A POLANYI-LIKE PARADOX IN LARGE LANGUAGE MODELS: THE GAP BETWEEN SPATIAL REPRESENTATION AND DECISION

Anonymous authors
Paper under double-blind review

ABSTRACT

Polanyi's paradox—that we can know more than we can tell—motivates a question for spatial decision-making in large language models (LLMs): can spatial information recovered from their internal representations support more accurate spatial predictions than the models' final choices? We investigate this question in a route-choice task requiring the integration of separate travel experiences to select the next move yielding the shortest route to a destination. Using 115,362 task inputs, we trace how spatial information recovered from hidden states relates to model choices across layers, and use causal interventions to test how intermediate computations influence output preferences. The results show that spatial connectivity and distance are partially recoverable from hidden states, with more accurate recovery of start-centered distances than global pairwise distances under the respective probing setups. Route-comparison accuracy exhibits a sharp increase at intermediate layers, suggesting an emergence-like transition in the accessibility of goal-relevant spatial information, consistent with the integration of fragmented spatial experiences. Route preferences obtained from hidden states are more accurate than the final choices in every evaluated model, revealing a discrepancy between internally recoverable spatial information and behavior. Causal interventions suggest that opposing component effects and their downstream changes may contribute to output-preference instability across layers. Together, these findings reveal a Polanyi-like representation–decision gap: spatial information recoverable through supervised readouts is not fully reflected in the models' choices. Our results distinguish the emergence of decodable spatial information from its reliable behavioral expression and provide a framework for diagnosing failures in LLM spatial decision-making.

1 INTRODUCTION

Goal-directed spatial decision-making in unfamiliar environments often requires integrating spatial information acquired across separate experiences (Ishikawa & Montello, 2006). Research on human navigation suggests that people organize such experiences into knowledge of relationships between places to support flexible, goal-directed choices (Javadi et al., 2017; Behrens et al., 2018; Crivelli-Decker et al., 2023; Epstein et al., 2017). Yet being able to draw on knowledge does not necessarily mean being able to fully articulate it—a distinction captured by Polanyi's paradox: "we can know more than we can tell"(Polanyi, 1966; Autor, 2014). This perspective motivates a related question for large language model (LLM)-based agents: to what extent do their goal-directed choices reflect the spatial information acquired from experience? Answering this question requires looking beyond an LLM's final choices and the explanations it generates, which do not, by themselves, reveal how spatial experience informs its decisions (Turpin et al., 2023; Lanham et al., 2023). An incorrect choice may arise because relevant spatial relations are not accurately represented, or because the model fails to use information already available within its computation (Lepori et al., 2026). Conversely, a correct choice does not establish which information the model relied on (McCoy et al., 2019).

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

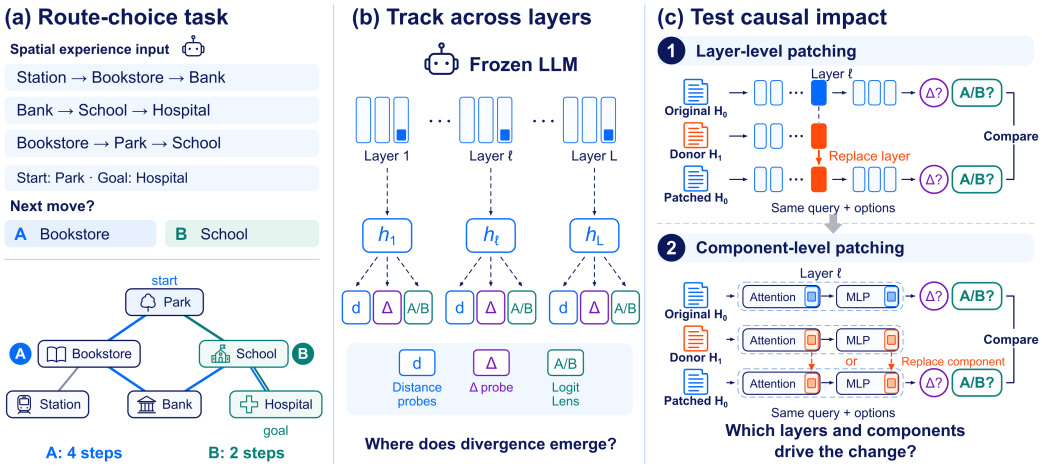

Figure 1: **Overview of our analysis of spatial decision-making.** (a) Given spatial experiences, a starting location, and a destination, the model chooses between two candidate next moves according to which yields a shorter route. The map and route lengths illustrate the task and are not provided to the model. (b) At each layer, distance probes, a Δ probe, and Logit Lens are applied independently to the same decision-token hidden state. (c) The patching schematics illustrate replacement of either the full decision-token state at a selected layer or an attention or MLP output at that token, one component per run. Activations are transferred from a matched donor run (H_1) to the original run (H_0), keeping the query and options fixed. Original and patched runs are compared through Δ predictions and A/B output preferences.

Recent studies show that LLMs can represent structures introduced through context(Park et al., 2025; Tehenan et al., 2025; Feng & Steinhardt, 2024; Han et al., 2025) and that interventions on related internal states can alter subsequent predictions (Hendel et al., 2023; Todd et al., 2024; Dai et al., 2024; Kowalyshyn et al., 2026). Other findings reveal that contextually encoded information is not always used effectively in downstream tasks (Lepori et al., 2026; An et al., 2026). These findings leave open how the relationship between internal knowledge and behavior evolves during spatial decision-making. It remains unclear which computations strengthen or weaken the correspondence between spatial information and behavioral preferences, and whether their effects persist to the final choice. Explaining these dynamics would connect evidence of spatial representation to a functional account of how spatial knowledge supports behavior.

We investigate these questions through a route-choice task(Figure 1), in which models receive spatial experiences and choose between two candidate next moves according to which yields a shorter route to the destination. We use linear probes to examine how accurately spatial relations and goal-dependent route comparisons can be recovered from hidden states across layers. We relate the inferred route preferences to the models' layerwise output preferences and final choices. We then intervene on layer states and individual attention and multilayer perceptron(MLP) components to test their causal effects on output preferences and whether these effects persist through subsequent computation.

Our main contributions are as follows:

- **A diagnostic framework for spatial decision-making.** We introduce a framework for examining how spatial information acquired from experience relates to goal-directed choices in LLMs. The framework jointly examines spatial relations, route comparisons, and output preferences within the same decision process, combining behavioral evaluation and hidden-state analyses with causal interventions.
- **Spatial representations and behavioral discrepancies.** Across Qwen3.5 and Gemma 3 models, hidden states support partial recovery of connectivity and distances, with lower probe errors for start-centered than global distances. Route-comparison accuracy rises sharply in intermediate layers, potentially reflecting cognitive emergence through the in-

tegration of spatial experiences, while agreement with output preferences strengthens later. In every evaluated model, route predictions inferred from hidden states are more accurate than final choices, revealing a consistent representation–decision gap.

- **Causal analysis of spatial decision-making.** Interventions across matched spatial histories reveal component effects in opposing directions that can weaken or reverse during subsequent computation. These findings suggest that opposing component influences and their downstream modification may contribute to the observed fluctuations in output preferences across layers.

2 Related Work

Spatial Cognition in Humans and LLMs. Human navigation involves learning relationships between places and using this knowledge to plan actions toward a goal(Javadi et al., 2017; Crivelli-Decker et al., 2023). Cognitive map accounts describe how relational knowledge can support flexible behavior (Behrens et al., 2018). This connection between spatial knowledge and its use provides a cognitive perspective for evaluating LLMs. Benchmarks such as SPARTQA and StepGame evaluate spatial question answering and multi-hop relational composition (Mirzaee et al., 2021; Shi et al., 2022). Text-based evaluations examine whether models can infer spatial structure from movement and observation sequences (Yamada et al., 2024) and use textual walkthroughs to answer mapping and navigation questions (Ding et al., 2024). Focusing on goal-directed planning, CogEval draws on human cognitive experiments and reveals systematic failures such as invalid trajectories and loops (Momennejad et al., 2023). Theory of Space extends the assessment to how spatial knowledge is acquired and revised during exploration, comparing active exploration with passive observation and evaluating both downstream performance and prompted cognitive maps(Zhang et al., 2026). These studies assess the acquisition and use of spatial knowledge through models' answers, plans, and generated maps.

Spatial Representations in LLMs. Internal analyses provide evidence that language models encode spatial structure and, in some settings, use it to guide predictions. Gurnee & Tegmark (2024) recover geographical coordinates from the hidden states of pretrained LLMs. Tehenan et al. (2025) examine linearly decodable spatial relations and their influence on model outputs. In a model trained specifically on Othello move sequences, Li et al. (2023) identify board-state representations and show that modifying them changes subsequent move predictions. Studies of task-trained transformers further identify map-like and path-dependent navigation strategies (Baumgartner et al., 2025). More recent work examines spatial representations during goal-directed task execution. Arghal et al. (2026) decode environmental states and action plans in a fully observable grid world and assess whether the agent's actions are consistent with its decoded maps. Using tasks motivated by human spatial cognition, An et al. (2026) report task-dependent spatial representations that can influence behavior but remain weakly integrated into final predictions. These studies motivate combining probing, which assesses information recoverability (Alain & Bengio, 2016; Belinkov, 2022), with interventions, which test the effects of changing internal computation (Vig et al., 2020; Geiger et al., 2021; Zhang & Nanda, 2024).

In-Context Representation Learning. In-context representation learning examines how LLMs form representations of new structure supplied in their inputs without updating model parameters. Park et al. (2025) show that random-walk traces can reorganize concept representations in pretrained LLMs to reflect graph structure specified in context. Kowalyshyn et al. (2026) further show that activation patching and steering of graph-related states can shift subsequent predictions, providing evidence that context-induced structure can influence model behavior. In task-trained transformers, Fang & Rajan (2025) identify intermediate computations cached in memory tokens and accessed at decision time. However, forming such representations does not guarantee their reliable use. Lepori et al. (2026) find that models struggle to apply novel structures encoded from context in next-token prediction and adaptive world-modeling tasks. Beyond spatial reasoning, latent-knowledge elicitation studies show that information recovered from activations can be more accurate than models' overt answers under the evaluated conditions (Burns et al., 2023; Mallen et al., 2024; Orgad et al., 2025; Gekhman et al., 2025). These findings establish both the capacity to represent contextual structure and limitations in applying it.

