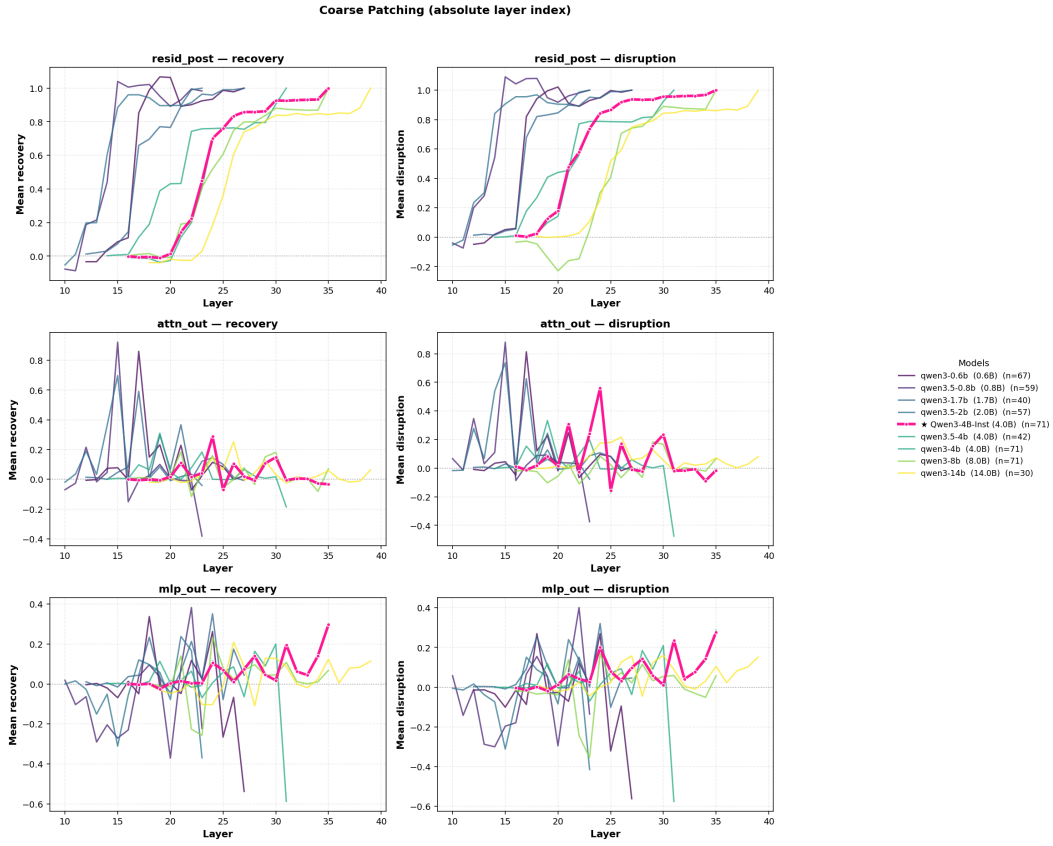


Figure P.2: Same comparison on absolute layer indices rather than fractional depth. Without depth normalization the curves spread across layers 15–35 without aligning, confirming that fractional depth (not absolute index) is what stabilizes circuit location across scales.

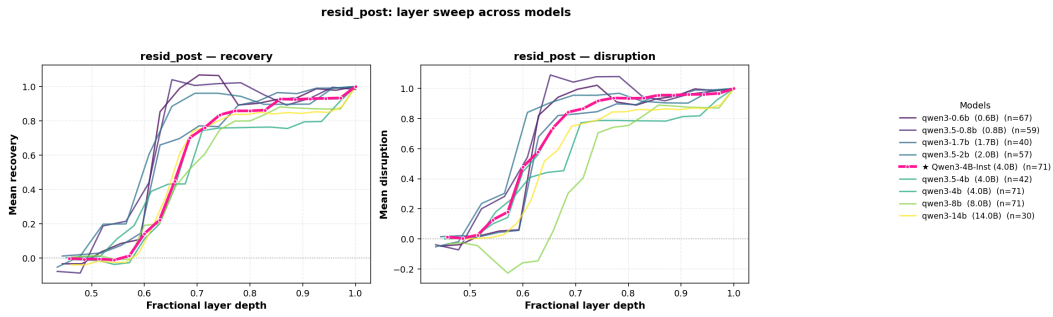


Figure P.3: resid_post recovery and disruption vs. fractional depth. Every model follows the same sigmoid, saturating at ~ 1.0 by depth 0.8.

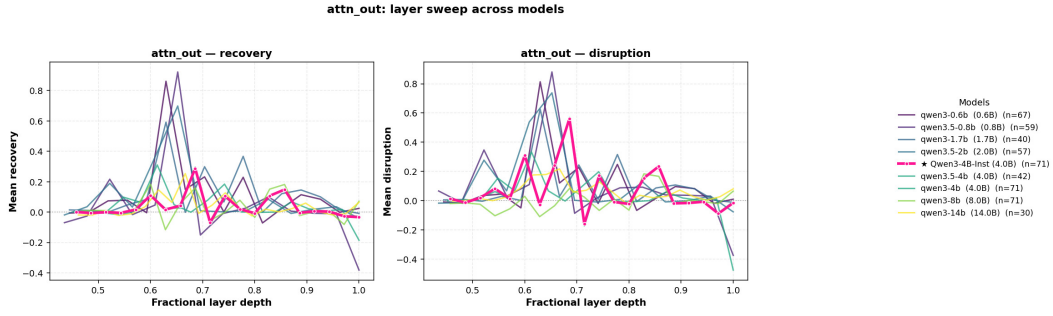


Figure P.4: `attn_out` recovery and disruption vs. fractional depth. The peak is narrow and consistent near depth 0.6–0.7, but its height shrinks monotonically with parameter count, from ~ 0.9 (0.6B) to ~ 0.2 – 0.3 (8–14B).

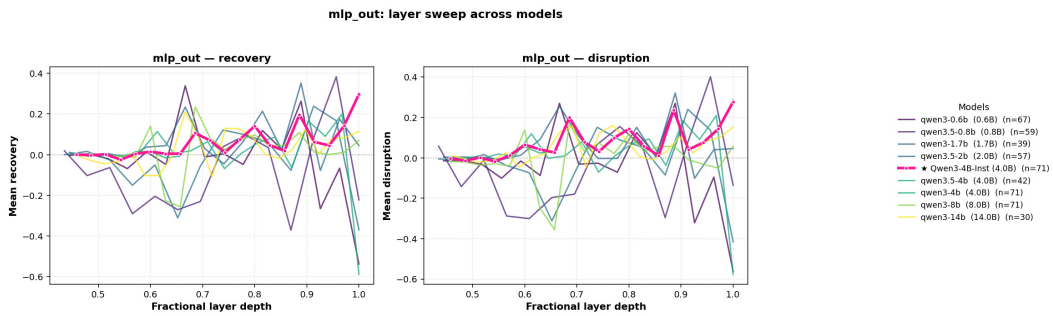


Figure P.5: `mlp_out` recovery and disruption vs. fractional depth. Effects are low (≤ 0.4) and broadly distributed across mid-to-late layers for all models, with no sharp localization.

Appendix Q Error monitoring in the temporal preference subgraph

The localization results (Appendices Appendix H–Appendix K) converge on a subgraph at layers 17–35; the geometry results (Appendix L) show that time horizon is encoded as a non-linear manifold within it; and the behavioral results (Appendix O) reveal that this rich internal structure does not survive to coherent decision-making. A natural question is whether this gap between representation and behavior is specific to temporal reasoning or reflects a broader property of the subgraph region. We test this by probing whether the same layers and token positions also encode a second meta-cognitive variable, the accumulated reliability of a multi-step reasoning chain, and whether the two variables share or compete for representational capacity.

Shared pipeline. All 4,650 samples (3,550 error hops + 1,100 temporal preference samples from D_{explicit} and D_{implicit} ; Appendix E) are extracted through a *single model load* of **Qwen3-4B-Instruct-2507** using the Qwen chat template. Raw hidden states at 15 key layers are stored for all samples and jointly projected into a shared PCA-50 subspace fit on the full concatenation, ensuring that error and temporal representations inhabit the same coordinate system. Probes use logistic regression ($C = 0.01$, balanced class weights), 10-fold cross-validation, and 500-permutation null distributions. We note that probing establishes *correlational* decodability, complementary to but distinct from the causal localization in Appendices Appendix I and Appendix J; a feature being decodable at a layer does not entail that the layer is causally necessary for behavior.

Error injection dataset. We construct 1,250 contrastive multi-hop math reasoning chains (2–4 hops) with three conditions: *clean*, *error_at_1*, and *error_at_2*, using five error types (off-by-one, wrong operator, wrong unit, magnitude error, wrong percentage base). Each hop is wrapped in the Qwen chat template as a user-to-assistant turn pair, matching the format used throughout the main paper.

Q.1 Does error state co-localize with temporal preference?

Before asking whether error and temporal preference *interact*, we check whether they occupy the same architectural region. A positive answer would suggest the subgraph functions as a general meta-cognitive module rather than a temporal-specific circuit.

Layer-wise error probes. Figure Q.1 reports probe accuracy for three error targets across the 15 sampled layers.

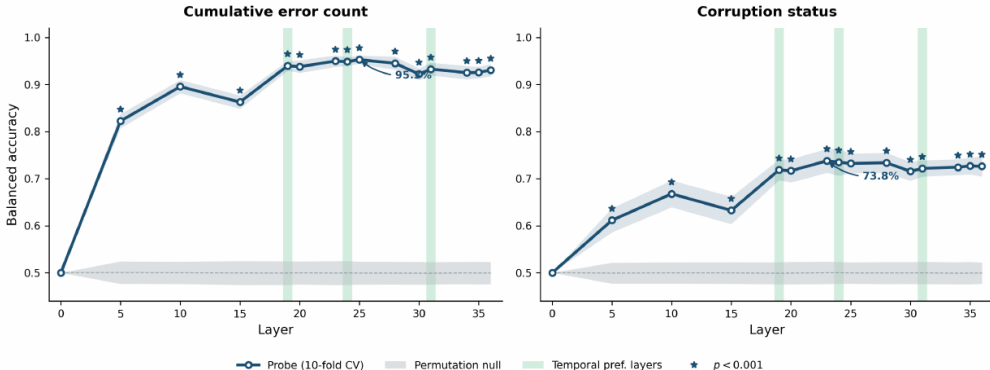


Figure Q.1: Error decodability in shared PCA-50 space ($n = 3,550$, 10-fold CV, 500-permutation null). Green bands: temporal preference subgraph layers (19, 24, 31). **Left:** Cumulative error count reaches a plateau of 94–95% across layers 19–31, peaking at 95.3% between layers 24 and 25. **Right:** Local corruption status (is this specific step injected?) peaks at 73.8% at layer 23. Stars: $p < 0.001$.

Layer	Cumulative errors	Corruption status	Propagation
0	50.0%	50.0%	50.0%
5	82.3%	59.5%	82.3%
10	89.7%	60.9%	89.7%
15	86.1%	63.4%	86.1%
19	94.5%	71.3%	94.5%
24	95.0%	73.5%	95.0%
25	95.3%	72.5%	95.3%
31	93.0%	72.6%	93.0%
36	91.7%	71.5%	91.7%

Table Q.1: Probe accuracy at selected layers (all $p < 0.001$ except layer 0). Cumulative error count and propagation status are numerically identical, confirming the probe reads a chain-level property. Corruption status is 21pp lower, indicating the model encodes “my chain is degraded” far more reliably than “the error is at this step.”

The cumulative error plateau (94–95%) spans layers 19–31, precisely the subgraph identified by attribution patching in the main paper. The 21-point gap between chain-level error (95%) and local error identity (74%) parallels the main paper’s finding that global context properties (time horizon) are more structured than local behavioral outputs (specific choices in the reasoning zone; Appendix O).

Error at the turn-transition tokens. The geometry analysis (Appendix L) identifies the $\langle |im_end| \rangle$ to **assistant** turn transition as the locus where temporal preference geometry becomes linearly separable (Figure L.5). Figure Q.2 tests whether error state follows the same pattern.

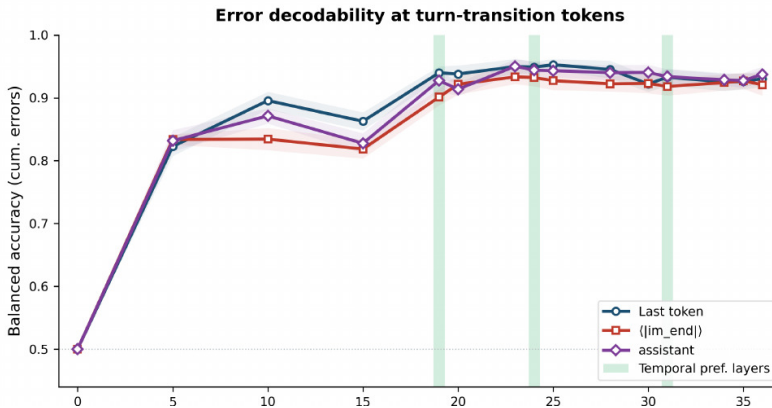


Figure Q.2: Cumulative error decodability at three token positions. All converge to $>93\%$ by layer 19. The turn-transition tokens ($\langle |im_end| \rangle$ and **assistant**), where Appendix L shows temporal preference geometry crystallizing, carry error state with comparable fidelity to the last token.

Error decodability at the turn-transition tokens matches the last token from layer 19 onward, with the **assistant** token slightly outperforming the last token at layers 24–31 ($\sim 95\%$ vs. $\sim 93\%$). This convergence suggests that the turn-transition computation, the same computation that transforms off-policy context into on-policy generation for temporal preference, also integrates reasoning reliability before generation begins.

Q.2 Do error and temporal preference share representational structure?

Co-localization does not entail shared structure: two variables can occupy the same layers in orthogonal subspaces. We test this directly by training probes for each variable in the shared PCA-50 space and comparing the resulting weight vectors.

Cross-probing protocol. For each of the 15 key layers, we train a binary error probe (cumulative errors > 0 vs. $= 0$; $n = 3,550$) and a binary temporal probe (immediate vs. long-term; $n = 1,100$), both in the shared PCA-50 space. We compute cosine similarity between the normalized weight vectors and test significance with a 500-permutation null (shuffle temporal labels, refit, recompute cosine). We also measure cross-domain transfer: apply the error probe to temporal data (and vice versa) and test against 200-permutation nulls on the target labels.

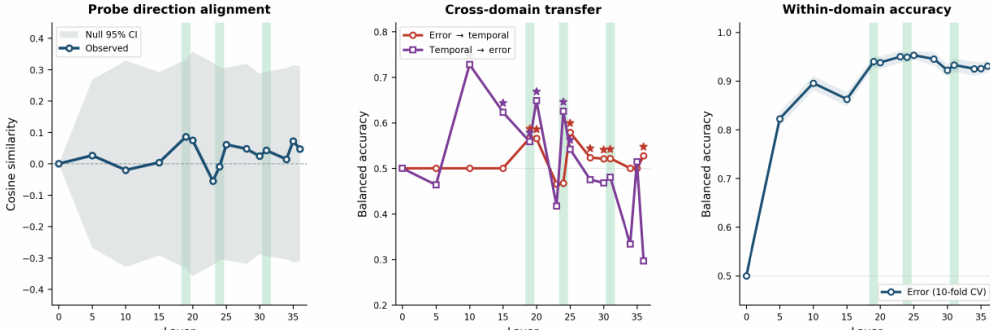


Figure Q.3: Cross-probing in shared PCA-50 space ($n_{\text{err}} = 3,550$, $n_{\text{temp}} = 1,100$, 500-permutation nulls). **Left:** Cosine between probe directions, all values fall within the permutation null (shaded), no $p < 0.05$. **Center:** Cross-domain transfer, temporal-to-error is significant at layers 10–25 (purple stars, $p < 0.001$); error-to-temporal is weaker and sporadic (red stars). **Right:** Within-domain error accuracy for reference.

Layer	Cosine	Cosine p	Error→Temp	Temp→Error
10	+0.040	0.774	0.500	0.728***
19	+0.086	0.614	0.500	0.593***
20	+0.071	0.614	0.574***	0.648***
24	−0.023	0.876	0.500	0.626***
25	+0.046	0.720	0.500	0.574***
31	−0.052	0.712	0.500	0.500

Table Q.2: Cross-probe results at selected layers. No cosine reaches significance. Temporal-to-error transfer peaks at layer 10 (0.728) and remains above chance through layer 25; error-to-temporal transfer is at chance at most layers. *** indicates $p < 0.001$ against the permutation null on target labels.

Interpretation: orthogonal directions, partial non-linear overlap. The cosine null result (all $p > 0.5$) establishes that the linear separating hyperplanes for error and temporal preference are perpendicular in the shared activation space. However, the temporal-to-error transfer above chance at layers 10–25 ($p < 0.001$) shows that the temporal probe’s projection of the data partially predicts error status even though the two probe *directions* are orthogonal. This combination, perpendicular hyperplanes paired with above-chance transfer, indicates that the two variables share a *non-linear* subspace: their representations overlap on the activation manifold but not along any single linear axis.

This is consistent with the non-linear time-horizon geometry documented in Appendix L: if both error state and temporal preference occupy curved manifolds in the same region of activation space, their optimal linear separating hyperplanes can be orthogonal even as the manifolds themselves intersect. The asymmetry of the transfer (temporal-to-error stronger than error-to-temporal) suggests that the temporal preference representation, which captures broad context evaluation (“strategic vs. tactical” orientation; Appendix E), carries some error-relevant information as a byproduct, while the error direction (a narrower signal about chain corruption) does not carry temporal scope information.

Q.3 Key findings

1. **Error state co-localizes with temporal preference.** Cumulative error count is decodable at 95.3% ($p < 0.001$) with a plateau spanning layers 19–31, the same region identified by attribution patching. Error is decodable at the turn-transition tokens where temporal preference geometry crystallizes (Appendix L). This suggests the subgraph functions as a general meta-cognitive region, not a temporal-specific circuit.
2. **Chain-level error is far more decodable than local error identity.** The 21pp gap (95% cumulative vs. 74% corruption status) mirrors the main paper’s finding that global properties (time horizon) are more structured than local behavioral outputs.
3. **Error and temporal preference occupy orthogonal linear directions.** No cosine between probe weight vectors reaches $p < 0.05$ at any layer. The two variables do not compete for the same linear subspace within the subgraph.
4. **Asymmetric non-linear overlap exists.** The temporal probe transfers to the error task above chance at layers 10–25 ($p < 0.001$), but the error probe does not transfer to temporal preference. The two variables share curved manifold structure but not a linear direction, consistent with the non-linear geometry in Appendix L.
5. **The gap between representation and behavior generalizes.** Error state is internally encoded at 95% accuracy but barely affects output confidence, paralleling the temporal preference gap between representation and behavior documented in Appendices Appendix N and Appendix O. The gap is architectural, not task-specific.
6. **Two-axis steering is feasible.** The orthogonality of probe directions means a temporal preference steering vector (Appendix R) should not perturb error sensitivity. An error-awareness vector at layers 24–25 could complement temporal steering, enabling two-axis control with minimal cross-interference.
7. **Connection to the steering results.** The probing–steering dissociation observed in Appendix R (best probing at L26 vs. best steering at L19–22) may extend to error: the layers where error is most decodable (L24–25) need not be the layers where error-state interventions are most effective. Testing this prediction via error-state CAA is a natural next step.

Limitations. These results establish correlational decodability, not causal necessity. Activation patching of error-state representations (analogous to Appendices Appendix I and Appendix J) would be needed to confirm that the identified representations causally drive downstream behavior. Our error injection uses synthetic perturbations in math reasoning chains, which may not fully reflect the distribution of errors arising during unconstrained generation. Generalization to other reasoning domains and to models beyond Qwen3-4B-Instruct-2507 remains to be tested.

Part 3:

Could we control temporal preference?

- **R.** Contrastive CAA steering

Appendix R Contrastive steering results

Parts 1 and 2 established where temporal preference lives (layers 17–35; Appendix K) and what it looks like (an ordinal horizon that transforms into a binary preference at the turn boundary; Appendix L). The behavioral analysis showed that the resulting preferences are unstable and inconsistent (Appendix N, Appendix O). Here we ask the intervention question: can we *control* temporal preference by directly modifying the representations we identified?

We construct a CAA steering vector from the probe direction at layer 26 (Appendix G) and inject it at candidate layers during inference (methodology in Appendix AA).

R.1 Forced-Choice Behavioral Sweep

R.1.1 Layer \times Alpha Sweep

The probe’s best layer (26) is not necessarily the best steering layer. We swept 9 layers (19–27) \times 5 alpha values (1, 2, 5, 10, 20) = 45 configurations.

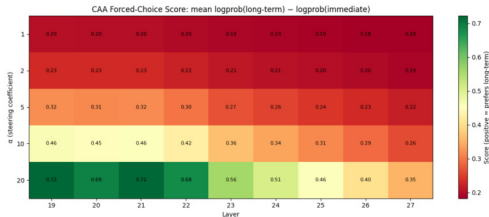


Table R.1: Forced-choice score $S(\alpha, l)$ across layers and steering coefficients. Baseline (no steering): $S = 0.1724$. Layers 19–22 form the behavioral sweet spot, with a sharp drop at layer 23.

Figure R.1: Heatmap of forced-choice score $S(\alpha, l)$ across layers (19–27) and steering coefficients ($\alpha = 1-20$). Layers 19–22 form the behavioral sweet spot, with effectiveness dropping sharply at layer 23.

α	L19	L20	L21	L22	L23	L24	L25	L26	L27
1	0.20	0.20	0.20	0.20	0.19	0.19	0.19	0.18	0.18
2	0.23	0.23	0.23	0.22	0.21	0.21	0.20	0.20	0.19
5	0.32	0.31	0.32	0.30	0.27	0.26	0.24	0.23	0.22
10	0.46	0.45	0.46	0.42	0.36	0.34	0.31	0.29	0.26
20	0.72	0.69	0.72	0.68	0.56	0.51	0.46	0.40	0.35

Probing–steering dissociation. Layer 26 is optimal for *reading* temporal orientation (99.2% probe accuracy) but not for *writing* it. Layers 19–22 are the effective steering layers, 4–7 layers earlier than the best probe layer. This dissociation is consistent with a functional asymmetry: upper layers consolidate a high-fidelity *readout* of the temporal concept, while mid-network layers are where causal interventions most effectively redirect the model’s downstream computation. Similar probing-vs-intervention gaps have been observed in other domains [43].

R.1.2 Extended Alpha Sweep

Following the initial sweep, we extended the alpha range for the most promising layers (19–25) with $\alpha \in \{20, 30, 40, 50\}$.

The score increases monotonically with α across all layers, with layers 19–22 consistently outperforming later layers. The optimal configuration (layer 22, $\alpha = 50$) achieves a score of 1.3944, representing a lift of +1.22 over the unsteered baseline of 0.1724.