Our work connects research on spatial representations with the broader problem of using spatial knowledge acquired in context. We approach goal-directed spatial decision-making from the perspective of the representation–behavior gap, focusing on the relationship between experience-derived spatial knowledge and its expression in final choices.

3 Problem Formulation

Environment and Spatial Experiences A spatial environment consists of locations and the connections between them. A spatial experience describes a sequence of visited locations, providing partial information about how places are connected. Multiple experiences can be linked through shared locations, allowing relationships between places encountered in different experiences to be inferred. In our setting, the LLM receives textual descriptions of these experiences, followed by a spatial decision question specifying a goal and candidate actions, as illustrated in Figure 1(a).

Route-Choice Task To examine how spatial experiences inform goal-directed decisions, we use a route-choice task. Each query specifies a starting location, a destination, and two neighboring locations labeled A and B. The model is asked to select its next move so as to reach the destination in the fewest total steps, using only connections established by the experiences. Each query is constructed to have a unique correct answer.

Dataset We construct environments with 6, 8, 10, 12, 14, or 16 locations (3, 4, 5, 6, 7, or 8 experiences, respectively). Each experience describes two consecutive moves through three distinct locations, with shared locations linking different experiences. For each selected route-choice setting, we construct eight matched variants combining two histories, two destinations, and two option orderings. The destinations favor different candidates under the original history. Exchanging their names throughout the experiences produces a counterfactual history that reverses the optimal choice for each destination while keeping the question and options unchanged. Swapping the option order changes the A/B labels without changing the preferred physical location. These variants support comparisons across changes in spatial information, goals, and answer-label assignments. The resulting dataset contains 19,364 distinct experience histories and 115,362 task input records.

Models We evaluate Qwen3.5 models with 4B, 9B, and 27B parameters, and Gemma 3 IT models with 4B, 12B, and 27B parameters[1]. All models remain frozen and receive the same English spatial task through their native chat templates. Qwen3.5 models are evaluated in non-thinking mode. We instruct all models to answer only A or B and use greedy decoding.

4 From Spatial Representations to Decisions

4.1 Spatial Information in Hidden States

Global spatial relations recovery We first examine whether the spatial relations established by the experiences can be recovered from the model's hidden states. At each layer, we train linear probes to predict shortest-path distances between all pairs of locations, using the hidden state at the final input token before answer generation. For input i, let $\mathcal{P}_i$ denote all unordered pairs of distinct locations appearing in the experiences. The road set $E_i \subseteq \mathcal{P}_i$ contains the pairs directly connected by roads established in the experiences. For each pair (u, v), the distance $d_i(u, v)$ is the minimum number of roads connecting the two locations using connections in E_i. Let $\hat{d}_i^{(\ell)}(u, v)$ denote the distance probe's prediction at layer ℓ, and let $\hat{E}_i^{(\ell)}$ denote the road set predicted by the adjacency probe at the same layer. We evaluate distance predictions using MAE and adjacency predictions using Jaccard similarity:

$$\mathrm{MAE}_i^{(\ell)} = \frac{1}{|\mathcal{P}_i|} \sum_{(u,v)\in\mathcal{P}_i} \left| \hat{d}_i^{(\ell)}(u, v) - d_i(u, v) \right|, \qquad \mathrm{Jaccard}_i^{(\ell)} = \frac{|\hat{E}_i^{(\ell)} \cap E_i|}{|\hat{E}_i^{(\ell)} \cup E_i|}. \tag{1}$$

[1] In preliminary compatibility checks, Gemma 4 E2B, 12B, and 31B did not consistently produce A or B as the first generated token under the same answer-format instruction. We therefore use Gemma 3 for comparisons across model sizes.

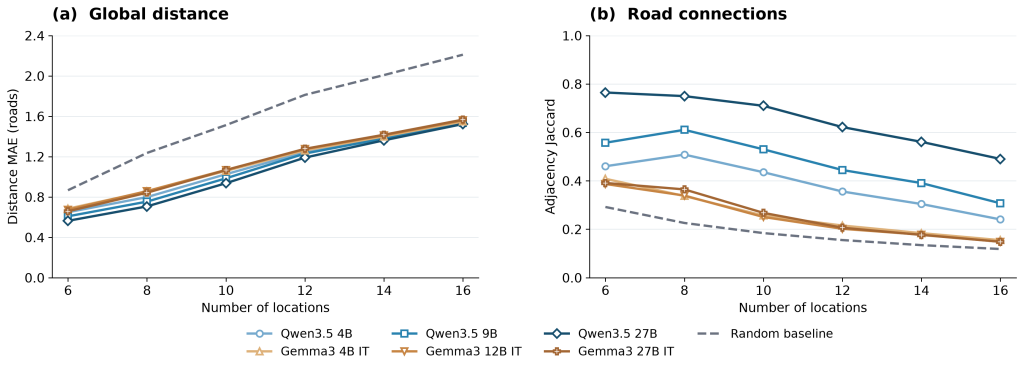

Figure 2: **Recovery of global spatial relations.** (a) Distance prediction, evaluated using mean absolute error (MAE). (b) Adjacency prediction, evaluated using Jaccard similarity. Each point shows the lowest MAE or highest Jaccard similarity observed across layers for a given model and environment size. Lower MAE and higher Jaccard similarity indicate better recovery.

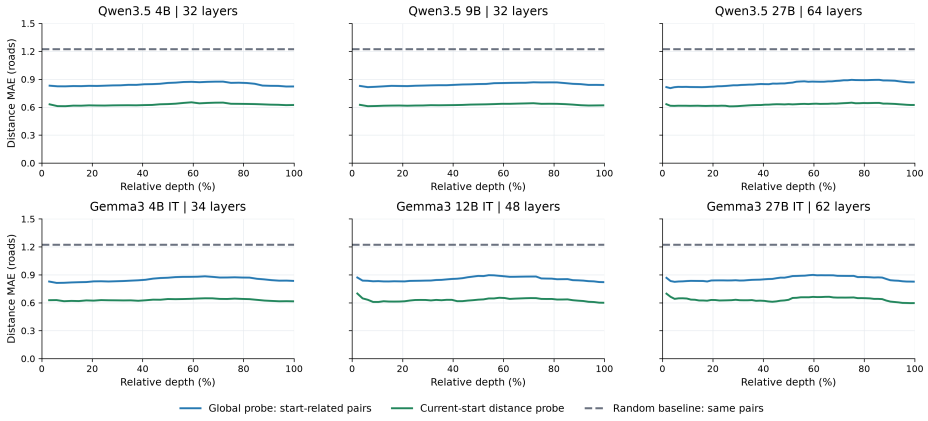

Figure 3: **Recovery of start-centered distances.** Layerwise MAE of the start-centered probe and global probe, evaluated on identical start-to-other location pairs and averaged equally across six environment sizes. Gray dashed lines show a training-distribution random baseline. Lower MAE indicates better recovery.

As shown in Figure 2, global spatial recovery generally declines as the number of locations increases. For Qwen3.5-27B, distance MAE rises from 0.563 at 6 locations to 1.523 at 16 locations, while adjacency Jaccard decreases from 0.765 to 0.490. Across the environment sizes, Qwen3.5 models achieve lower distance errors and higher adjacency Jaccard scores than Gemma 3 models. Within the Qwen3.5 family, larger models improve both metrics, with particularly clear differences in adjacency recovery: at 16 locations, Jaccard increases from 0.240 for 4B to 0.307 for 9B and 0.490 for 27B, compared with 0.148–0.155 across the Gemma 3 models. These results indicate that hidden states retain recoverable information about global spatial relations, although recover remains incomplete and becomes less accurate in larger environments.

Start-centered distances The global probes treat distances between fixed pairs of locations as separate prediction targets. Human navigation research has identified neurons that encode directions and distances to environmental reference points relative to the navigating individual (Kunz et al., 2021). In Othello models, defining piece identities relative to the current player reveals linear representations of board states that are less apparent under fixed color labels(Nanda et al., 2023).

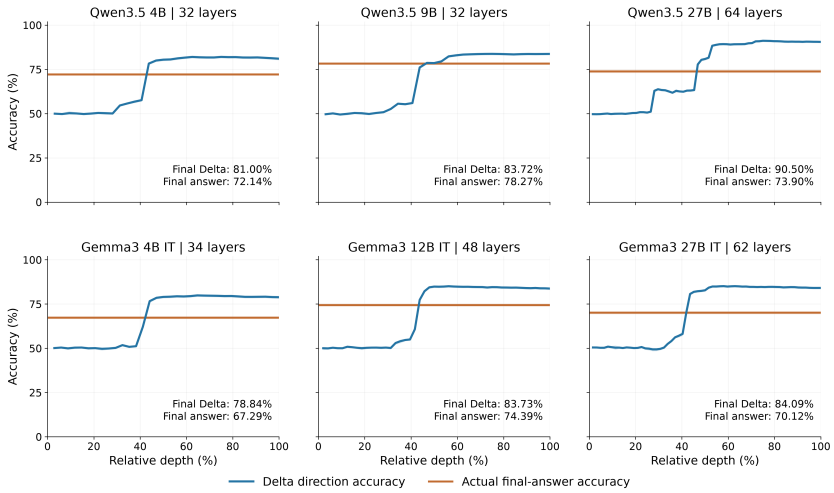

Figure 4: **Route comparison accuracy across layers.** Delta direction accuracy at each layer, compared with the accuracy of the model's actual final answers. The horizontal axes indicate relative network depth. Shaded regions indicate pointwise 95% confidence intervals estimated via bootstrap resampling.

Inspired by these findings, we examine whether organizing distance predictions around the query's starting location improves their linear recovery from hidden states.

We train start-centered probes to predict distances from the query's starting location to every other location, using the same hidden states as the global probes. Figure 3 compares their layerwise MAE, averaged equally across the six environment sizes. Start-centered probes yield lower prediction errors at every layer in all models. These results show that distances anchored at the current start can be recovered with substantially lower errors than the full set of pairwise distances under the respective probing setups.

4.2 Route Comparisons and Final Choices

Route Comparison Accuracy. We next examine whether the relative costs of the two candidate routes can be predicted from hidden states. For input i, let a_i and b_i denote the locations assigned to options A and B, respectively, and let g_i denote the goal. Since both candidates are one move from the starting location, their identical initial costs cancel, giving the route-cost difference:

$$\Delta_i = d_i(b_i, g_i) - d_i(a_i, g_i). \tag{2}$$

Positive values favor A, while negative values favor B. At each layer, we train a linear probe to predict Δ_i directly from the hidden state at the final input token, using the same grouped cross-validation protocol as the distance probes. Let $\hat{\Delta}_i^{(\ell)}$ denote its prediction at layer ℓ. We evaluate direction accuracy, counting a prediction as correct when $\hat{\Delta}_i^{(\ell)}\Delta_i > 0$, and compare it with the accuracy of the model's actual final answers.