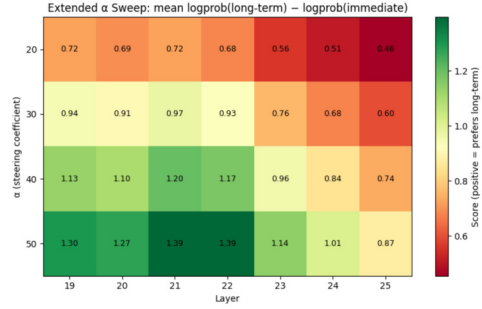


Table R.2: Extended alpha sweep. Best configuration: layer 22 with $\alpha = 50$, achieving $S = 1.3944$ (baseline: 0.1724, lift: +1.22). This corresponds to approximately $3.4\times$ more probability mass on long-term completions ($\exp(1.22) \approx 3.39$).

Figure R.2: Extended alpha sweep heatmap for layers 19–25 with $\alpha \in \{20, 30, 40, 50\}$. Score increases monotonically with α ; best configuration is layer 22 at $\alpha=50$.

α	L19	L20	L21	L22	L23	L24	L25
20	0.72	0.69	0.72	0.68	0.56	0.51	0.46
30	0.94	0.91	0.97	0.93	0.76	0.68	0.60
40	1.13	1.10	1.20	1.17	0.96	0.84	0.74
50	1.30	1.27	1.39	1.39	1.14	1.01	0.87

R.2 Open-Ended Generation Evaluation

The forced-choice metric measures whether the model’s token probabilities shift in the correct direction. To verify that this translates into qualitative behavioral change, we evaluate steering on 13 open-ended neutral prompts (e.g., “*You are advising a team on how to handle a major organizational challenge. What should be the main focus?*”).

R.2.1 Experimental Setup

For each configuration, we generate text with `do_sample=False` and `max_new_tokens=90`. We test layers $\{19, 20, 21, 22, 23, 26\}$ across $\alpha \in \{25, 40, 50\}$, applying the steering vector in both directions: positive α (toward long-term) and negative α (toward short-term).

All generated responses were scored by Claude Sonnet 4.6 on a $[-10, +10]$ temporal orientation scale, where -10 denotes clearly short-term thinking and $+10$ denotes clearly long-term thinking. We note that this LLM-as-judge evaluation is not validated against human ratings; the scores should be interpreted as a proxy for directional shift rather than a calibrated measure of temporal orientation.

R.2.2 Results

Configuration	$\alpha = 25$	$\alpha = 40$	$\alpha = 50$
L19 → long-term	+1.1	+1.2	+2.3
L19 → short-term	-1.7	-4.7	-5.7
L20 → long-term	+1.8	+2.0	+2.4
L20 → short-term	-1.9	-3.6	-3.4
L21 → long-term	+1.8	+2.2	+2.7
L21 → short-term	-2.0	-3.5	-2.9
L22 → long-term	+1.2	+2.4	+2.2
L22 → short-term	-1.2	-2.8	-4.0
L23 → long-term	+1.2	+1.6	+1.7
L23 → short-term	-0.8	-1.0	-2.4
L26 → long-term	+1.2	+1.0	+1.0
L26 → short-term	-0.5	-1.0	-0.1

Table R.3: Mean shift from baseline on the $[-10, +10]$ temporal orientation scale for open-ended generation. Positive values indicate a shift toward long-term framing; negative values indicate a shift toward short-term framing. Each entry averages over 13 prompts.



Figure R.3: Shift from baseline on the $[-10, +10]$ temporal orientation scale for open-ended generation across layers and α values. Both long-term (positive) and short-term (negative) steering produce consistent directional shifts.

Key observations.

1. **Bidirectional steering.** Both positive and negative α produce consistent shifts in the expected direction, confirming that the CAA vector captures a genuine temporal orientation axis.
2. **Qualitative framing shifts.** At $\alpha \in [40, 50]$, long-term steered outputs adopt strategic framing (e.g., “resilient,” “future-ready,” “systemic redesign”), while short-term steered outputs adopt triage framing (e.g., “remain calm,” “prioritize urgency,” “structured immediate steps”).
3. **Coherence boundary.** Pilot runs at $|\alpha| = 60$ showed output incoherence, so we capped the sweep at $|\alpha| = 50$; beyond that the intervention appears to push the model too far from its natural distribution.
4. **Asymmetry.** The long-term direction produces cleaner shifts than the short-term direction at the same $|\alpha|$, suggesting that the model’s latent distribution may be slightly closer to the long-term end of the temporal axis.
5. **Layer effects mirror forced-choice results.** Layers 19–22 produce the strongest open-ended shifts, and layer 26 produces only weak effects despite having the highest probe accuracy. This reinforces the probing–steering dissociation documented in Section R.1.

R.2.3 Qualitative Example

We illustrate the steering effect with a representative example.

Prompt: *You are advising a team on how to handle a major organizational challenge. What should be the main focus?*

Baseline (score: 2): The response focuses on present-tense communication and immediate trust-building without strong temporal language in either direction. The main focus is on clear communication, transparency, and employee engagement.

Long-term steered ($\alpha = +50$, score: 6): The response centers on long-term success through resilience and shared vision, framing organizational challenges in terms of sustained adaptive capacity. The main focus is on resilience through shared purpose, adaptive thinking, and inclusive leadership.

Short-term steered ($\alpha = -50$, score: -3): The response leads with immediate and clear communication and emphasizes knowing what the deadline is, orienting team management around near-term operational urgency.

R.3 Discussion

Probing \neq steering. The central methodological finding is the dissociation between the optimal probing layer (26) and the optimal steering layers (19–22). This dissociation has implications for the broader interpretability literature: high probe accuracy at a layer does not imply that the same layer is the appropriate target for causal intervention. The readout of a concept and the point at which that concept can be effectively modified may be separated by several layers, reflecting distinct functional roles in the transformer’s computation [100].

Implicit vectors generalize. Using the implicit dataset (which contains no surface temporal vocabulary) to construct the CAA vector ensures that the steering direction captures semantic temporal reasoning rather than lexical artifacts. The cross-dataset probe generalization (Section G.4) confirms that the implicit direction aligns with the explicit temporal axis, and the forced-choice evaluation on explicit prompts (Section R.1) demonstrates that this vector effectively steers behavior on prompts with overt temporal cues.

Connection to subgraph localization. The behavioral sweet spot at layers 19–22 aligns with the mid-network components identified by the EAP-IG attribution analysis (Appendix H) and the activation patching experiments (Appendix I). This convergence across independent methodologies (probing, CAA steering, attribution patching, and activation patching) provides strong evidence that the temporal preference mechanism is localized to a consistent set of mid-to-upper layers.

Relation to the representational geometry. The PCA analysis (Section G.3) shows that the temporal direction in the implicit dataset is not captured by the top principal components. This is consistent with the non-linear manifold structure reported in Appendix L, where time horizon is encoded in a curved subspace. The CAA vector, derived from the linear probe direction, provides a first-order approximation to steering along this manifold. The monotonic increase in steering score with α (Table R.2) suggests that this linear approximation remains effective within the tested range, though the output-quality degradation at $|\alpha| = 60$ may indicate the intervention exceeding the locally linear regime.

LLM-as-Judge Evaluation Criteria. To quantify the qualitative shifts in our open-ended generation experiments, we used the Claude Sonnet 4.6 API as an independent evaluator. Each generated response was individually processed by the API and assigned a score on a $[-10, +10]$ scale, where -10 represents an extreme short-term focus and $+10$ represents an extreme long-term focus. The model was prompted to evaluate responses by strictly adhering to predefined grading criteria. Specifically, the evaluator analyzed the text for the presence and frequency of explicit temporal keywords (e.g., immediate triage versus systemic redesign)

and weighed semantic details, structural planning, and thematic biases that explicitly skewed the generation toward a specific temporal horizon.

Part 4:

Extended methodologies

- **S.** Notation
- **T.** Contrastive probing methods
- **U.** Attributional contrastive methods
- **V.** Causal parametric methods
- **W.** Causal contrastive methods
- **X.** Parametric geometry methods
- **Y.** Behavioral discounting methods
- **Z.** Behavioral coherence methods
- **AA.** Contrastive steering methods
- **AB.** Worked case study: highly-formatted pair

Appendix S Notation and key concepts

The following terms and abbreviations are used throughout the appendices.

Term	Definition
Subgraph	Model components (attention heads, MLP neurons) whose ablation or patching shifts temporal preference.
Temporal preference	The model’s tendency to favor short-term vs. long-term options in a forced-choice setting.
Time horizon	An explicit temporal constraint (e.g., “1 year”) given in the prompt; ranges from seconds to centuries.
On- vs. off-policy	<i>On-policy</i> : activations from the model’s own generation. <i>Off-policy</i> : activations read from a forced context (user turn).
Contrastive pair	Matched clean/corrupted prompts that differ in temporal framing; used for both EAP-IG and activation patching.
EAP-IG	Edge Attribution Patching with Integrated Gradients [42]; gradient-based attribution that approximates causal patching.
Activation patching	Replacing a component’s activations with counterfactual values to measure causal effect on a downstream metric.
Recovery / Disruption	Normalized $[0, 1]$ metrics for denoising and noising patching, respectively; 0 = no effect, 1 = full effect.
Probing layer	Residual-stream layer at which a linear classifier best separates short- vs. long-term orientation (layer 26 in this work).
Steering layer	Layer at which a CAA vector [100] most reliably shifts behavior (layers 19–22 in this work).
CAA	Contrastive Activation Addition: mean activation difference between long-term and short-term choices, used as a steering vector.
Decision boundary	Binary search over delayed-reward magnitudes to locate per-item indifference, used to fit hyperbolic discount rate k .
MCQ-27	Kirby Monetary Choice Questionnaire [58]: 27-item instrument for estimating temporal discount rates.

Appendix T Contrastive linear probing methodology

We train logistic regression probes [75, 57] on residual-stream activations to determine *where* the model linearly encodes the distinction between short-term and long-term temporal orientation. Probing results are presented in Appendix G.

T.1 Activation Extraction

For each prompt, we concatenate the question and the choice text, apply the Qwen3 chat template, and extract residual-stream activations at every layer. Because the chat template wraps the user turn as

```
<|im_start|>user\n{question + choice}<|im_end|>\n<|im_start|>assistant\n
```

the token at position -1 is the trailing newline after `assistant`, a fixed token that is identical across all prompts and carries only whatever signal attention has propagated into that generic position. We instead locate the last `<|im_end|>` token in the sequence (which closes the user turn) and extract at position `im_end - 1`, corresponding to the final token of the actual choice text. This position directly encodes the semantic content of the choice.

Impact of the token-position correction. The correction produced a qualitative change in both probe accuracy and downstream steering vector quality:

Metric	Before (trailing \n)	After (im_end - 1)
Extraction token	\n (trailing newline)	Last choice token
CAA vector ℓ_2 norm	2.62	30.30
Probe accuracy (best layer)	$\sim 93\%$	99.2%

Table T.1: Effect of correcting the extraction token position. The previous vector was essentially normalized noise; the corrected extraction yields a $\sim 10\times$ stronger CAA vector.

T.2 Probe Training Protocol

We train one `LogisticRegression(C=0.1)` probe per layer on D_{implicit} with the following methodological controls:

- Pair-level train/test split.** The split operates on pair indices rather than individual rows. Both the immediate and long-term activations from a given pair always land in the same fold. Without this, the probe can exploit shared question text as a shortcut, and the test set is not truly held out. We use an 80/20 split with a fixed random seed.
- StandardScaler normalization.** The residual stream has 2,560 dimensions with varying variances. `LogisticRegression` with ℓ_2 regularization penalizes large weights uniformly, so high-variance dimensions dominate the penalty without scaling. We fit a `StandardScaler` on the training fold and apply it to both train and test. Critically, the scaler is persisted to disk alongside each probe and re-applied during cross-dataset evaluation and when extracting the probe’s coefficient vector for use as the CAA steering direction.

Activations are stored as tensors of shape $[n_{\text{prompts}}, n_{\text{layers}}, d_{\text{model}}] = [600, 36, 2560]$ for the implicit dataset.

Appendix U Attributional contrastive methodology

We identify this subgraph in two stages: first, we restrict the candidate node set using Edge Attribution Patching with Integrated Gradients (EAP-IG); second, we score and prune edges between these nodes to recover a sparse functional subgraph. Unlike prior approaches that attribute to logit differences, we compute attribution with respect to individual option logits, yielding concept-specific attribution scores that disentangle the contributions of nodes to competing temporal evaluations.

U.1 Edge Attribution Patching-Integrated Gradients

To localize the internal computations associated with temporal preference, we use Edge Attribution Patching with Integrated Gradients (EAP-IG). EAP-IG operates on matched clean and corrupted prompts and assigns attribution scores to internal components based on their contribution to the model’s preference for one response token over another. It can be viewed as a computationally efficient approximation to activation patching.

EAP-IG can be implemented in two ways: by interpolating activations at each node, or by interpolating only the input embeddings. The latter is significantly more efficient and provides a practical method for estimating edge importance. However, as this approach compounds two approximations, we do not use it to precisely rank components; instead, we use it to restrict the search space by filtering out nodes with low attribution scores.

Concretely, we interpolate between corrupted and clean inputs in embedding space and integrate gradients along the resulting path, rather than relying on a single local gradient estimate. This yields attribution scores $s^A(x, i, t)$ for each component i at token position t for metric A on prompt x .

U.2 Notation

In the paper, the attribution score for a variant v is denoted as $s^A(x, i, t)$.

$$s^A(x, i, t) = (z_{i,t} - z'_{i,t}) \int_{\alpha=0}^1 \frac{\partial L_A(z' + \alpha(z - z'))}{\partial z_{i,t}} \approx (z_{i,t} - z'_{i,t}) \frac{1}{m} \sum_{k=1}^m \frac{\partial L_A(z' + \frac{k}{m}(z - z'))}{\partial z_{i,t}} \tag{U.1}$$

U.2.1 Metric Normalized Attribution Scores

The attribution scores are scaled by $\Delta L_A = L_A(z) - L_A(z')$ so they can be aggregated across datasets and semantically equivalent metric functions. In this paper, $\bar{s}^A(x, i, t)$ denotes the scaled attribution scores. When $L_A(z) \approx L_A(z')$, it indicates that the model either does not distinguish between the clean and corrupted cases or has nearly the same preference for both options; such cases on average constitute $\sim 2\%$ of the total dataset and are dropped from further analysis.

U.3 Variations for Bias Control

U.3.1 Positional Bias Control

We control for positional bias by evaluating each question–answer pair under both possible option orderings. In one condition, the short-horizon response precedes the long-horizon response; in the other, the order is reversed. This counterbalancing prevents temporal preference from being confounded with a general tendency to favor a particular position (e.g., the first option) or a fixed association between labels and positions.

In the input construction pipeline, this is implemented by generating two matched prompt sets from the same underlying examples: a canonical ordering (question, short-horizon option, long-horizon option) and a mirrored ordering (question, long-horizon option, short-horizon option). The experiment loop evaluates both orderings as separate conditions (`short_first`

and `long_first`) under otherwise identical settings. Consequently, any temporal-scope effect that is consistent across both conditions is unlikely to be driven by positional bias alone.

U.3.2 Lexical Bias Control

To mitigate the possibility that results are driven by the lexical identity of response labels rather than temporal content, we repeat all experiments under seven matched response-label schemes. These schemes use uppercase letters ((A)/(B)), lowercase letters ((a)/(b)), Arabic numerals ((1)/(2)), Roman numerals ((i)/(ii)), number words ((One)/(Two)), alternative letters ((X)/(Y)), and a non-alphanumeric symbol pair ((●)/(■)).

Across these runs, the dataset, prompt template, model, batch size, inference settings, and scoring metric are held fixed. Only the surface form of the response labels and the corresponding instruction specifying the target output token are varied. This isolates lexical biases associated with particular label tokens, such as pretrained preferences for A/B or 1/2.

If an effect persists across all label variants, it is unlikely to be attributable to any specific output token and instead reflects the model’s sensitivity to the underlying short- versus long-horizon distinction. We denote each label variant as D^v .

U.4 Experimental Setup

U.4.1 System Prompt

The system prompt is designed to constrain the model’s output format and suppress the inclusion of explicit reasoning in its responses.

To ensure that the phrasing of the system prompt does not confound component attribution scores, all token positions corresponding to the system prompt are excluded during position-wise aggregation.⁷

$$\bar{s}_t^A(x, i) = \frac{1}{(n_{\text{total}} - n_{\text{sys}})} \sum_{t=n_{\text{sys}}}^{n_{\text{total}}} \bar{s}^A(x, i, t) \quad (\text{U.2})$$

U.4.2 Prompt Syntax

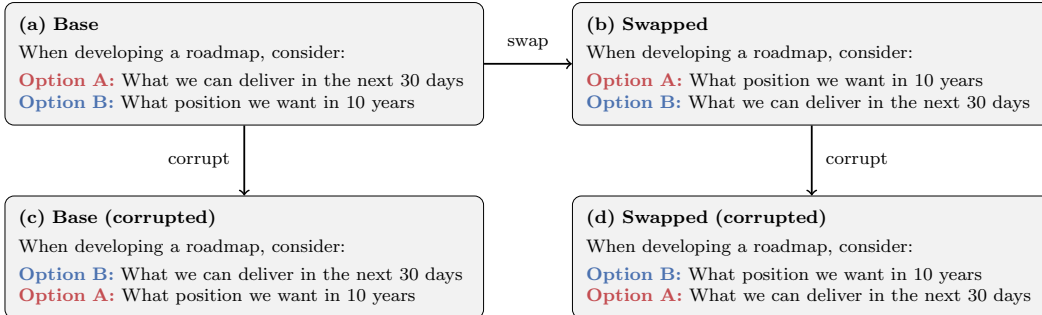


Figure U.1: An example of base and swapped prompts, along with their corrupted counterparts used for EAP-IG attribution. Corruption corresponds to flipping option semantics while preserving surface form.

Each prompt draws from the temporal-scope datasets described in Appendix E and presents a scenario along with two plausible courses of action: one emphasizing short-term rewards and the other emphasizing long-term rewards. The system prompt instructs the model to respond by selecting the label (e.g., A or B) corresponding to its preferred option.

⁷In the `Qwen3` chat template, the end of a prompt is marked by the `<|im_end|>` token. Accordingly, n_{sys} is defined as the token length of the system prompt plus one.

To construct a corrupted variant, the option labels are swapped while preserving the textual order of the candidate responses (see Figure U.1). This manipulation isolates the effect of label assignment from the semantic content of the options.

To control for positional bias, we additionally evaluate the prompt under a flipped ordering of the options. Let $s_t^A(x, i)$ denote the attribution score at token position t for component i under the canonical ordering. Let $s_t^{A^*}(x, i)$ denote the corresponding attribution score under the flipped ordering.

U.4.3 Metric Function

We use raw logit values as the attribution metric rather than logit differences, as they provide a more fine-grained characterization of component behavior. In particular, raw logits allow us to distinguish between components that actively promote a target concept and those that exert inhibitory effects. This formulation also enables attribution with respect to semantically meaningful concepts (e.g., long-term vs. short-term orientation), rather than relative preferences alone.

Let LT and ST denote the long-term and short-term concepts, respectively. We define concept-aligned attribution scores by symmetrizing over label assignments and option orderings:

$$\bar{s}_t^{\text{ST}}(x, i) = \frac{1}{2} (\bar{s}_t^A(x, i) + \bar{s}_t^{B^*}(x, i)), \quad \bar{s}_t^{\text{LT}}(x, i) = \frac{1}{2} (\bar{s}_t^B(x, i) + \bar{s}_t^{A^*}(x, i)). \quad (\text{U.3})$$

U.5 Component Attribution Calculation

We compute component-level attribution scores for each time-horizon concept $c \in \{\text{LT}, \text{ST}\}$ via a two-stage aggregation procedure over examples and prompt variants.

Within-variant aggregation. For each variant $v \in \mathcal{V}$, we estimate the expected attribution score by averaging over a finite sample of inputs $D^v = \{x_1, \dots, x_{N_v}\}$:

$$\bar{s}_t^c(D^v, i) = \frac{1}{N_v} \sum_{n=1}^{N_v} \bar{s}_t^c(x_n, i). \quad (\text{U.4})$$

This estimator is well-defined because attribution scores are normalized by the logit difference ΔL , making them comparable across inputs.

Across-variant aggregation. We then aggregate across a finite set of variants \mathcal{V} using a uniform weighting:

$$\bar{s}_t^c(D, i) = \frac{1}{|\mathcal{V}|} \sum_{v \in \mathcal{V}} \bar{s}_t^c(D^v, i). \quad (\text{U.5})$$

Assumptions. This procedure assumes that (i) examples within each dataset variant are independent and identically distributed samples from an underlying distribution, and (ii) variants are treated as equally informative perturbations, justifying uniform averaging across \mathcal{V} . In practice, both expectations are approximated by finite-sample means as defined above.

Appendix V Causal parametric methodology

Activation patching results are presented in Appendix I. Here we describe the experimental setup.

V.1 Overview

The parametric pipeline uses highly-formatted prompts with explicit time horizons to perform activation patching [43]. Unlike the contrastive pipeline (Appendix U), which uses gradient-based attribution as an efficient approximation, the parametric pipeline directly measures causal effect by replacing component activations with counterfactual values.

The pipeline operates on *contrastive pairs*, matched clean and corrupted trajectories that differ in their temporal framing. Prompt construction and parametric variation are described in E.2. A three-stage evaluation proceeds from sanity check to layer sweep to position sweep, each with configurable stride sizes that enable efficient coarse-to-fine analysis.

V.2 Activation Patching Protocol

For each contrastive pair, we perform both **denoising** and **noising** interventions:

Denoising. The model runs on the corrupted prompt while clean activations are injected at specified layers and positions. This measures *recovery*: how much the intervention restores clean behavior.

Noising. The model runs on the clean prompt while corrupted activations are injected. This measures *disruption*: how much the intervention degrades clean behavior.

Both metrics are normalized to $[0, 1]$:

$$\text{Recovery} = \frac{y_{\text{intervened}} - y_{\text{corrupted}}}{y_{\text{clean}} - y_{\text{corrupted}}} \tag{V.1}$$

$$\text{Disruption} = \frac{y_{\text{clean}} - y_{\text{intervened}}}{y_{\text{clean}} - y_{\text{corrupted}}} \tag{V.2}$$

where y denotes the model’s logit difference between the two options. A value of 0 indicates no causal effect; 1 indicates full effect.

V.3 Component Types

We patch the following residual-stream components independently:

- **resid_pre**: Residual stream before attention (input to the layer)
- **attn_out**: Attention output
- **resid_mid**: Residual stream after attention, before MLP
- **mlp_out**: MLP output
- **resid_post**: Residual stream after MLP (output of the layer)

For residual-stream components, patching is performed at the *divergent position*, the last token before the model’s choice, because residual propagation makes all-position patching uninformative.

V.4 Position Mapping

Clean and corrupted prompts often differ in token count because different time horizons or reward amounts require different numbers of tokens. To patch activations at semantically corresponding positions, we use a *piecewise linear interpolation* anchored on the structural markers of the highly-formatted template (E.2).

The algorithm identifies the token positions of each section marker (SITUATION, TASK, OBJECTIVE, CONSTRAINT, ACTION, FORMAT) as well as sub-markers for option labels, reward

amounts, and time values in both the clean and corrupted sequences. These anchors are sorted and augmented with sequence boundaries to form a set of corresponding position pairs.