Figure 4 shows a sharp increase in route-comparison accuracy in intermediate layers across all six models. In the curves averaged equally across the six environment sizes, the largest adjacent-layer improvement occurs at approximately 42–47% of network depth. This sharp transition marks a substantial improvement in the accuracy with which goal-dependent route comparisons can be recovered from hidden states. At the final layer, probe accuracy exceeds final-answer accuracy in every model, with gaps ranging from 5.4 to 16.6 percentage points.

Agreement with Output Preferences. We further examine the alignment between the route preferences predicted by the Delta probe and the model's output preferences across layers. At each layer, we apply the model's native final normalization and output head to the hidden state at the final input

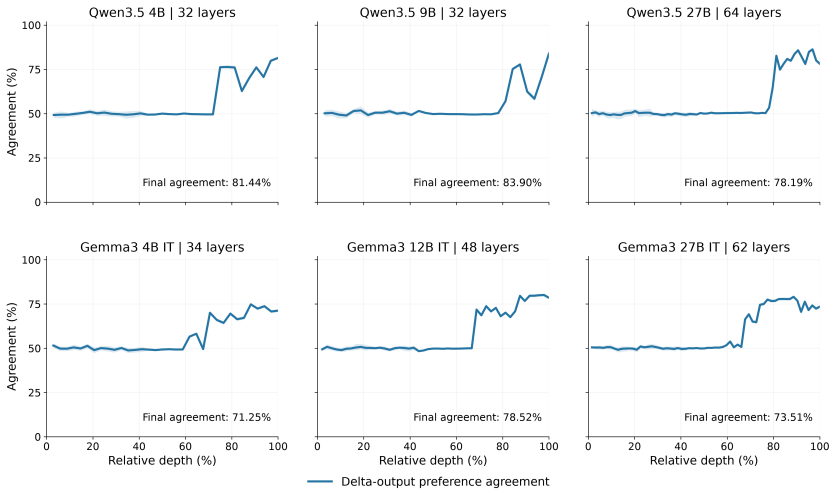

Figure 5: **Agreement with output preferences across layers.** Agreement between the Delta probe's preferred option and the preference obtained through Logit Lens for the same input at the same layer. The horizontal axes indicate relative network depth. Shaded regions indicate pointwise 95% confidence intervals estimated via bootstrap resampling.

token (nostalgebraist, 2020). Let $z_i^{(\ell)}(A)$ and $z_i^{(\ell)}(B)$ denote the resulting logits for A and B. We define the output margin as

$$m_i^{(\ell)} = z_i^{(\ell)}(A) - z_i^{(\ell)}(B). \tag{3}$$

Positive values favor A, while negative values favor B. For the same input and layer, the two preferences agree when $\hat{\Delta}_i^{(\ell)} m_i^{(\ell)} > 0$. We compute agreement among non-tied inputs within each environment group and then average equally across groups. This metric measures consistency between the two preferences: agreement can occur when both favor the incorrect option.

Figure 5 shows that agreement with output preferences increases substantially at later layers. The largest adjacent-layer increase in the size-averaged curves occurs at approximately 68–84% of network depth, later than the corresponding increase in route-comparison accuracy at 42–47%. Final-layer agreement ranges from 71.2% to 83.9%, indicating incomplete alignment between the two preferences. The later-layer trajectories are also non-monotonic: Qwen3.5-27B, for example, reaches a peak agreement of 86.3% before declining to 78.2% at the final layer. The sharp improvement in route-comparison accuracy therefore precedes a later, fluctuating increase in alignment with output preferences.

Layer-level interventions. We first examine whether replacing late-layer hidden states changes output preferences. For each model, we select the layer immediately preceding the largest adjacent-layer increase in Delta–Logit Lens agreement, averaged equally across the six environment sizes. We use matched inputs with the same start, goal, and option ordering, but different experience histories that reverse the correct route choice. At the selected layer, we replace the recipient's hidden state at the final input token with the donor's corresponding state, retain the other token states, and resume normal computation.

We measure output changes using the donor's original final-layer Delta preference as a fixed reference. Let $m_i^{(\ell)}$ and $m_{i,\mathrm{swap}}^{(\ell)}$ denote the original and intervened recipient's A/B logit margins at layer ℓ, obtained through Logit Lens. Let $\hat{\Delta}_{d,i}^{(L)}$ denote the donor's original Delta prediction at the final layer L. The oriented effect is

$$\mathrm{CE}_i^{(\ell)} = \mathrm{sign}\left(\hat{\Delta}_{d,i}^{(L)}\right)\left(m_{i,\mathrm{swap}}^{(\ell)} - m_i^{(\ell)}\right). \tag{4}$$

Positive values indicate increased output support for the option favored by the donor's original Delta prediction; negative values indicate reduced support. We use this definition for both whole-state and

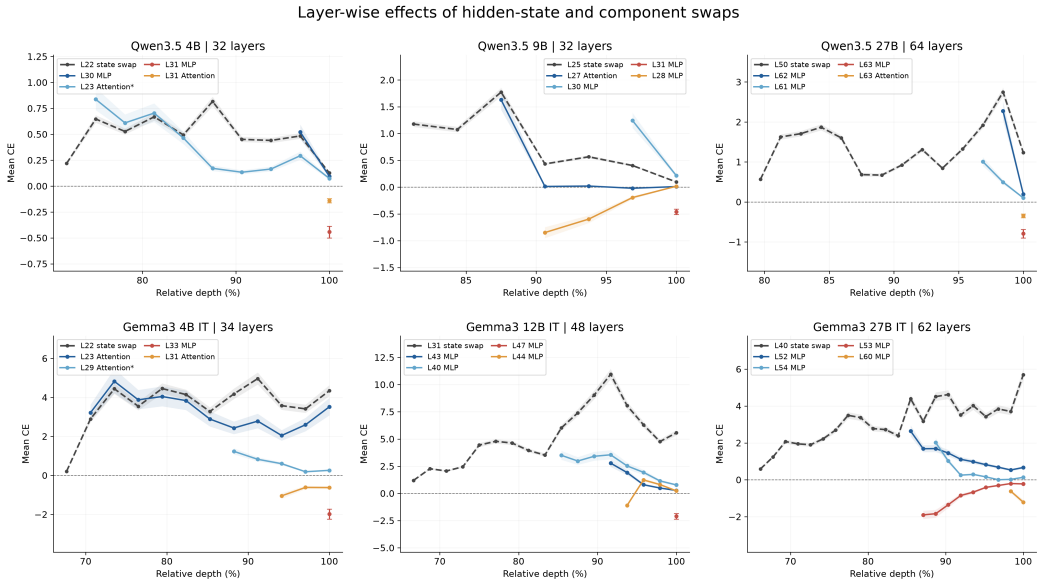

Figure 6: **Layerwise effects of hidden-state and component swaps.** Changes in A/B logit margins are oriented by the donor's original final-layer Delta preference. Gray dashed curves show hidden-state swaps; blue and red/orange curves show swaps of components screened as support and competition candidates, respectively. Shading and error bars indicate pointwise 95% bootstrap confidence intervals.

component replacements. Results are averaged equally across environment groups within each size and then across sizes.

The gray dashed curves in Figure 6 show positive mean final effects in every evaluated model. The effects nevertheless evolve non-monotonically across layers. Between the first downstream layer and the final layer, the mean effect decreases in the evaluated Qwen3.5 models and increases in the Gemma 3 models. For example, it decreases from 1.075 to 0.096 in Qwen3.5-9B and increases from 1.238 to 5.701 in Gemma 3-27B. Thus, transferring a hidden state influences final output preferences, while subsequent computation substantially modifies the effect.

Component-level interventions. We next examine the effects of individual attention and MLP branches. We screen components by projecting their original residual contributions onto the A/B output direction under the original final normalization scale. Based on these contributions relative to the original final-layer Delta preferences, we select two supporting and two opposing candidates per model. For each model, we select the two components with the largest positive and the two with the most negative mean projected contributions relative to the original final-layer Delta preference, termed supporting and competing candidates, respectively. For each candidate, we replace its residual contribution at the final input token with the corresponding contribution from the original donor run and resume normal computation. Each intervention replaces only one component. These experiments use the same 960 inputs per model. Using Equation 4, we track effects from the output of the intervened block to the final layer.

The colored curves in Figure 6 show that mean final effects generally follow the screening categories: supporting candidates produce positive effects, whereas most opposing candidates produce negative effects. Their persistence varies across components. For Qwen3.5-27B, replacing the layer-62 MLP produces a mean effect of $+2.272$ at that layer, which attenuates to $+0.194$ at the final layer. These results identify individual branches that influence output preferences, while showing that their local effects may weaken, persist, or reverse during subsequent computation.

5 DISCUSSION

What Kind of Spatial Representation Is Recovered? Our probes recover partial information about environmental connectivity and distance, with lower prediction errors for start-centered distance targets. This difference raises the possibility that the organization of spatial information matters for its accessibility in a particular task. Human navigation research highlights the processing of network topology (Javadi et al., 2017), spatial relations relative to the navigating individual (Kunz et al., 2021), and representations shaped by context and goals (Crivelli-Decker et al., 2023). These findings provide a conceptual perspective on the task-related organization of spatial knowledge. Our results also complement evidence for graph representations learned in context (Park et al., 2025) and imperfect spatial maps elicited from foundation models (Zhang et al., 2026). They support the recoverability of partial, task-relevant topological information, while leaving open whether the model maintains a unified cognitive map.

Discrepancies Between Probe Predictions and Model Choices Delta probe predictions outperform final choices across the evaluated models, while inferred route preferences do not always align with output preferences obtained through Logit Lens. Route-comparison accuracy rises sharply in intermediate layers, whereas agreement with output preferences strengthens later. This sharp increase may reflect cognitive emergence through the integration of spatial experiences. Related work shows that internal signals can distinguish correct answers even when models favor incorrect responses (Orgad et al., 2025; Gekhman et al., 2025; Lepori et al., 2026; Wang & Xu, 2026). Our findings characterize a related discrepancy in goal-directed choices based on spatial experiences supplied in context. Conceptually, prior findings and our results suggest a phenomenon in LLMs analogous to Polanyi's paradox(Polanyi, 1966): information recoverable from their internal states is not always reflected in their outputs.