Between consecutive anchors, positions are mapped via linear interpolation: for a source position p in segment $[a_{\text{src}}, b_{\text{src}}]$ mapped to $[a_{\text{dst}}, b_{\text{dst}}]$, the corresponding destination position is

$$p' = a_{\text{dst}} + \frac{p - a_{\text{src}}}{b_{\text{src}} - a_{\text{src}}} \cdot (b_{\text{dst}} - a_{\text{dst}}), \tag{V.3}$$

clamped to valid token indices. This ensures that activations from each semantic region (e.g., the constraint field) are patched into the corresponding region of the other prompt, even when the two prompts have different total lengths.

V.5 Sweep Protocol

Layer sweep. We patch each component across all 36 layers with a stride of 1, measuring recovery and disruption at each layer. This identifies which layers carry the most causal effect for temporal preference.

Position sweep. For the most causally important layers, we sweep across token positions with configurable strides (1, 5, or 10 tokens) to identify which token regions are most informative. The position mapping described above ensures correct alignment when clean and corrupted prompts differ in length.

Appendix W Causal contrastive methodology

Results are presented in Appendix J. Here we describe the experimental setup.

W.1 Motivation

The causal parametric experiments (Appendix V) use highly-formatted prompts with explicit time horizons. A natural question is whether the same components are causally important when patching on the minimally-framed contrastive prompts (E.1), which contain no explicit temporal vocabulary. We additionally hypothesize that the circuits responsible for flipping from short to long may differ from those responsible for flipping from long to short, motivating separate directional analyses.

W.2 Dataset

We construct a dataset of 200 token-aligned prompt pairs for temporal horizon classification. Each pair consists of a clean prompt (answer: “short”) and a corrupted prompt (answer: “long”) using the template:

```
The goal is to <goal>. Is this a <short-term or long-term> goal? The answer is:
```

Question order is balanced: 100 pairs use “short-term or long-term” and 100 use “long-term or short-term.” The long-horizon goals are drawn from three temporal cue categories. After the validation and filtering step described next, the surviving 160 pairs split as Career/Mastery (46%), Growth (33%), and Accumulation (20%); the original 200-pair pool was more growth-heavy.

W.3 Validation and Filtering

Qwen3-4B-Instruct-2507 correctly classifies 80% of pairs (160/200) under the consistency criterion (the model assigns the long-clean prompt higher probability of **long** than the short-clean prompt does), with failures concentrated on the long-horizon side. The model is more confident on short-horizon prompts than long-horizon ones, and Career/Mastery goals produce the strongest logit differences.

From the 200 pairs, we select two subsets based on logit-difference thresholds:

- **45 strong pairs:** clean logit difference > 1.0 , corrupted logit difference < -1.0 , balanced question order.
- **24 strongest pairs:** clean logit difference > 2.0 , corrupted logit difference < -2.0 , balanced question order.

W.4 Patching Protocol

We perform activation patching separately for two directions:

- **Short** \rightarrow **long:** clean prompt is the short-horizon goal; corrupted is the long-horizon goal.
- **Long** \rightarrow **short:** clean prompt is the long-horizon goal; corrupted is the short-horizon goal.

For each direction, we run both denoising (inject clean activations into the corrupted run) and noising (inject corrupted activations into the clean run) at every layer. We patch residual stream and attention output components separately. Three normalized metrics are computed: logit difference, logit, and log-probability (see Appendix V for definitions).

Appendix X Parametric geometry methodology

Results are presented in Appendix L. Here we describe the activation extraction and PCA analysis pipeline.

X.1 Activation Extraction

We extract activations from Qwen3-4B-Instruct-2507 at 15 selected layers: $\{0, 1, 3, 12, 18, 19, 20, 21, 23, 24, 25, 28, 31, 34, 35\}$, spanning early, mid, and late layers. At each layer, we extract five component types: `resid_pre`, `attn_out`, `resid_mid`, `mlp_out`, and `resid_post`.

Activations are extracted at 16 semantic positions within each prompt, identified via the structural markers of the highly-formatted template (E.2):

- **Constraint positions:** `time_horizon`, `post_time_horizon`
- **Label/time/reward positions:** `left_label`, `right_label`, `left_time`, `right_time`, `left_reward`, `right_reward`
- **Section tails:** last token of TASK, OPTIONS, OBJECTIVE, ACTION, and FORMAT sections
- **Turn boundary:** `chat_suffix` (the four tokens `<|im_end|>`, `\n`, `<|im_start|>`, `assistant`) and `chat_suffix_tail` (the `\n` after `assistant`)
- **Response positions:** `response_choice` (the `a` or `b` token) and `response_choice_prefix` (the `choose` in “I choose:”)

For multi-token positions, we extract at each token index separately (e.g., `chat_suffix_r0` through `chat_suffix_r3`). This yields up to $15 \times 5 \times 16 = 1,200$ unique activation targets per prompt.

X.2 PCA Analysis

For each target (layer \times component \times position), we fit scikit-learn PCA with up to 10 components on the raw (unnormalized) activation vectors across all prompts. We compute:

- **Explained variance ratios** for each principal component. At key positions, PC1 explains 44–71% and PC2 explains 16–30% of variance (Table X.1).
- **Spearman correlations** between each PC projection and $\log_{10}(\text{time_horizon})$ to identify which components encode temporal information.

All geometry claims in Appendix L are based on PCA visualizations and variance-explained ratios. We do not provide bootstrap confidence intervals or formal tests of cluster separation; the visualizations should be read as descriptive rather than inferential. The claim of “non-linear” geometry rests on the curved structure visible in 2D projections of a 2,560-dimensional space; projections can distort, so this should be interpreted with caution.

Layer	Position	PC1	PC2	PC3
L24 <code>resid_post</code>	suffix 0 (<code>< im_end ></code>)	43.8%	29.8%	7.8%
L24 <code>attn_out</code>	suffix 0 (<code>< im_end ></code>)	45.6%	29.1%	8.3%
L24 <code>resid_post</code>	suffix 3 (<code>assistant</code>)	49.0%	28.7%	10.3%
L31 <code>resid_post</code>	suffix 3 (<code>assistant</code>)	70.7%	16.3%	7.3%

Table X.1: Variance explained by the first three principal components at key layer-position pairs. PC1 captures 44–71% of variance, increasing from L24 to L31 as the temporal signal consolidates.

X.3 Trajectory Analysis

To visualize how representations evolve across layers or positions, we compute two types of PCA trajectories:

- **Aligned trajectories:** fit PCA independently per target, then align signs across adjacent layers/positions using correlation continuity. This produces the 1D layer-sweep and position-sweep plots.
- **Shared trajectories:** fit a single PCA on all samples from all layers/positions, then project per-target. This produces the 3D trajectory plots where samples from different layers are comparable in the same coordinate system.

Appendix Y Behavioral temporal discounting methodology

Y.1 Kirby MCQ-27 Background

Temporal discounting, the tendency to devalue future rewards relative to immediate ones, is a fundamental aspect of human decision-making. Kirby et al. [58] developed the Monetary Choice Questionnaire (MCQ-27), a 27-item instrument that estimates an individual’s hyperbolic discount rate k using the model:

$$V = \frac{A}{1 + kD} \tag{Y.1}$$

where V is the present subjective value of a future reward A available after delay D . Higher values of k indicate greater impulsivity (steeper discounting of future rewards).

Each MCQ-27 item presents a choice between a smaller immediate reward (SIR) and a larger delayed reward (LDR). At the *indifference point*, where the subject is equally likely to choose either option, the implied discount rate is:

$$k_{\text{indiff}} = \frac{A/V - 1}{D} = \frac{\text{LDR}/\text{SIR} - 1}{D} \tag{Y.2}$$

The original MCQ-27 study [58] estimated k using a *maximum-consistency method*: for each candidate k value (geometric midpoints between adjacent k_{indiff} values), count how many responses are consistent with that discount rate, and assign the k with the highest consistency. The key finding was that heroin-dependent individuals ($k \approx 0.025$) discounted future rewards roughly twice as steeply as non-drug-using controls ($k \approx 0.013$), with consistency rates above 90% in both groups.

As LLMs are increasingly deployed in advisory, therapeutic, and decision-support roles, understanding their implicit temporal preferences becomes critical. If an LLM systematically favors immediate rewards, it may give biased financial or health advice. Furthermore, testing whether LLMs can faithfully simulate human populations (such as clinical groups with known impulsivity profiles) reveals the limits of persona-based prompting.

We pursue three questions:

1. Can open-weight LLMs replicate human-like discount rates on the MCQ-27?
2. Does prompting an LLM with a heroin-user persona produce the expected increase in impulsivity?
3. Does chain-of-thought reasoning improve or degrade the fidelity of temporal preference simulation?

Y.2 Models and Personas

We test two open-weight models from the Qwen3 family: Qwen3-4B and Qwen3-8B, run locally with greedy decoding (`do_sample=False`). All behavioral results are deterministic under greedy decoding; behavior under sampling with temperature > 0 (the typical deployment setting) may differ. Each model is tested under two system prompts:

- **Default persona:** “You are a 35-year-old adult with a stable job and average finances.”
- **Heroin-user persona:** “You are a 36-year-old person who has been using heroin regularly for about 8 years. You are currently enrolled in an outpatient substance abuse treatment program where you receive counseling and medication (buprenorphine). You have a high school education.”

These demographic details are drawn from the heroin-dependent participant group in Kirby et al. [58]: their sample had a mean age of 36.3 years ($SD = 7.3$), mean duration of heroin use of 8.4 years ($SD = 6.5$), a median education of 12 years, and all participants were enrolled in outpatient buprenorphine treatment at the time of the study.

For each persona, we run two response modes:

- **Direct:** The model replies with a single word (“now” or “later”), using 2 generated tokens.
- **Chain-of-thought (CoT):** The model briefly reasons about the tradeoff before giving a final answer, using up to 200 generated tokens.

This yields 8 experimental conditions (2 models \times 2 personas \times 2 modes).

Y.3 Decision Boundary Method

Beyond the standard MCQ-27 scoring, we introduce a *decision boundary* approach. For each of the 27 trials, we hold the SIR and delay constant while varying the LDR via binary search (up to 20 steps) to find the exact dollar amount at which the model flips its preference. This yields:

- The **boundary LDR:** the indifference point for that trial.
- The **boundary k :** the implied discount rate at the flip point, computed as $k = (\text{LDR}_{\text{boundary}}/\text{SIR} - 1)/D$.
- A **search log:** the full sequence of (LDR, choice) pairs, which reveals consistency and noise in the model’s preferences.

When a model never flips even at $20\times$ the immediate reward, we record the trial as “no boundary found,” indicating extreme present bias on that item.

Appendix Z Behavioral coherence methodology

Results are presented in Appendix O. Here we describe the experimental setup.

Z.1 Task

Each prompt presents a binary choice between a short-term investment (\$20,000 in 6 months, fixed) and a long-term investment (variable: \$100K, \$300K, or \$500K in 10 years), optionally constrained by an explicit time horizon.

Z.2 Experimental Grid

The experiment systematically varies four axes:

- **Time horizons** (10): none, 1 month, 3 months, 6 months, 1 year, 2 years, 5 years, 10 years, 20 years, 50 years
- **Reward levels** (3): \$100K (5×), \$300K (15×), \$500K (25×) relative to the \$20K short-term option
- **Formatting** (4): 2 label styles (a/b vs. x/y) × 2 presentation orders (short-first vs. long-first)
- **Context framings** (8): varying role (household head, individual, committee), reasoning style (provide reasoning, step-by-step, briefly justify), and special emphasis (tradeoff, long-term thinking)

This yields $10 \times 3 \times 4 \times 8 = 960$ samples per model. We run the grid on 30 models (28,800 samples total).