Fluctuations in Output Preferences Output preferences can repeatedly switch between the two options across late layers. Changes in output preference also occur for inputs whose Delta predictions retain the same sign, indicating that answer preferences can change without a corresponding change in the inferred route preference. One possible explanation is that successive components exert opposing influences on the A/B logits. Our component analyses and interventions reveal effects in both directions, suggesting that an emerging preference can be reinforced by some components and counteracted by others. Related work on the Hydra effect and self-repair documents downstream compensation following component ablation(McGrath et al., 2023; Rushing & Nanda, 2024).

6 CONCLUSION

We presented a diagnostic framework for examining how spatial experiences relate to goal-directed choices in LLMs. Across the Qwen3.5 and Gemma 3 families, spatial connectivity and distance are partially recoverable from hidden states, with more accurate recovery of start-centered distances than global pairwise distances under the respective probing setups. Route-comparison accuracy rises sharply in intermediate layers. This transition may reflect cognitive emergence through the integration of spatial experiences, while agreement with output preferences strengthens later and remains incomplete. In every evaluated model, route preferences inferred from hidden states are more accurate than final choices. Interventions reveal component effects in opposing directions that can weaken or reverse during subsequent computation, suggesting a possible contribution to fluctuations in output preferences across layers. Conceptually, these findings characterize a Polanyi-like representation–decision gap in LLM spatial decision-making: spatial information recoverable from internal states is not fully reflected in the models' final choices. Future work should test whether this recoverable information can guide more accurate choices and whether the observed patterns extend to more complex environments.

AI USE STATEMENT

In this work, we used generative AI tools to help develop conceptual frameworks, refine research hypotheses, design and provide feedback on research methodology and experiments, implement methods, assist with data analysis and interpretation of results, and assist with translation. We have not used generative AI tools to generate synthetic datasets or clean and reformat datasets,

and the remaining required disclosure tasks are not applicable to this work. Additionally, we used generative AI tools for brainstorming, literature search and summarization, paper structuring, and drafting and editing text. We have reviewed all AI-assisted work: research ideas and literature were checked against original sources, AI-assisted code and analyses were manually reviewed and tested, and interpretations were verified against the experimental results. We take responsibility for the final content of this work, including text, claims, analyses, and artifacts produced with the aid of generative AI.

Reproducibility statement

We provide anonymized code, data, and data-generation scripts in our anonymous repository, together with the experimental details required to reproduce our results.

References

Guillaume Alain and Yoshua Bengio. Understanding intermediate layers using linear classifier probes. *arXiv preprint arXiv:1610.01644*, 2016. URL https://arxiv.org/abs/1610.01644.

Jiyuan An, Liner Yang, Mengyan Wang, Luming Lu, Weihua An, and Erhong Yang. From human cognition to neural activations: Probing the computational primitives of spatial reasoning in LLMs. *arXiv preprint arXiv:2603.26323*, 2026. URL https://arxiv.org/abs/2603.26323.

Raghu Arghal, Fade Chen, Niall Dalton, Evgenii Kortukov, Calum McNamara, Angelos Nalmpantis, Moksh Nirvaan, Gabriele Sarti, and Mario Giulianelli. A behavioural and representational evaluation of goal-directedness in language model agents. In *Proceedings of the 43rd International Conference on Machine Learning*, 2026. URL https://arxiv.org/abs/2602.08964.

David Autor. Polanyi's paradox and the shape of employment growth. Working Paper 20485, National Bureau of Economic Research, September 2014. URL https://www.nber.org/papers/w20485.

Caroline Baumgartner, Eleanor Spens, Neil Burgess, and Petru Manescu. Cognitive maps in language models: A mechanistic analysis of spatial planning. In *NeurIPS 2025 Workshop on CogInterp: Interpreting Cognition in Deep Learning Models*, 2025. URL https://openreview.net/forum?id=lz14CoUBkq.

Timothy E. J. Behrens, Timothy H. Muller, James C. R. Whittington, Shirley Mark, Alon B. Baram, Kimberly L. Stachenfeld, and Zeb Kurth-Nelson. What is a cognitive map? organizing knowledge for flexible behavior. *Neuron*, 100(2):490–509, 2018. doi: 10.1016/j.neuron.2018.10.002. URL https://doi.org/10.1016/j.neuron.2018.10.002.

Yonatan Belinkov. Probing classifiers: Promises, shortcomings, and advances. *Computational Linguistics*, 48(1):207–219, 2022. doi: 10.1162/coli_a_00422. URL https://aclanthology.org/2022.cl-1.7/.

Collin Burns, Haotian Ye, Dan Klein, and Jacob Steinhardt. Discovering latent knowledge in language models without supervision. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://arxiv.org/abs/2212.03827.

Jordan Crivelli-Decker, Alex Clarke, Seongmin A. Park, Derek J. Huffman, Erie D. Boorman, and Charan Ranganath. Goal-oriented representations in the human hippocampus during planning and navigation. *Nature Communications*, 14:2946, 2023. doi: 10.1038/s41467-023-35967-6. URL https://www.nature.com/articles/s41467-023-35967-6.

Qin Dai, Benjamin Heinzerling, and Kentaro Inui. Representational analysis of binding in language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 17468–17493. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.emnlp-main.967. URL https://aclanthology.org/2024.emnlp-main.967/.

Peng Ding, Jiading Fang, Peng Li, Kangrui Wang, Xiaochen Zhou, Mo Yu, Jing Li, Matthew R. Walter, and Hongyuan Mei. MANGO: A benchmark for evaluating mapping and navigation abilities of large language models. In *Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=6vEfyp0o68.

Russell A. Epstein, Eva Zita Patai, Joshua B. Julian, and Hugo J. Spiers. The cognitive map in humans: Spatial navigation and beyond. *Nature Neuroscience*, 20(11):1504–1513, 2017. doi: 10.1038/nn.4656. URL https://www.nature.com/articles/nn.4656.

Ching Fang and Kanaka Rajan. From memories to maps: Mechanisms of in-context reinforcement learning in transformers. *arXiv preprint arXiv:2506.19686*, 2025. URL https://arxiv.org/abs/2506.19686.

Jiahai Feng and Jacob Steinhardt. How do language models bind entities in context? In *The Twelfth International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/9d1b7fc578c0d2d6431fc26d736ecaf3-Abstract-Conference.html.

Atticus Geiger, Hanson Lu, Thomas Icard, and Christopher Potts. Causal abstractions of neural networks. In *Advances in Neural Information Processing Systems*, volume 34, pp. 9574–9586. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper/2021/hash/4f5c422f4d49a5a807eda27434231040-Abstract.html.

Zorik Gekhman, Eyal Ben David, Hadas Orgad, Eran Ofek, Yonatan Belinkov, Idan Szpektor, Jonathan Herzig, and Roi Reichart. Inside-out: Hidden factual knowledge in LLMs. In *Second Conference on Language Modeling*, 2025. URL https://openreview.net/forum?id=f7GG1MbsSM.

Wes Gurnee and Max Tegmark. Language models represent space and time. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/0a6059857ae5c82ea9726ee9282a7145-Abstract-Conference.html.

Seungwook Han, Jinyeop Song, Jeff Gore, and Pulkit Agrawal. Emergence and effectiveness of task vectors in in-context learning: An encoder decoder perspective. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 21871–21897. PMLR, 2025. URL https://proceedings.mlr.press/v267/han25h.html.

Roee Hendel, Mor Geva, and Amir Globerson. In-context learning creates task vectors. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 9318–9333. Association for Computational Linguistics, 2023. doi: 10.18653/v1/2023.findings-emnlp.624. URL https://aclanthology.org/2023.findings-emnlp.624/.

Toru Ishikawa and Daniel R. Montello. Spatial knowledge acquisition from direct experience in the environment: Individual differences in the development of metric knowledge and the integration of separately learned places. *Cognitive Psychology*, 52(2):93–129, 2006. doi: 10.1016/j.cogpsych.2005.08.003. URL https://doi.org/10.1016/j.cogpsych.2005.08.003.

Amir-Homayoun Javadi, Beatrix Emo, Lorelei R. Howard, Fiona E. Zisch, Yichao Yu, Rebecca Knight, Joao Pinelo Silva, and Hugo J. Spiers. Hippocampal and prefrontal processing of network topology to simulate the future. *Nature Communications*, 8:14652, 2017. doi: 10.1038/ncomms14652. URL https://www.nature.com/articles/ncomms14652.

Katharine Kowalyshyn, Timothy Duggan, Daniel Little, and Michael C. Hughes. Belief or circuitry? causal evidence for in-context graph learning. *arXiv preprint arXiv:2605.08405*, 2026. doi: 10.48550/arXiv.2605.08405. URL https://arxiv.org/abs/2605.08405.

Lukas Kunz, Armin Brandt, Peter C. Reinacher, Bernhard P. Staresina, Eric T. Reifenstein, Christoph T. Weidemann, Nora A. Herweg, Ansh Patel, Melina Tsitsiklis, Richard Kempter, Michael J. Kahana, Andreas Schulze-Bonhage, and Joshua Jacobs. A neural code for egocentric spatial maps in the human medial temporal lobe. *Neuron*, 109(17):2781–2796.e10, 2021. doi: 10.1016/j.neuron.2021.06.019.

Tamera Lanham, Anna Chen, Ansh Radhakrishnan, Benoit Steiner, Carson Denison, Danny Hernandez, Dustin Li, Esin Durmus, Evan Hubinger, Jackson Kernion, Kamilė Lukošiūtė, Karina Nguyen, Newton Cheng, Nicholas Joseph, Nicholas Schiefer, Oliver Rausch, Robin Larson, Sam McCandlish, Sandipan Kundu, Saurav Kadavath, Shannon Yang, Thomas Henighan, Timothy Maxwell, Timothy Telleen-Lawton, Tristan Hume, Zac Hatfield-Dodds, Jared Kaplan, Jan Brauner, Samuel R. Bowman, and Ethan Perez. Measuring faithfulness in chain-of-thought reasoning. *arXiv preprint arXiv:2307.13702*, 2023. URL https://arxiv.org/abs/2307.13702.