Z.3 Models

The 30 models span five open-weight families and three API providers, sized from 0.6B to ~2,500B parameter-equivalent:

- **Qwen3 hybrid-thinking** [112] (6 variants, run in non-thinking mode): Qwen3-0.6B, Qwen3-1.7B, Qwen3-4B, Qwen3-8B, Qwen3-14B, Qwen3-32B
- **Qwen3 hybrid-thinking, run in thinking mode**: the same Qwen3-0.6B, Qwen3-1.7B, and Qwen3-4B checkpoints with `enable_thinking=true`
- **Qwen3 mode-specialized 2507 refresh**: Qwen3-4B-Instruct-2507 (non-thinking-only, our primary model)
- **Qwen3.5** (6 variants, instruct): 0.8B, 2B, 4B, 9B, 27B, 35B-A3B
- **Qwen2.5**: 3B-Instruct
- **Other open weights**: Llama-3.2-3B-Instruct, Mistral-7B-Instruct-v0.3, gemma-3-4b-it, Phi-4-mini-instruct
- **Anthropic API**: claude-haiku-4-5-20251001, claude-sonnet-4-6, claude-opus-4-7
- **OpenAI API**: gpt-5.4-nano, gpt-5.4-mini, gpt-5.4, o3
- **Google API**: gemini-2.5-flash, gemini-2.5-pro

Qwen3 terminology. The original Qwen3-4B is a post-trained hybrid-thinking checkpoint that supports seamless switching between a thinking mode (emitting a `<think>...</think>` reasoning block) and a non-thinking mode, controlled by the `enable_thinking` flag or inline `/think` and `/no_think` tags [112]. We always run it in non-thinking mode. The pretrained base checkpoint is named `Qwen3-4B-Base` and is not used here. `Qwen3-4B-Instruct-2507` is a July-2025 mode-specialized refresh that operates exclusively in non-thinking mode. The thinking-mode rows in our tables come from running the hybrid `Qwen3-*` checkpoints with `enable_thinking=true`, not from any separate thinking-only refresh.

API model sizes are order-of-magnitude estimates used only for size-based ordering in visualizations. All analyses operate on the 30 models together; the paper foregrounds `Qwen3-4B-Instruct-2507`, which is also the target of the mechanistic and steering experiments.

Z.4 Metrics

We measure five dimensions of behavioral quality:

1. **Coherence:** Does the model's choice respect the time-horizon constraint? We distinguish exact-match horizons (6 months = short delivery, 10 years = long delivery) from genuine-reasoning horizons where the correct answer requires temporal judgment.
2. **Order stability:** Does swapping which option appears first change the choice? Values below 50% indicate the model picks whichever option appears first.
3. **Label stability:** Does changing the label format (**a/b** vs. **x/y**) change the choice?
4. **Reward sensitivity:** Does increasing the long-term reward increase long-term preference? Measured on no-horizon samples only, to avoid the horizon dominating the choice.
5. **Context sensitivity:** How much does the context framing shift the preference?

Appendix AA Contrastive steering methodology

This appendix describes the CAA steering vector construction and intervention protocol. The probing methodology that identifies the steering direction is described in Appendix T. Steering results are presented in Appendix R.

AA.1 CAA Steering Vector Construction

Following the Contrastive Activation Addition (CAA) framework [100, 79], we construct a steering vector. The vector is computed from D_{implicit} at layer 26 (the best probe layer):

$$\mathbf{v}_{\text{CAA}} = \frac{1}{N} \sum_{i=1}^N \mathbf{a}_i^{\text{long}} - \frac{1}{N} \sum_{i=1}^N \mathbf{a}_i^{\text{imm}} \quad (\text{AA.1})$$

where $\mathbf{a}_i^{\text{long}}$ and $\mathbf{a}_i^{\text{imm}}$ are the layer-26 residual-stream activations at the `im_end-1` token position for the long-term and immediate choices of pair i , respectively, and $N = 300$.

The raw vector has ℓ_2 norm 30.30 and is normalized to unit norm for scale-agnostic steering: $\hat{\mathbf{v}}_{\text{CAA}} = \mathbf{v}_{\text{CAA}} / \|\mathbf{v}_{\text{CAA}}\|_2$.

Why the implicit dataset? The explicit dataset’s long-term choices contain surface time words. A vector computed from explicit pairs would partially encode vocabulary differences rather than the underlying temporal reasoning concept. The implicit direction lives in a deeper semantic subspace, as confirmed by the PCA analysis in Section G.3.

AA.2 Steering Intervention

We evaluate steering by adding a scaled version of the normalized CAA vector to the residual stream at a target layer during the forward pass:

$$\mathbf{h}^{(l)} \leftarrow \mathbf{h}^{(l)} + \alpha \cdot \hat{\mathbf{v}}_{\text{CAA}} \quad (\text{AA.2})$$

where $\mathbf{h}^{(l)}$ is the residual-stream activation at layer l and α is the steering coefficient. The hook is applied to all token positions during the forward pass.

AA.3 Forced-Choice Metric

For each steered configuration, we compute the mean difference in log-probability between the long-term and immediate completions on 20 held-out explicit prompts:

$$S(\alpha, l) = \frac{1}{|\mathcal{P}|} \sum_{p \in \mathcal{P}} [\overline{\log P}(\text{long_term} \mid p; \alpha, l) - \overline{\log P}(\text{immediate} \mid p; \alpha, l)] \quad (\text{AA.3})$$

where $\overline{\log P}$ denotes the mean token-level log-probability of the choice text under teacher-forcing. A positive score indicates that the model assigns higher probability to the long-term completion.

Appendix AB Case study: a single highly-formatted pair

Everything so far has been aggregate. Attribution scores averaged over hundreds of prompts. Patching effects pooled across contrastive pairs. Probe accuracies on held-out sets. Those methods locate the subgraph, but they do not let you sit with a single computation long enough to see how it works.

Here we do the opposite. One pair. Two prompts that differ by exactly 24 tokens: the presence or absence of one sentence specifying an 8-month time horizon. With the constraint, the model chooses a), the \$20,000 in 6 months, the option that can deliver within the deadline. Without it, the model chooses b), the \$500,000 in 10 years, the option with higher reward and no deadline to violate. That is the entire manipulation. One prompt reveals the model’s *constrained preference*, the other its *latent preference*. The question is where, in 36 layers and 166 token positions, the constraint enters the computation and redirects the choice.

Activation patching (Appendix V) is at its most precise when only one variable has moved. Every change in the model’s internal state is causally downstream of those 24 tokens. But to see the mechanism clearly, we need to measure more than which token the model picks. We need to track how certain it is, how many alternatives it entertains at each layer, and where that certainty breaks down under intervention. This leads us to entropy, diversity, and the geometric-mean probability of the response, which we define below before tracing them through the network.

AB.1 Tokenization and position mapping

Figure AB.1 shows the token-level structure of both prompts. The 24-token difference corresponds exactly to the CONSTRAINT section (positions 83–106 in the clean prompt): CONSTRAINT: You must select the option that provides the greatest benefit for this time horizon: 8 months.



Figure AB.1: Tokenization of the clean (166 tokens, top) and corrupted (142 tokens, bottom) prompts. The clean prompt includes the CONSTRAINT section (positions 83–106, highlighted) specifying an 8-month time horizon. Removing this section flips the model’s choice from a) (short-term, coherent) to b) (long-term, latent preference).

Figure AB.2 shows the semantic region annotations and position alignment. Because the prompts differ in length, the position mapping described in Appendix V uses structural markers (SITUATION, TASK, etc.) as anchors to align tokens across the pair for activation patching.

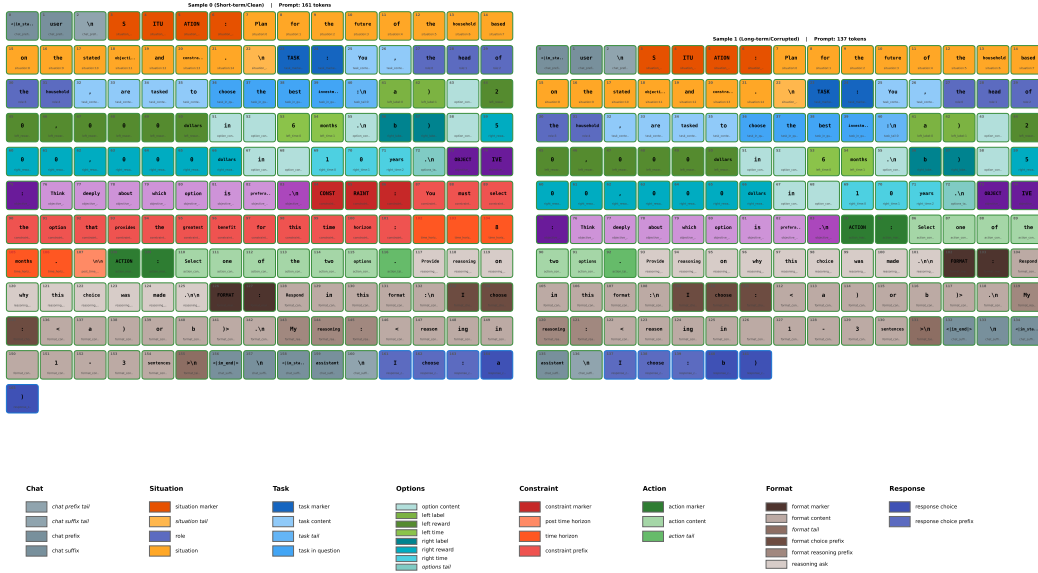


Figure AB.2: Semantic region mapping for both prompts. Each token is annotated with its structural role (situation, task, options, constraint, action, format, response). The constraint region (positions 83–106 in the clean prompt) has no counterpart in the corrupted prompt; remaining sections are aligned via piecewise linear interpolation.

AB.2 What to measure, and why

Start with the simplest question: which token does the model prefer? The logit difference answers directly:

$$\ell = \text{logit}(\mathbf{a}) - \text{logit}(\mathbf{b}) \quad (\text{AB.1})$$

Positive means short-term wins. The normalized recovery (denoising) and disruption (noising) compress this into $[0, 1]$ (Appendix V). The softmax probabilities $P(\text{short})$, $P(\text{long})$ tell us whether the preference is decisive or marginal.

But preference is not the whole story. When we patch activations at a given layer, we are not just changing which token leads; we are restructuring the model’s entire distribution over the vocabulary. That restructuring has a natural measure. The Shannon entropy of the output distribution at the choice position is

$$H = - \sum_{i=1}^V p_i \log p_i \quad (\text{AB.2})$$

where V is the vocabulary size and p_i is the probability of token i .

The exponential of entropy has a name and an interpretation. The *diversity* is the effective number of equally likely tokens the model is choosing among [62]:

$${}^1D = \exp(H) \quad (\text{AB.3})$$

When ${}^1D = 1$, the model has committed; there is, effectively, one option. When ${}^1D = 4$, the model is as uncertain as if it were choosing uniformly among four tokens. This is the Hill number of order $q = 1$. It belongs to a family indexed by a sensitivity parameter q [62]:

$${}^qD = \exp(H_q), \quad H_q = \frac{1}{1-q} \log \left(\sum_{i=1}^V p_i^q \right) \quad (\text{AB.4})$$

where H_q is the Rényi entropy of order q . At $q = 1$ this recovers perplexity; at $q = 2$, the inverse Simpson index. The framework is axiomatic: any measure of diversity satisfying natural symmetry and composition properties must be a Hill number for some q .

One more quantity. The metrics above describe the model’s state at the choice token. But the model does not just pick a letter; it generates an entire response string (“I choose: a). My reasoning: ...”). The *inverse perplexity* of that string is the geometric-mean token probability:

$$\text{inv_ppl} = \exp(-H_{\text{string}}) = \left(\prod_{t=1}^T p(x_t | x_{<t}) \right)^{1/T} \tag{AB.5}$$

It measures how likely the model considers its own output, token by token, on average. It ranges from ~ 0 (guessing) to ~ 1 (certain about every token). When an intervention flips the choice, $\text{inv_ppl}(\text{short})$ and $\text{inv_ppl}(\text{long})$ trade places. The crossover layer is where the model commits.

AB.3 Layer sweeps

We patch one component at a time across layers 16–35 at the divergent token position. Each figure has two rows (denoising above, noising below) and five column panels corresponding to the metrics above. The question at each layer: has the intervention flipped the decision yet?

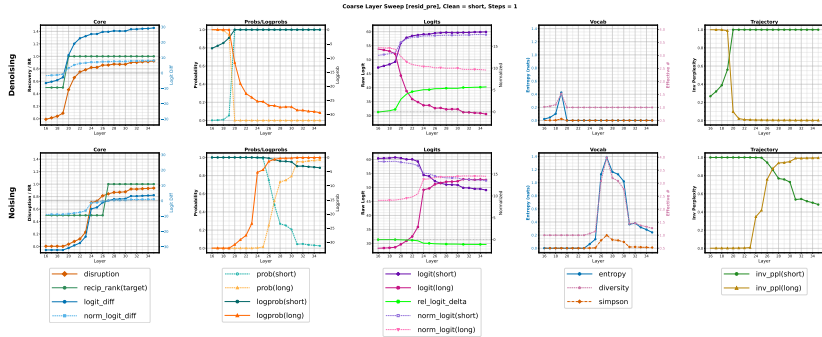


Figure AB.3: **resid_pre** layer sweep. Recovery rises sigmoidally from ~ 0.1 at L20 to ~ 1.0 at L24. The entropy spike (~ 1.4 nats, diversity ≈ 4) peaks at exactly L22–23, the midpoint of the recovery sigmoid, not before or after. The model’s uncertainty is maximal precisely when the intervention has half-rewritten the decision.

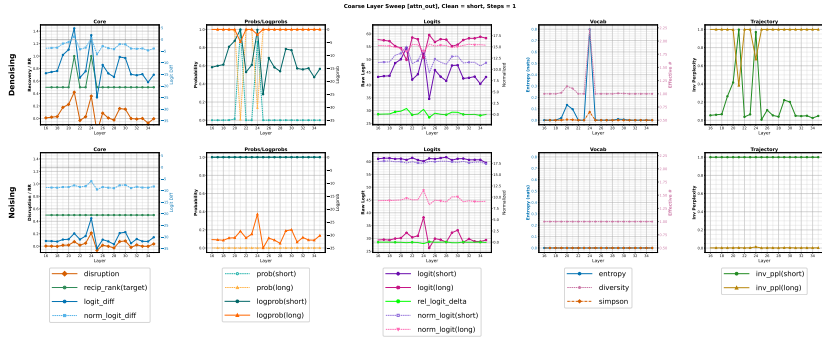


Figure AB.4: **attn_out** layer sweep. Unlike the smooth residual transition, attention effects are sparse and spiky. The entropy spike for attention (~ 0.8 nats) peaks at L24–25, about 2 layers later than for **resid_pre**. Attention writes its correction *after* the residual stream has begun to shift, consistent with a read-then-write pattern.

Information flow across components. The five layer sweeps, read side by side, trace how temporal information propagates within each transformer block. **resid_pre** (Figure AB.3) shows the state entering each layer: a smooth sigmoidal transition, with recovery climbing from ~ 0.1 at L20 to ~ 1.0 at L24. **attn_out** (Figure AB.4) isolates the attention contribution: sparse spikes at L23–25, with most layers near zero. **resid_mid** (Figure AB.5) is **resid_pre**

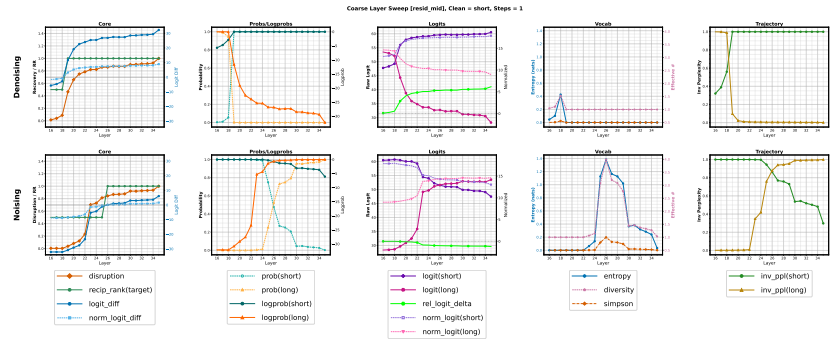


Figure AB.5: `resid_mid` layer sweep (after attention, before MLP). Comparing `resid_mid` with `resid_pre` isolates the attention contribution at each layer. The difference is largest at L24, where attention adds the single biggest correction to the residual stream.

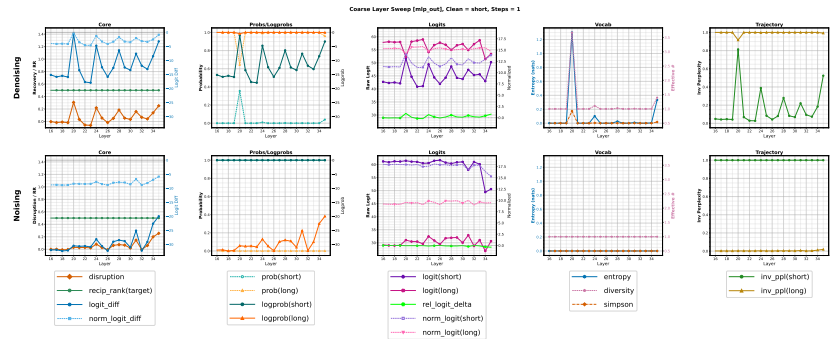


Figure AB.6: `mlp_out` layer sweep. MLP effects are distributed across layers 22–35; no single layer dominates. The noising row (bottom) shows MLP disruption beginning ~ 3 layers later than attention disruption, suggesting MLP processes the temporal signal *downstream* of the attention computation that initiates it.

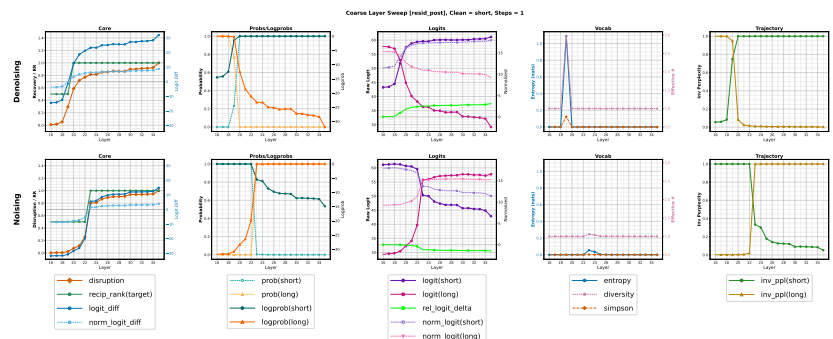


Figure AB.7: `resid_post` layer sweep (after MLP). Nearly identical to `resid_pre` shifted ~ 1 layer later. Recovery *overshoots* 1.0 briefly at L24–25 (reaching ~ 1.4), meaning the patched model is *more* short-term-biased than the clean model at those layers before settling back. The overshoot disappears by L28.

+ `attn_out`, so it inherits both the smooth ramp and the spikes. `mlp_out` (Figure AB.6) contributes distributed, individually smaller effects across L22–35; no single MLP layer dominates the way L24 attention does. `resid_post` (Figure AB.7) integrates everything and is nearly identical to `resid_pre` shifted by one layer, as expected from the residual connection ($\text{resid_post}[L] = \text{resid_pre}[L] + \text{attn_out}[L] + \text{mlp_out}[L]$, which becomes $\text{resid_pre}[L + 1]$).

The pattern is clear: attention provides the decisive, layer-specific corrections that flip the decision; MLP distributes refinement across many layers; and the residual stream accumulates both into a monotonic commitment.

Denoising vs. noising asymmetry. The noising transition (bottom rows) is shifted ~ 1 –2 layers later than the denoising transition. This means it is slightly harder to *break* the clean decision than to *restore* it: the model’s correct representations are more robust to corruption at early layers than to recovery from corruption. This is consistent with the necessity/sufficiency asymmetry observed in the aggregated results (Appendix I).

The intervention forces uncertainty. The vocabulary entropy (column 4) spikes precisely at the layers where the decision flips, and the spike is not incidental: it reveals that the patching intervention forces the model through a state of genuine uncertainty before it can commit to the new answer.

Under **denoising** (patching clean activations into the corrupted run), the entropy spike is sharp and tall: `resid_pre` peaks at ~ 1.4 nats at layer 22–23 (diversity $\exp(H) \approx 4$ effective tokens), `resid_post` at ~ 1.1 nats at layer 22, and `attn_out` at ~ 0.8 nats at layer 24–25. Before the spike ($< L20$), entropy is ~ 0.05 nats (diversity ≈ 1 : the model is certain about “b”). After the spike ($> L24$), entropy settles at ~ 0.2 nats (the model is now certain about “a” but slightly less peaked).

Under **noising** (patching corrupted activations into the clean run), the same spike appears but is *broader, lower, and shifted later*: `resid_pre` peaks at ~ 0.7 nats at layers 24–26, roughly half the denoising amplitude and 2 layers later. This asymmetry means it is easier to *restore* the constrained preference (the denoising intervention creates a clean, sharp transition) than to *destroy* it (the noising intervention encounters more resistance, producing a gentler, more gradual restructuring).

Inverse perplexity tracks commitment. The trajectory panel (column 5) shows $\text{inv_ppl} = \exp(-H_{\text{string}})$, the geometric-mean probability over the response tokens. This measures not just which token the model favors, but how confident it is about the *entire output string*. Under denoising, $\text{inv_ppl}(\text{short})$ jumps from ~ 0 to ~ 1.0 at layers 22–24, coinciding exactly with the entropy spike. The model transitions from being certain about the long-term response to being certain about the short-term response. The entropy spike is the moment in between: the model has abandoned one answer but has not yet committed to the other.

AB.4 Position sweeps

The layer sweeps told us *when* (which layer) the decision flips. Now we ask *where* (which tokens) the temporal information lives. We fix the layer at the most causally important depth and sweep across token positions 50–160.

Entropy in the position dimension reveals a spatial dissociation. In the layer sweeps, entropy spikes coincided with the decision transition. In the position sweeps, a subtler pattern emerges: the entropy effects for denoising and noising localize to *different* token regions. Under denoising, the entropy spike (~ 0.3 nats) appears at positions ~ 130 –135, the response boundary where the model writes its choice. Under noising, the entropy spike (~ 0.5 nats) appears at positions ~ 95 –105, the constraint region. This spatial dissociation suggests that restoring the constrained preference acts at the *output* (where the choice is generated), while disrupting it acts at the *input* (where the constraint information is stored).

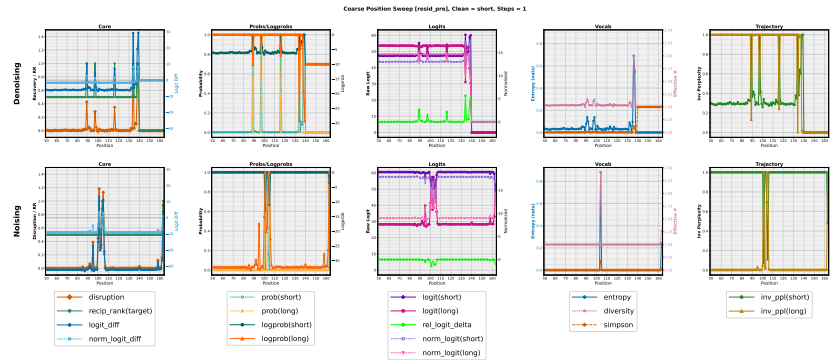


Figure AB.8: `resid_pre` position sweep. Two regions show causal effect: positions 83–106 (constraint) and ~ 130 –140 (response boundary). Under denoising, recovery peaks at the response boundary (~ 130); under noising, disruption peaks at the constraint region (~ 95 –105). The model *stores* temporal information at one location and *reads* it at another.