Michael A. Lepori, Tal Linzen, Ann Yuan, and Katja Filippova. Language models struggle to use representations learned in-context. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 14841–14857. Association for Computational Linguistics, 2026. doi: 10.18653/v1/2026.acl-long.676. URL https://aclanthology.org/2026.acl-long.676/.

Kenneth Li, Aspen K. Hopkins, David Bau, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. Emergent world representations: Exploring a sequence model trained on a synthetic task. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=DeG07_TcZvT.

Alex Mallen, Madeline Brumley, Julia Kharchenko, and Nora Belrose. Eliciting latent knowledge from quirky language models. In *Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=nGCMLATBit.

R. Thomas McCoy, Ellie Pavlick, and Tal Linzen. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 3428–3448. Association for Computational Linguistics, 2019. doi: 10.18653/v1/P19-1334. URL https://aclanthology.org/P19-1334/.

Thomas McGrath, Matthew Rahtz, Janos Kramar, Vladimir Mikulik, and Shane Legg. The Hydra effect: Emergent self-repair in language model computations. *arXiv preprint arXiv:2307.15771*, 2023. doi: 10.48550/arXiv.2307.15771. URL https://arxiv.org/abs/2307.15771.

Roshanak Mirzaee, Hossein Rajaby Faghihi, Qiang Ning, and Parisa Kordjamshidi. SPARTQA: A textual question answering benchmark for spatial reasoning. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 4582–4598. Association for Computational Linguistics, 2021. doi: 10.18653/v1/2021.naacl-main.364. URL https://aclanthology.org/2021.naacl-main.364/.

Ida Momennejad, Hosein Hasanbeig, Felipe Vieira Frujeri, Hiteshi Sharma, Nebojsa Jojic, Hamid Palangi, Robert Ness, and Jonathan Larson. Evaluating cognitive maps and planning in large language models with CogEval. In *Advances in Neural Information Processing Systems*, volume 36, pp. 69736–69751, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/dc9d5dcf3e86b83e137bad367227c8ca-Abstract-Conference.html.

Neel Nanda, Andrew Lee, and Martin Wattenberg. Emergent linear representations in world models of self-supervised sequence models. In *Proceedings of the 6th BlackboxNLP Workshop: Analyzing and Interpreting Neural Networks for NLP*, pp. 16–30, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.blackboxnlp-1.2. URL https://aclanthology.org/2023.blackboxnlp-1.2/.

nostalgebraist. Interpreting GPT: The logit lens. LessWrong, August 2020. URL https://www.lesswrong.com/posts/AcKRB8wDpdaN6v6ru/interpreting-gpt-the-logit-lens. Technical blog post.

Hadas Orgad, Michael Toker, Zorik Gekhman, Roi Reichart, Idan Szpektor, Hadas Kotek, and Yonatan Belinkov. LLMs know more than they show: On the intrinsic representation of LLM hallucinations. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/a712d461e57201efe35d429a6f1731c1-Abstract-Conference.html.

Core Francisco Park, Andrew Lee, Ekdeep Singh Lubana, Yongyi Yang, Maya Okawa, Kento Nishi, Martin Wattenberg, and Hidenori Tanaka. ICLR: In-context learning of representations. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/83fe5a77502e3d4cfab5960aed0ee6c3-Abstract-Conference.html.

Michael Polanyi. *The Tacit Dimension*. Doubleday, Garden City, NY, 1966.

Cody Rushing and Neel Nanda. Explorations of self-repair in language models. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 42836–42855. PMLR, 2024. URL https://proceedings.mlr.press/v235/rushing24a.html.

Zhengxiang Shi, Qiang Zhang, and Aldo Lipani. StepGame: A new benchmark for robust multi-hop spatial reasoning in texts. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 11321–11329, 2022. doi: 10.1609/aaai.v36i10.21383. URL https://ojs.aaai.org/index.php/AAAI/article/view/21383.

Matthieu Tehenan, Christian Bolivar Moya, Tenghai Long, and Guang Lin. Linear spatial world models emerge in large language models. *arXiv preprint arXiv:2506.02996*, 2025. URL https://arxiv.org/abs/2506.02996.

Eric Todd, Millicent L. Li, Arnab Sen Sharma, Aaron Mueller, Byron C. Wallace, and David Bau. Function vectors in large language models. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/4ae163cb8788970e53b4fd9578141139-Abstract-Conference.html.

Miles Turpin, Julian Michael, Ethan Perez, and Samuel R. Bowman. Language models don't always say what they think: Unfaithful explanations in chain-of-thought prompting. In *Advances in Neural Information Processing Systems*, volume 36, pp. 74952–74965. Curran Associates, Inc., 2023. doi: 10.52202/075280-3275. URL https://papers.nips.cc/paper/2023/hash/ed3fea9033a80fea1376299fa7863f4a-Abstract-Conference.html.

Jesse Vig, Sebastian Gehrmann, Yonatan Belinkov, Sharon Qian, Daniel Nevo, Yaron Singer, and Stuart Shieber. Investigating gender bias in language models using causal mediation analysis. In *Advances in Neural Information Processing Systems*, volume 33, pp. 12388–12401. Curran Associates, Inc., 2020. URL https://papers.nips.cc/paper/2020/hash/92650b2e92217715fe312e6fa7b90d82-Abstract.html.

Zeyu Wang and Xinming Xu. Knowing isn't always saying: When do spatial encodings reach answers in vision-language models? *arXiv preprint arXiv:2608.22916*, 2026. URL https://arxiv.org/abs/2608.22916. Accepted to EMNLP 2026 Main Conference.

Yutaro Yamada, Yihan Bao, Andrew K. Lampinen, Jungo Kasai, and Ilker Yildirim. Evaluating spatial understanding of large language models. *Transactions on Machine Learning Research*, 2024. URL https://openreview.net/forum?id=xkiflfKCw3.

Fred Zhang and Neel Nanda. Towards best practices of activation patching in language models: Metrics and methods. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/06a52a54c8ee03cd86771136bc91eb1f-Abstract-Conference.html.

Pingyue Zhang, Zihan Huang, Yue Wang, Jieyu Zhang, Letian Xue, Zihan Wang, Qineng Wang, Keshigeyan Chandrasegaran, Ruohan Zhang, Yejin Choi, Ranjay Krishna, Jiajun Wu, Fei-Fei Li, and Manling Li. Theory of space: Can foundation models construct spatial beliefs through active exploration? In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/dd9c5ce8803e1898d438e636fbae0236-Abstract-Conference.html.

A Dataset Construction and Task Examples

A.1 Environment and Experience Generation

We construct environments containing 6, 8, 10, 12, 14, or 16 locations. Each environment has the same number of distinct roads as locations. Roads are undirected and have unit traversal cost. The resulting graph is connected and contains exactly one cycle, potentially with branching paths attached to it. This graph provides the ground truth for evaluation; the model receives only textual descriptions of travel experiences. Locations are drawn from a fixed, ordered vocabulary: Hospital, School, Park, Bookstore, Bank, Station, Pharmacy, Library, Supermarket, Post Office, Museum, Cinema, Bakery, Hotel, Restaurant, and Fire Station. An environment with N locations uses the first N names.

For six locations, the source generator enumerates connected graphs with six roads and samples from this set. For larger environments, it proposes sets of N roads between named locations and retains proposals that cover all locations and form a connected graph. Each size has a source pool of 2,000 base histories and 20 additional histories generated for preliminary implementation checks. The base histories are balanced with respect to whether Hospital and School share a direct road. This is a source-generation constraint, rather than the prediction target of the route-choice task. All source histories satisfying the route-query constraints below are eligible for the final dataset.

We partition each graph's roads into paths of two consecutive moves using randomized backtracking. Each path becomes one experience containing three distinct locations. Every road appears in exactly one experience, while locations can recur across experiences. We then randomize experience order and independently reverse the presentation direction of each experience. Thus, the six environment sizes contain 3, 4, 5, 6, 7, and 8 experiences, respectively. Identical names refer to the same location throughout a history. Consecutive locations within an experience establish a road; boundaries between experiences establish no additional roads. Location count and experience count increase together, while individual experience length remains fixed. Comparisons across sizes therefore concern their combined variation.

A.2 Route-Query Construction

For each source history, we enumerate eligible starting locations, candidate moves, and destinations. A starting location must have exactly two neighbors, which become the two candidate next moves. The candidates must have equal degree in the experienced graph and equal mention counts in the history. A destination must differ from the start and both candidates. Its shortest distance from the start must be two, three, or four steps, and it must not appear in the same experience as the start. Consequently, the required route cannot be obtained from a single experience. We retain only queries with a unique optimal candidate. The cost of a candidate consists of the initial move from the start to that candidate, followed by the shortest remaining route to the destination. Shortest paths are computed using the roads established by the experiences. The remaining route may revisit previously visited locations, including the original start.

To construct matched variants, we identify pairs of destinations that favor opposite candidates under the original history. The two destinations must also have equal degree and equal mention counts. We enumerate all combinations satisfying these requirements, without selecting queries according to model responses. Ground-truth roads, distances, route costs, and answer labels are stored separately from the model-facing inputs.

A.3 Matched History, Goal, and Option Variants

Each matched query group contains two histories, two destinations, and two option orderings, yielding eight inputs(Figure 7). Each model input contains only one history, one destination, and one option ordering. The eight variants are evaluated as separate inputs.

History variants. The original history is denoted by H_0. We construct a counterfactual history, H_1, by exchanging the names of the two selected destinations throughout the experiences. The start and candidate names remain unchanged. Because the destinations are matched in degree and mention count, this exchange preserves these statistics for every named location, as well as the

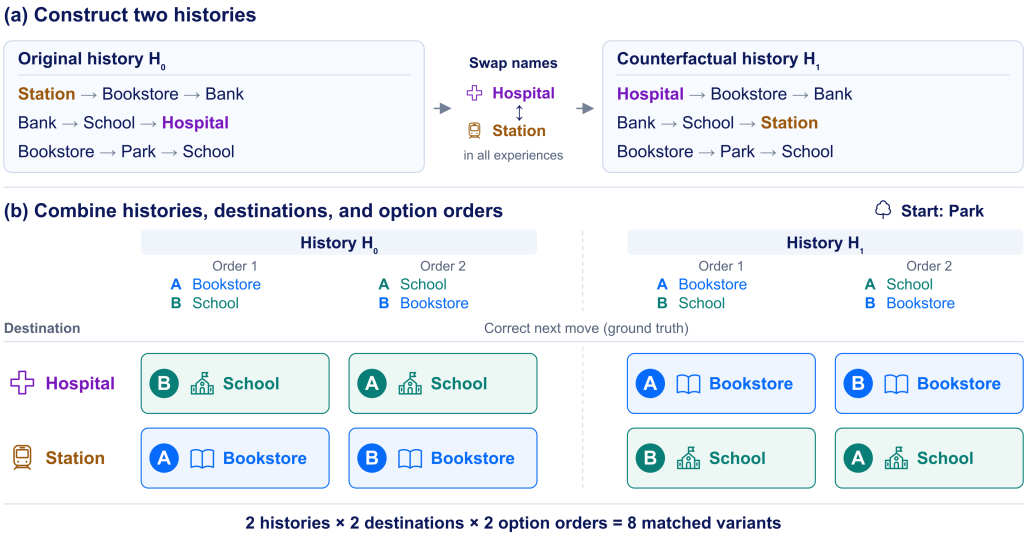

Figure 7: **Construction of eight matched route-choice variants.** (a) The counterfactual history H_1 is constructed by exchanging Hospital and Station throughout the original history H_0. (b) Each history is combined with two destinations and two option orders, yielding eight matched variants with a fixed starting location and candidate locations. Each cell shows the correct next location and its answer label; colors consistently identify locations across option orders. For the same query, changing the history reverses the optimal next location. Swapping the option order changes the correct answer label without changing that location. The displayed answers indicate task ground truth, not model predictions, and are not provided to the model.

unlabeled graph topology. For a fixed question and option ordering, the history exchange reverses the optimal candidate. It can change several named road connections simultaneously; it is not restricted to editing a single road.

Goal variants. Within each history, we ask separate questions about the two selected destinations while holding the start and candidate locations fixed. The preferred physical candidate differs between the two goals.

Option-order variants. For every history–goal combination, we include both assignments of the candidate locations to labels A and B. Swapping the assignments reverses the correct answer label while preserving the preferred physical location.

A.4 Dataset Composition and Evaluation Groups

The final dataset includes all inputs referenced by at least one eligible matched query group. Inputs shared by multiple groups from the same source are stored once. Across the six sizes, 3,957 source histories contribute eligible queries. Including counterfactual variants, the dataset contains 19,364 distinct experience histories, 16,792 matched query groups, and 115,362 input records. The correct A and B labels are equally frequent at every size. Table 1 reports the counts.

We define evaluation groups separately at each environment size. For six locations, the initial grouping uses a Weisfeiler–Lehman graph hash that assigns Hospital and School a shared anchor role and all other locations a second role. For larger environments, the hash ignores location names and groups graphs by structural signatures. Hash collisions conservatively merge groups. We subsequently merge groups connected by counterfactual transformations or identical named graphs. All inputs derived from a source, including its history, goal, and option-order variants, remain together during cross-validation. These groups, rather than individual query variants, define the units used to prevent overlap between probe training and evaluation data. The six-location grouping preserves the

Locations	Experiences	Histories	Query groups	Input records	Eval. groups
6	3	329	202	1,616	8
8	4	1,256	831	6,356	22
10	5	2,529	2,046	14,676	183
12	6	3,494	3,045	21,092	525
14	7	5,377	4,825	32,578	889
16	8	6,379	5,843	39,044	964
Total	–	19,364	16,792	115,362	2,591

Table 1: Dataset composition. Experiences are counted per history; histories include original and counterfactual versions. Query groups contain eight matched variants and may share inputs. Evaluation groups collect related environments and their variants for grouped evaluation.

designated anchor roles and should not be interpreted as a complete holdout of all unlabeled graph topologies. Probe fitting and statistical procedures are described in Appendix C.

A.5 A Complete Six-Location Example

The following example, illustrated in Figure 7, is taken from the generated dataset. Its original history, H_0, contains three experiences:

> Experience 1: Station → Bookstore → Bank
> Experience 2: Bank → School → Hospital
> Experience 3: Bookstore → Park → School

The starting location is Park, the candidate moves are Bookstore and School, and the two destinations are Hospital and Station. Exchanging Hospital and Station throughout the experiences produces H_1:

> Experience 1: Hospital → Bookstore → Bank
> Experience 2: Bank → School → Station
> Experience 3: Bookstore → Park → School

For example, one query following H_0 is:

> Using only roads supported by these experiences, you are at Park and want to reach Hospital in the fewest total steps. Which place should you move to first?
> A. Bookstore
> B. School
> Answer only A or B.

Under H_0, moving first to School yields the two-step route Park → School → Hospital. Moving first to Bookstore requires four steps, so the correct answer is B. Under H_1, the same question instead favors Bookstore, through Park → Bookstore → Hospital, and the correct answer becomes A.

Table 2 lists all eight variants. Route costs include the initial move. The costs and correct answers are evaluator annotations and are not included in model inputs.

B Models, Prompts, and Activation Extraction

B.1 Models and Computational Setup

We evaluate six publicly available conversational models from the Qwen3.5 and Gemma 3 families. Table 3 lists their checkpoint identifiers and text-decoder configurations. All models receive text-only inputs, and their parameters remain fixed throughout the experiments. Linear probes are trained separately, as described in Appendix C.

Inference uses PyTorch 2.8.0 with its CUDA 12.8 build and Transformers 5.14.1 on NVIDIA H100 80GB GPUs. We load the native `Qwen3_5ForConditionalGeneration` and

History	Goal	Option A	Option B	Cost A	Cost B	Answer
H_0	Hospital	Bookstore	School	4	2	B
H_0	Hospital	School	Bookstore	2	4	A
H_0	Station	Bookstore	School	2	4	A
H_0	Station	School	Bookstore	4	2	B
H_1	Hospital	Bookstore	School	2	4	A
H_1	Hospital	School	Bookstore	4	2	B
H_1	Station	Bookstore	School	4	2	B
H_1	Station	School	Bookstore	2	4	A

Table 2: Eight matched variants of a six-location example. All queries start at Park. Each row is evaluated as a separate model input.

Checkpoint	Decoder blocks	Hidden dimension
`Qwen/Qwen3.5-4B`	32	2,560
`Qwen/Qwen3.5-9B`	32	4,096
`Qwen/Qwen3.5-27B`	64	5,120
`google/gemma-3-4b-it`	34	2,560
`google/gemma-3-12b-it`	48	3,840
`google/gemma-3-27b-it`	62	5,376

Table 3: Model checkpoints and text-decoder configurations. Layer indices throughout the paper are zero-based.

`Gemma3ForConditionalGeneration` implementations with BF16 weights and the SDPA attention backend. Models run in evaluation mode under `torch.inference_mode()`. The checkpoint configurations, tokenizer files, software versions, and execution settings are recorded in the run manifests.

B.2 Prompt Construction

Every input consists of a single user message containing the environment rules, the experience history, and one route-choice question. No additional system message or solved demonstration is supplied. All models receive identical user-message text for a given input, rendered through their own tokenizers and native chat templates.

The complete user-message template is:

Route-choice prompt

Below are independent travel experiences in the same virtual town.
Identical names refer to the same place. Consecutive visits within an experience provide evidence for a bidirectional road of one step.
No roads are implied between separate experiences.

Experience 1: [place] → [place] → [place]
. . .
Experience [number]: [place] → [place] → [place]

Using only roads supported by these experiences, you are at [start]
and want to reach [destination] in the fewest total steps.
Which place should you move to first?
A. [candidate assigned to A]
B. [candidate assigned to B]
Answer only A or B.

Bracketed fields and the ellipsis are placeholders and do not appear in actual inputs. Each experience occupies one line, and all experiences in the selected history are included. We apply the native chat template with `add_generation_prompt=True`. For Qwen3.5, we additionally set

`enable_thinking=False`. Its rendered generation prefix contains an empty thinking block, followed by the position from which the answer is predicted:

```
<|im_start|>user
[task prompt]<|im_end|>
<|im_start|>assistant
<think>

</think>
```

The Gemma 3 template renders the same user message as:

```
<bos><start_of_turn>user
[task prompt]<end_of_turn>
<start_of_turn>model
```

The empty Qwen thinking block is supplied by the template. No reasoning tokens are generated before answer evaluation. For every input and tokenizer, we verify that appending either A or B to the generation prefix adds exactly one token without changing the preceding tokenization.

B.3 Inference and Answer Extraction

We perform a complete forward pass over each rendered input with `use_cache=False` and `logits_to_keep=1`. The model's answer is the highest-scoring next token over the entire vocabulary. We decode this token and remove surrounding whitespace. An answer is valid only when the remaining text is exactly A or B.

Behavioral accuracy counts a valid answer as correct when it matches the task label. Invalid outputs are retained and counted as incorrect. We also save the raw generated-token identity, the A and B logits, and their probabilities normalized over the full vocabulary. Consequently, the relative A/B preference can be examined separately from the model's unrestricted answer.

Inputs are grouped by token length for batching, without padding or truncation. The original activation collection uses batch size one for the three Qwen3.5 models and Gemma 3 12B and 27B. Gemma 3 4B uses batches of two equally long inputs, with smaller batches where required at the end of a group. Batch configurations are accepted only after numerical comparison with reference executions.

B.4 Activation Position and Layer Indexing

We analyze activations at the `decision` position, defined as the final token of the fully rendered input, including the native assistant-generation prefix, immediately before the first answer token is predicted. This position is determined separately for each tokenized input and need not coincide with the end of the user message. No generated answer token is included in the extracted states.

We record the residual-stream output at this position after every decoder block using forward hooks. A state indexed by layer ℓ is the output after block ℓ, including that block's residual updates and before the model's final output normalization. The embedding output is not counted as a decoder layer.

Activations are computed in the model's native execution and copied to FP32 arrays for storage. This conversion preserves the captured values without performing an additional FP32 model forward pass. Stored activations are indexed by input, decoder block, token position, and hidden dimension.

B.5 Native Logit Lens and Numerical Validation

For Logit Lens, we apply the model's existing final normalization and output projection to each saved `decision` state. We reuse the original modules and BF16 weights without fitting an additional transformation. The implementation follows the output interface of the exact model class used for activation collection. To reproduce the original computation, the offline evaluation preserves the

source batch size, sequence length, target-row position, and output-head input layout. The saved state is placed at its original position in a tensor of the corresponding shape before final normalization. Because the normalization and output projection operate independently on token rows, unused rows can be filled with zeros. Preserving their shape avoids numerical differences introduced by evaluating an isolated token under a different computational layout.