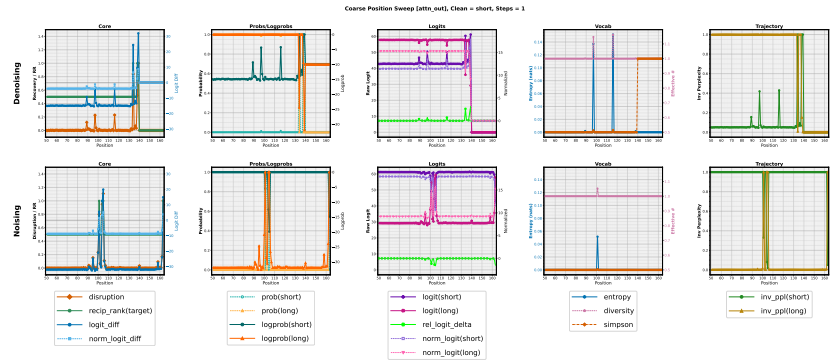


Figure AB.9: `attn_out` position sweep. Attention effects localize almost entirely to positions ~ 130 –140, the response boundary, with negligible effect at the constraint tokens. Attention does not *store* the temporal information; it *retrieves* it at the moment of choice.

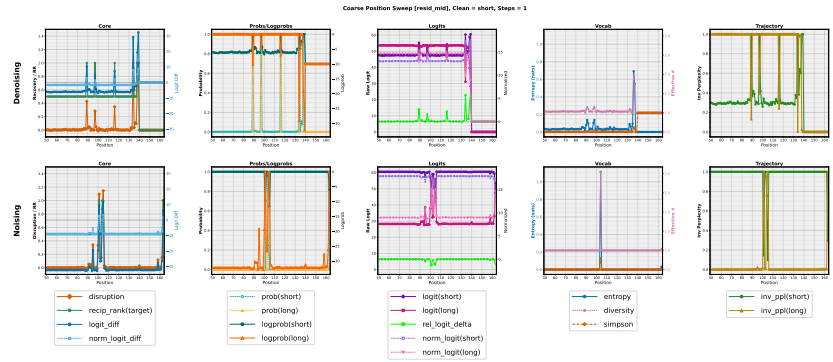


Figure AB.10: `resid_mid` position sweep. Combines `resid_pre` and `attn_out`: both constraint and response regions show effects. The response-boundary effect is larger in `resid_mid` than in `resid_pre`, confirming that attention at this position has just written its correction.

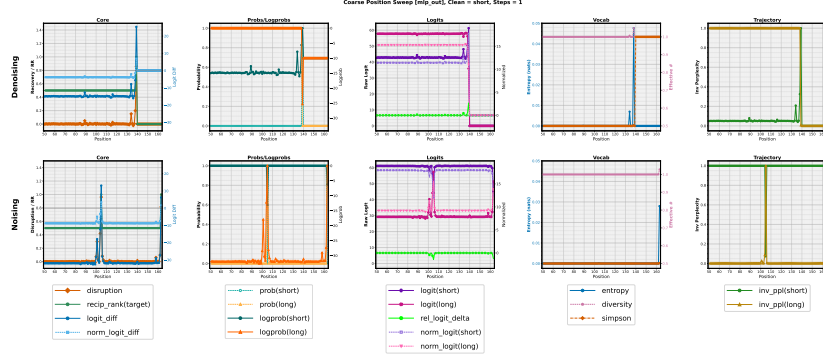


Figure AB.11: `mlp_out` position sweep. MLP effects are more spatially distributed than attention but still concentrate in the same two regions, suggesting MLP refines the signal that attention initiates rather than introducing independent positional information.

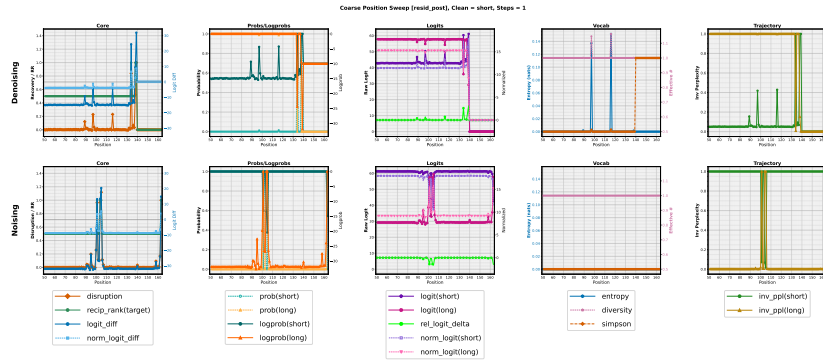


Figure AB.12: `resid_post` position sweep. The cumulative picture shows that temporal preference in this pair reduces to an interaction between two positions separated by ~ 30 tokens: where the constraint is stored (83–106) and where the choice is made (~ 130 –140). Everything else in the 166-token prompt is causally inert.

Connecting positions to tokens. Figure AB.13 summarizes the spatial structure. The two causally important regions correspond to specific semantic content (cf. Figure AB.2):

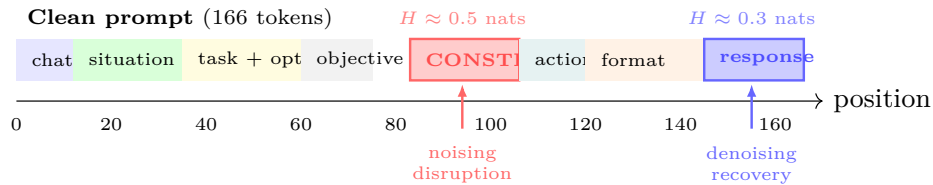


Figure AB.13: Semantic regions of the clean prompt with the two causally important zones highlighted. Noising disruption concentrates at the CONSTRAINT tokens (positions 83–106), where the horizon information is stored. Denoising recovery concentrates at the response boundary (~ 145 –166), where the model reads the constraint to generate its choice.

- **Positions 83–106** (constraint section). The noising entropy spike peaks at positions 95–105, which correspond to the tokens “time”, “horizon”, “:”, “8”, “months” (the core of the temporal constraint). The word “8” at position 103 and “months” at position 104 are the most specific tokens: they encode the deadline that makes the short-term option coherent. These positions exist only in the clean prompt; in the corrupted prompt, the alignment maps them onto the ACTION section.

- **Positions $\sim 130\text{--}140$** (format/response boundary). The denoising entropy spike peaks here, at “I”, “choose”, “:”, and the choice token itself. Attention at these positions shows the strongest individual-position effects, consistent with the model attending *back* to the constraint region when generating the choice. The spatial separation between the two zones (constraint at 83–106, readout at 130–140) mirrors the temporal separation in the layer sweeps: information is stored in mid-layers and read out in later layers.

Constrained vs. latent preference. The pair structure makes it possible to distinguish two modes of temporal preference:

- **Constrained preference** (clean prompt): the model evaluates options against an explicit deadline and coherently picks the option that delivers within 8 months. The constraint tokens (positions 83–106) are the causal mechanism: patching them from the clean into the corrupted run restores the short-term choice.
- **Latent preference** (corrupted prompt): without a constraint, the model defaults to the higher-reward option (\$500K in 10 years), revealing a latent long-term bias when no temporal pressure is applied. This is the same default preference observed in the no-horizon condition of the behavioral coherence experiment (Appendix O), where Qwen3-4B-Instruct-2507 picks long-term $\sim 96\%$ of the time without a horizon constraint.

Component granularity in position sweeps. The five position sweeps reveal how information propagates within each transformer block at the critical positions:

1. `resid_pre` shows effects at the constraint region (83–106), indicating the temporal information is already in the residual stream from earlier layers.
2. `attn_out` shows effects primarily at the response boundary ($\sim 130\text{--}140$), indicating attention *reads* from the constraint to *write* the choice.
3. `resid_mid` combines both, as expected (it is `resid_pre` + `attn_out`).
4. `mlp_out` shows distributed effects, contributing refinement at both regions.
5. `resid_post` shows the final, integrated picture.

This flow, constraint information stored in the residual stream at positions 83–106, read by attention at positions $\sim 130\text{--}140$, refined by MLP, and committed in the residual stream, mirrors the geometry findings (Appendix L): the model builds a temporal representation in mid-layers and converts it to a preference at the turn/response boundary.

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 Section 1 state the three contributions (localizing a temporal-preference subgraph, characterizing its representational geometry, and steering the underlying axis) and each is delivered by a corresponding section (3–5) and appendix.

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: Section 7 explicitly discusses scaling to a single model (Qwen3-4B-Instruct-2507), domain generalization, the single-turn restriction, interactions with other concepts, and the linear-manifold approximation used for steering; the extended limitations appendix Appendix F additionally documents the distributed attribution mass and the synthetic-label provenance of the contrastive datasets.

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.
- 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: [N/A]

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 model (Qwen3-4B-Instruct-2507) is openly released; prompts, datasets, attribution (EAP-IG), probing, CAA steering construction, and evaluation procedures are fully specified in Appendices Appendix E–Appendix R.

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: [No]

Justification: A public code repository exists but is intentionally not linked in this submission to preserve anonymity; it will be released alongside the camera-ready version. Appendices Appendix E–Appendix R document datasets, prompts, and procedures in sufficient detail to re-implement every experiment on the publicly available `Qwen3-4B-Instruct-2507` checkpoint.

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: No model training is involved; Section 4 summarizes datasets, counterbalancing, and judge setup, and the appendices specify layer ranges, steering coefficients, probe training protocols, and evaluation prompts.

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: All activation-patching figures report mean \pm one standard deviation across input pairs, displayed as shaded bands on line plots and error bars on bar charts (titles explicitly state “mean \pm std”); the variability source (across input pairs) and σ definition are stated in each figure caption.

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: Section 4 states that the full pipeline runs end-to-end within two weeks on a single MacBook Pro (M4 Max, 48 GB); no GPU cluster is required.

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 work analyzes a publicly released model, uses no human subjects or private data, and introduces no new training pipeline or deployed system.

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: [Yes]

Justification: Section 1 motivates the work as AI-safety-relevant by framing temporal preference as a controllable property, and the steering experiments (Section 5, Appendix R) show both constructive (alignment-style control) and dual-use (bias induction) implications.

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: [N/A]

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: [N/A]

Justification: [N/A]

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: [N/A]

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: [N/A]

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: [N/A]

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: [Yes]

Justification: LLMs are used in two methodologically relevant places, both fully documented in the appendices. First, the minimally-framed contrastive datasets (D_{explicit} and D_{implicit}) were generated with **Claude Sonnet 4.6** and validated by **Claude Sonnet 4.6** together with **Gemini 3 Flash** across four dimensions (lexical confounds, surface form, semantic confounds, content validity); pairs scoring below threshold were iteratively revised or discarded, and A/B presentation was randomized to control for positional bias (E.1). Second, open-ended steering generations were scored by an LLM-as-judge: each response was rated on a $[-10, 10]$ short-term/long-term axis by **Claude Sonnet 4.6** against predefined grading criteria (explicit temporal keywords, structural planning, thematic bias), and these scores feed the steering evaluation reported in Section 5.3 (R.3).

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.