We validate activation collection by comparing forward passes with and without recording hooks. These checks require logits to agree with absolute and relative tolerances of 10^{-5}, together with identical full-vocabulary greedy predictions. Saved activation arrays are also checked for finite values, correct dimensions, and exact equality after serialization.

For Logit Lens, interface checks compare the final-layer projection with the complete model's vocabulary logits. Across all formal inputs, the final-layer A/B logits, full-vocabulary-normalized A/B probabilities, and unrestricted greedy token are checked against the original inference records. The numerical tolerances remain 10^{-5}, and greedy-token agreement must be exact. Intermediate-layer Logit Lens values describe the preferences obtained by applying the final output interface to an earlier hidden state. At the last layer, the same procedure reproduces the model's actual output interface.

C Probe Training and Statistical Evaluation

C.1 Prediction Targets

We train separate linear probes for each model, environment size, and decoder layer. The main analyses use the `decision` state defined in Appendix B. Each probe receives a single hidden-state vector, without concatenating layers. All spatial targets are computed from the roads established by the input's own experience history. Counterfactual inputs therefore use their corresponding counterfactual graphs. Targets are independent of the model's answer.

Global spatial relations. For each unordered pair of distinct locations, we train an independent output head to predict its shortest-path distance. The adjacency analysis uses a separate head for the binary target indicating whether that pair shares a direct road. Heads are fitted independently, even when their computations share a matrix factorization.

Start-centered distances. We train one output head for each fixed destination location. For input i, the head assigned to location v predicts $d_i(s_i, v)$, where s_i is the query's starting location and $d_i(u, v)$ is the shortest-path distance between locations u and v in that input's experienced graph.

Route comparisons. A separate scalar probe predicts $\Delta_i = d_i(b_i, g_i) - d_i(a_i, g_i)$, where a_i and b_i are the locations assigned to A and B, and g_i is the destination. Positive values favor A and negative values favor B. This probe predicts the difference directly from the hidden state; its prediction is not obtained by subtracting the outputs of the global distance probes.

C.2 Weighted Ridge Regression

All probes use weighted ridge regression. For one output head, let $\mathcal{T}$ denote its eligible fitting inputs, y_i its target, z_i the standardized hidden-state vector, and w_i the fitting weight. We solve

$$\min_{\beta,b} \sum_{i\in\mathcal{T}} w_i\big(y_i - \beta^\top z_i - b\big)^2 + \alpha\|\beta\|_2^2, \tag{5}$$

where β is the coefficient vector, b is an unpenalized intercept, and α controls regularization.

For each fit, feature means and standard deviations are computed using only the fitting inputs and their weights. Standard deviations below 10^{-10} are replaced by one. After standardization, features are divided by the square root of the hidden dimension. Target centering is also computed exclusively from the fitting inputs. Base weights give equal total weight to each evaluation group, equal weight to distinct histories within a group, and equal weight to inputs sharing a history. They are normalized to have mean one over the eligible fitting records. For adjacency heads, positive and negative targets

are additionally balanced to have equal total weight, followed by the same mean-one normalization. Distance and Delta probes do not use class balancing.

We solve the regression problems in FP64. Where identical states are compressed for computation, their original weights are summed and their targets are aggregated with those weights. This preserves the least-squares problem and the effective regularization strength. Batched implementations are checked against reference computations using absolute and relative tolerances of 10^{-7}, with identical selected regularization parameters.

C.3 Nested Grouped Cross-Validation

We use five outer folds and three inner folds, constructed separately at each environment size with seed 20260914. The evaluation groups defined in Appendix A remain intact. Consequently, related histories and their goal and option-order variants cannot be divided between fitting and evaluation sets. Fold construction balances input counts without splitting groups and does not use model answers or activations. The same fold assignments are reused across models and probe types. For start-centered heads, self-distance inputs are removed within these existing folds without reallocating the remaining inputs.

Within each outer training set, we select $\alpha \in \{0.001, 0.1, 10\}$ separately for each output head. Distance and Delta probes use weighted mean absolute error for inner validation; adjacency probes use weighted mean squared error. Although validation criteria differ, all heads are fitted with the squared-error objective in Equation 5. After selecting α, we refit the head on the complete outer training set and predict the held-out outer fold. Repeating this process yields one out-of-fold prediction for every eligible input–target combination. Neither fitting nor hyperparameter selection uses that input's evaluation group.

C.4 Random Baselines

We compare spatial recovery with random predictions generated without using language-model activations.

Distance. For each environment size and outer cross-validation fold, we estimate an empirical distance distribution from the training inputs. The global baseline pools distances across all unordered pairs of distinct locations, with equal weight to inputs and pairs within each input. For the start-centered comparison, we instead pool distances from each input's starting location to all other locations, excluding self-distances. Each held-out target is predicted by an independent draw from the corresponding training distribution, without conditioning on location identity. The start-centered baseline is evaluated on the same start-to-location pairs as both probes. Held-out labels are used only for scoring.

Adjacency. For each input, every unordered pair of distinct locations is independently predicted to share a road with probability 0.5. We evaluate the resulting road set using Jaccard similarity with the true road set. Distance and adjacency predictions are generated separately; random distances are not computed from the randomly predicted roads.

Sampling and aggregation. We perform 10,000 repetitions and report the mean baseline performance. Within each repetition, we compute MAE or Jaccard for each input and average inputs equally within each environment size. Summaries across sizes give each of the six sizes equal weight. The baselines are independent of model and layer, and no layer selection is applied.

C.5 Metrics and Matched Comparisons

Distance predictions remain continuous when computing MAE; we do not round them to integer distances. For global recovery, each input's MAE averages over all unordered pairs of distinct locations. Adjacency scores are thresholded at 0.5, and recovery is evaluated using the Jaccard similarity between the predicted and true road sets. The adjacency regression scores are not interpreted as calibrated probabilities.

For the start-centered comparison, we extract the global probe's predictions for exactly the same start-to-location pairs evaluated by the start-centered probe. Both methods are therefore scored on identical inputs and target pairs at each layer, with self-distances excluded. For Delta direction accuracy, a prediction $\hat{\Delta}_i$ is correct only when $\hat{\Delta}_i \Delta_i > 0$. A prediction of exactly zero is recorded as a tie and does not count as correct. All task targets have nonzero true Delta.

For agreement with output preferences, we compare the sign of the Delta prediction with the sign of the same-layer A-minus-B Logit Lens margin. If either value is exactly zero, the input is recorded as a tie and excluded from the conditional agreement denominator. Agreement is independent of ground-truth correctness: two preferences can agree while both favor the wrong answer.

C.6 Aggregation and Uncertainty

Training weights and reporting weights serve different purposes. We retain the aggregation used by each analysis and distinguish the following summaries.

Global spatial summaries. The cross-model global comparison averages input-level metrics equally within each environment size and then averages the six sizes equally. The original grouped probe reports additionally provide summaries with equal evaluation-group weight, equal history weight within each group, and equal input weight within each history.

Start-centered comparisons. The matched distance curves give equal weight to evaluation groups, to observed starting locations within each group, to histories within each group–start combination, and to inputs sharing that history and start. The same weights are applied to both probes.

Route comparisons and output agreement. Route comparison accuracy and actual-answer accuracy first average inputs equally within each evaluation group and then average groups equally. Conditional agreement is computed among non-tied inputs within each group, followed by equal averaging over groups with a defined agreement rate. When combining environment sizes, each size receives weight $1/6$.

Confidence intervals are obtained by resampling entire evaluation groups with replacement and recomputing the reported statistic. The original graph-probe summaries use 1,000 bootstrap replicates; supplementary within-input start-relative analyses use 2,000; the route-comparison and Delta–Logit Lens agreement curves use 10,000. Intervals are the 2.5th and 97.5th bootstrap percentiles.

For the accuracy gap between Delta predictions and actual answers, both quantities use the same resampled groups. Cross-size summaries resample groups separately within each size before averaging sizes equally. Undefined conditional metrics retain their support counts and are not replaced by zero.

These are pointwise intervals conditional on the saved out-of-fold predictions. Probes are not retrained during bootstrap resampling. The intervals do not account for selection across layers or multiple comparisons. Cross-validation folds are not treated as independent experimental replicates.

D Intervention Protocols and Validation

D.1 Matched Whole-State Interventions

We construct donor–recipient pairs from matched inputs that differ in their spatial experiences while preserving the starting location, destination, candidate moves, and A/B assignment. The history change reverses the objectively preferred option. Both intervention directions are retained, together with both destinations and both option orders. The intervention set contains 67,168 matched history pairs across the six environment sizes, yielding 134,336 directed donor–recipient configurations per model and intervention layer. Inputs are included regardless of their original answer correctness or probe predictions.

For a recipient input, we replace the hidden state at the decision token after block k with the corresponding state from the donor's unmodified forward pass. All other token states at the intervention boundary retain their recipient values, and subsequent computation proceeds normally. Each run

Model	Intervention block
Qwen3.5-4B	22
Qwen3.5-9B	25
Qwen3.5-27B	50
Gemma3-4B	22
Gemma3-12B	31
Gemma3-27B	40

Table 4: Late-layer whole-state intervention sites. All indices are zero-based. Each listed layer denotes the block whose output is replaced.

modifies a single layer. In particular, interventions at different layers are evaluated separately rather than applied sequentially within one forward pass.

For each model, we select the block immediately preceding the largest adjacent-layer increase in Delta–Logit Lens agreement, averaging equally across environment sizes. Table 4 lists these late-layer sites. Layer selection uses the original observational curves and is therefore exploratory.

D.2 Measuring Downstream Changes

Let O, D, and P denote the original recipient, natural donor, and patched recipient, respectively. We apply the frozen, layer-specific out-of-fold Delta probes to their hidden states. The donor and recipient share an evaluation group and therefore use the same held-out probe at a given layer. No probe is fitted to intervened activations.

For downstream layer ℓ, we measure the absolute change from the original recipient's Delta prediction:

$$\left|\hat{\Delta}_P^{(\ell)} - \hat{\Delta}_O^{(\ell)}\right|.$$

The intervention layer itself is used to verify replacement identity; downstream changes are evaluated at $\ell > k$.

For output effects, let $m_X^{(\ell)} = z_X^{(\ell)}(A) - z_X^{(\ell)}(B)$ be the native Logit Lens margin for condition X at layer ℓ. Let L denote the final block index. Following Equation 4, we orient the margin change using the donor's original final-layer Delta preference:

$$E_{\text{state}}^{(\ell)} = \operatorname{sign}\left(\hat{\Delta}_D^{(L)}\right)\left(m_P^{(\ell)} - m_O^{(\ell)}\right).$$

This reference is fixed across downstream layers and is also used for component replacements. Positive values indicate increased output support for the donor's original Delta preference; they do not necessarily indicate an answer change or improved correctness on the recipient task.

D.3 Component Selection and Intervention Sample

We select attention and MLP branches using their descriptive contributions to the original A/B output margin. Contributions are computed using the original final normalization scale and output projection, then oriented by the sign of the original final-layer Delta prediction. This statistic describes how a branch's write projects onto the output preference under a fixed normalization scale.

Within each environment size, evaluation groups are assigned deterministically to an 80% screening partition and a remaining internal-check partition. A support candidate must have a positive mean contribution and a positive contribution on more than half of the inputs in every environment-size-by-option-order cell. Competition candidates satisfy the corresponding negative conditions. Candidates are ranked by their role-oriented mean contribution, weighting the six environment sizes equally. We retain the first two candidates per role for each model (Table 5).

The internal-check results do not alter candidate selection. Qwen3.5-4B L23 attention and Gemma3-4B L29 attention satisfy the screening conditions in 10 of 12 and 11 of 12 internal-check cells, respectively; these candidates remain included. Since the original data had already been explored, this partition assesses internal stability rather than providing independent confirmation.

Component interventions use a previously fixed sample of 960 inputs per model, with identical input identities across all six models. For each environment size, we select 20 complete sets of

Model	Support candidates	Competition candidates
Qwen3.5-4B	L30 MLP, L23 attention	L31 MLP, L31 attention
Qwen3.5-9B	L27 attention, L30 MLP	L31 MLP, L28 MLP
Qwen3.5-27B	L62 MLP, L61 MLP	L63 MLP, L63 attention
Gemma3-4B	L23 attention, L29 attention	L33 MLP, L31 attention
Gemma3-12B	L43 MLP, L40 MLP	L47 MLP, L44 MLP
Gemma3-27B	L52 MLP, L54 MLP	L53 MLP, L60 MLP

Table 5: Components selected for individual natural counterfactual replacement. Support and competition refer to the observational screening statistic. Layer indices are zero-based.

eight matched variants, giving 160 inputs. Selection prioritizes evaluation groups and route strata with fewer previously selected sets, with deterministic hash-based tie breaking. It does not use model answers, probe predictions, or intervention outcomes. The six-location sample covers eight evaluation groups; each larger environment size covers 20 evaluation groups.

D.4 Natural Counterfactual Component Replacements

We replace each selected component's effective residual write with the corresponding write from a natural opposite-history input. The donor and recipient have the same start, destination, candidate moves, and A/B assignment, and differ only in the matched H_0/H_1 history. The donor vector comes from its unmodified forward pass.

Each run replaces one attention or MLP branch at the decision token. Other token positions retain their recipient values at the intervention boundary, and all subsequent operations are recomputed normally. For Gemma3, replacement is applied after the branch-specific normalization, where the branch output enters the residual stream. The affected-block output and all subsequent decision-token states are recorded. For an attention replacement, the affected-block output already includes the response of the MLP within that block.

The experiment reuses the four fixed candidates and the same 960 inputs per model without reselection. Each input is paired with its opposite-history counterpart, giving 480 unordered pairs and 960 directed replacements per component. Both directions are already included in this count. Across six models and four components, there are 23,040 replacement configurations.

D.5 Component Effects and Pretreatment Groups

Counterfactual replacement effects. Let S denote a single-component replacement and D its natural donor. As for whole-state replacement, the primary effect is

$$E^{(\ell)}_{\text{swap}} = \text{sign}\left(\hat{\Delta}^{(L)}_D\right)\left(m^{(\ell)}_S - m^{(\ell)}_O\right).$$

The donor's original final-layer Delta sign remains fixed throughout the intervened computation. We separately examine effects oriented by the original final-layer Delta difference, $\text{sign}(\hat{\Delta}^{(L)}_D - \hat{\Delta}^{(L)}_O)$, and by the natural final-output difference, $\text{sign}(m^{(L)}_D - m^{(L)}_O)$. The subset where these two directions conflict is defined before intervention.

We retain groups with opposite original Delta preferences, the same nonzero preference, or a Delta tie at either endpoint. Recipient and donor agreement with their respective original outputs is also determined before intervention. Same-preference pairs can reveal changes in preference strength without testing transfer between opposite preferences.

Donor alignment is the proportion of inputs whose output margin supports the fixed original donor Delta preference, among inputs with a nonzero donor reference. Exact output ties remain in this denominator and are separately recorded. Strict preference flips require a change between opposite nonzero A/B margins. Transitions into or out of ties, changes in the actual greedy answer, and correctness on the recipient task are reported separately. A zero directional reference is left undefined, rather than assigned a zero effect.

Model	Intervention layer k	First downstream layer $k+1$ Mean CE (95% interval)	Final layer Mean CE (95% interval)
Qwen3.5-4B	22	$+0.647$ $[+0.611, +0.684]$	$+0.126$ $[+0.119, +0.134]$
Qwen3.5-9B	25	$+1.075$ $[+1.027, +1.126]$	$+0.096$ $[+0.091, +0.101]$
Qwen3.5-27B	50	$+1.625$ $[+1.560, +1.691]$	$+1.236$ $[+1.181, +1.293]$
Gemma3-4B	22	$+2.896$ $[+2.729, +3.071]$	$+4.346$ $[+4.072, +4.630]$
Gemma3-12B	31	$+2.272$ $[+2.187, +2.360]$	$+5.568$ $[+5.335, +5.814]$
Gemma3-27B	40	$+1.238$ $[+1.174, +1.301]$	$+5.701$ $[+5.386, +6.010]$

Table 6: Whole-state replacement effects at the first downstream layer and the final layer. CE uses the donor's original final-layer Delta preference as a fixed directional reference. Each model uses the full set of 134,336 directed configurations. Results use equal weights for evaluation groups within each environment size and for the six sizes; brackets show pointwise 95% group-bootstrap intervals. Layer indices are zero-based.

D.6 Aggregation, Uncertainty, and Numerical Validation

For whole-state interventions, we average directed configurations within each evaluation group and then weight groups equally. Because an input may participate in multiple matched pairs, original and patched outcomes are compared on the same directed configurations. The whole-state curves use the full matched-pair set, whereas component interventions use the fixed 960-input sample. For component interventions, each input receives its original within-scale evaluation-group weight. These weights are preserved when reporting pretreatment subgroups: subgroup estimates normalize the retained weights, rather than assigning equal weight anew to the groups represented in that subgroup. Groups with no subgroup members remain part of the bootstrap sampling population. Results combined across environment sizes weight each size equally. If a subgroup has no defined directional observations at one size, its six-size mean is reported as undefined rather than silently dropping that size.

We compute pointwise 95% percentile intervals using 10,000 evaluation-group bootstrap resamples. Paired comparisons reuse the same group draws. Combined-scale estimates resample independently within each scale. These intervals describe uncertainty for the specified analyses; they do not correct for exploratory layer or component selection. Raw logit-effect magnitudes are interpreted within each model.

Before accepting intervention outputs, we verify unmodified replay, self-replacement, unchanged upstream states, intervention identity, and native output calculations. Native activation and logit comparisons use absolute and relative tolerances of 10^{-5}; frozen probe reproduction uses 10^{-7}. Unmodified and self-replacement runs must preserve the original greedy answer. Computation retains the validated source precision and batch layout, and no model or probe parameters are updated. For counterfactual component replacements, the inserted vector must exactly match the saved natural-donor write. We additionally verify the donor–recipient mapping in both directions and the identity of original baselines across components.

E Additional Intervention Results

This appendix reports additional results for whole-state and component replacements across the six models and six environment sizes. Intervention protocols and statistical procedures follow Appendix D.

Model	Component	Final donor-oriented effect (95% interval)	Donor alignment change (pp)
Qwen3.5-4B	L30 MLP	$+0.094$ $[+0.081, +0.108]$	$+6.56$
	L23 attention	$+0.074$ $[+0.064, +0.085]$	$+4.70$
	L31 MLP	-0.442 $[-0.501, -0.387]$	-10.05
	L31 attention	-0.139 $[-0.160, -0.120]$	-4.68
Qwen3.5-9B	L27 attention	$+0.010$ $[+0.005, +0.015]$	$+0.00$
	L30 MLP	$+0.214$ $[+0.189, +0.239]$	$+5.89$
	L31 MLP	-0.460 $[-0.510, -0.411]$	-6.09
	L28 MLP	$+0.017$ $[+0.011, +0.022]$	$+0.52$
Qwen3.5-27B	L62 MLP	$+0.194$ $[+0.170, +0.219]$	$+2.58$
	L61 MLP	$+0.100$ $[+0.088, +0.113]$	$+1.17$
	L63 MLP	-0.793 $[-0.903, -0.689]$	-7.93
	L63 attention	-0.347 $[-0.394, -0.302]$	-3.38
Gemma3-4B	L23 attention	$+3.521$ $[+3.086, +3.977]$	$+25.65$
	L29 attention	$+0.262$ $[+0.227, +0.297]$	$+1.05$
	L33 MLP	-1.978 $[-2.238, -1.731]$	-2.91
	L31 attention	-0.623 $[-0.697, -0.552]$	-0.86
Gemma3-12B	L43 MLP	$+0.271$ $[+0.237, +0.306]$	$+2.27$
	L40 MLP	$+0.783$ $[+0.703, +0.864]$	$+7.33$
	L47 MLP	-2.089 $[-2.354, -1.834]$	-4.23
	L44 MLP	$+0.237$ $[+0.204, +0.270]$	$+0.84$
Gemma3-27B	L52 MLP	$+0.668$ $[+0.597, +0.742]$	$+2.18$
	L54 MLP	$+0.136$ $[+0.105, +0.165]$	$+1.05$
	L53 MLP	-0.220 $[-0.259, -0.185]$	-0.31
	L60 MLP	-1.212 $[-1.354, -1.078]$	-1.60

Table 7: Natural counterfactual component-replacement results. Effects are oriented by the donor's original final-layer Delta preference, with original within-size evaluation-group weights and equal weighting of the six sizes. Intervals are pointwise 95% group-bootstrap intervals.