GRAPH REPRESENTATIONS IN J-SPACE

Anonymous authors
Paper under double-blind review

ABSTRACT

Language models appear to have a *global workspace*: a small part of their internal state that holds active concepts and supports multi-step reasoning. The Jacobian lens reads this workspace as individual words, but reasoning also requires knowing how things are *related*. Answering "Who is the teacher of the banker of John?" requires following a graph whose nodes are people and whose edges are relations. We investigate whether the workspace holds these edges or only the nodes. We compare questions about the same story that differ in one relation word, and edit the workspace to test which changes affect the answer. Across multiple models, the workspace holds both nodes and edges, but treats them differently. Editing a person affects the answer where the name appears and at every model depth; editing a relation matters only at the relation word and in the middle layers. The two are never available at the same place and time, so the workspace never holds the connected graph. By the time the answer forms, the relation that determined it has left the workspace, although it remains elsewhere in the model. Workspace inspection at the moment of answering can therefore miss the relation responsible for the answer. Our findings motivate interpretability methods that trace relational information across token positions and layers, rather than relying on a final workspace snapshot.

1 INTRODUCTION

When people work through a problem, they seem to keep a few relevant items in mind and make them available to different parts of the brain. Global workspace theory describes this shared, limited store as a *global workspace* (Baars, 1988; Dehaene & Changeux, 2011). The idea has also inspired machine-learning architectures (Goyal et al., 2022; VanRullen & Kanai, 2021). Recently, Gurnee et al. (2026) reported that language models develop such a workspace on their own. Their *Jacobian lens* (J-lens) reads a model's internal state as a weighted list of words representing the concepts it is currently working with. The part of the state that the lens reads, called J-space, behaves like a workspace in several ways. The model can report its contents when asked, and the workspace carries intermediate reasoning steps that are never written out. Removing it disrupts multi-step reasoning while leaving fluent text generation and simple factual recall intact.

Because the lens reads individual words, J-space naturally represents *things*, such as people, places, and objects. Reasoning, however, usually depends on how those things are *related*. Consider the question "Who is the teacher of the banker of John?" (Figure 1A). The model must first identify John's banker, Lisa, and then Lisa's teacher, Mark. The people form the *nodes* of a small graph, and the relations form its *edges*. Keeping the relevant people in mind is not enough: the model must also know which relation connects each pair. Gurnee et al. (2026) identify relational structure as something the lens does not recover. Yet multi-step reasoning depends heavily on relations and fails when the workspace is removed. Whether the workspace holds the edges needed for this reasoning has not been thoroughly investigated. Answering this question matters both for understanding how models reason and for checking their reasoning by inspecting the workspace.

We therefore ask: **when a language model reasons over a graph, does its workspace hold the graph, or only its nodes?** We consider three hypotheses (Figure 1B). The workspace may hold only nodes, with relations applied elsewhere (H1). It may hold nodes and relations together as a labelled graph (H2). Or it may bind them into units, storing a statement such as "Mark is Lisa's teacher" as one unit (H3). To distinguish these hypotheses, we use *minimal edge pairs*: two questions about the same story that differ in a single relation word. For example, we compare "Who is the *teacher* of the banker of John?" with "Who is the *doctor* of the banker of John?" Everything else is identical, so any difference in how the model processes the questions must originate from that relation word.

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.

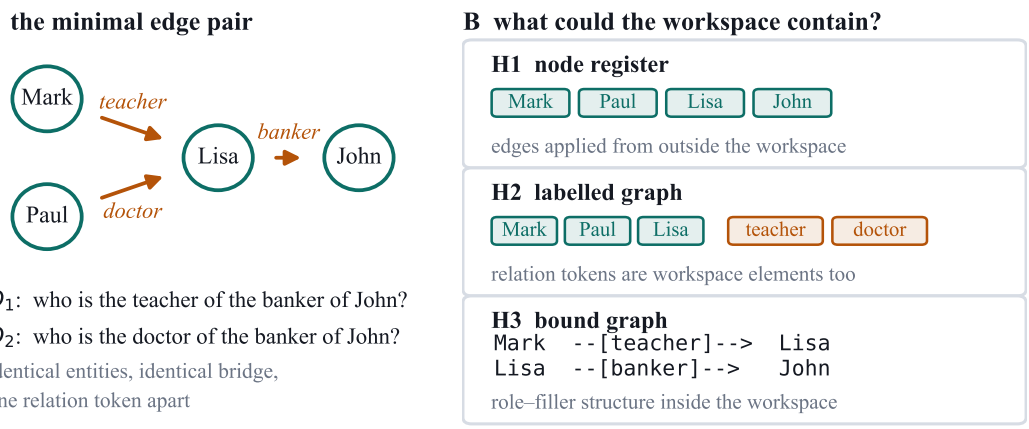


Figure 1: **A:** A minimal edge pair. Two questions about the same in-context graph share all entities, including the bridge entity (*Lisa*), which is not named in either question. They differ in only one relation token, so the identity of that edge determines which answer is correct. **B:** The possible contents of the workspace and each hypothesis's predictions for lens-coordinate interventions.

We then edit the workspace directly, replacing one person or relation with another at selected token positions and model depths, and test whether the answer changes accordingly.

The result differs from all three hypotheses. The workspace holds relations, but handles them differently from people. A person is *broadcast*: editing a name at positions where it appears changes the answer at every depth of the model. A relation is *local*: editing it changes the answer only at the relation word itself and only in the middle layers. After that window, editing the relation in the workspace no longer affects the answer. Because a person and the relation linking them are never available at the same place and time, the workspace never holds the connection between them. By the time the answer forms, the relation that determined it has left the workspace, although the model still carries it elsewhere. A monitor that reads the workspace when the model answers can therefore see the conclusion without seeing which relation produced it.

Our main findings are:

- Minimal edge pairs isolate the relation being queried. Because the questions differ only in that relation, differences in the model's behavior can be traced to it (§3).
- Nodes are broadcast, while relations remain local. Workspace edits show that people and relations affect the answer from different positions and at different depths, across 43 of 48 layer-by-model tests. Replacing the J-lens with a simpler lens makes the effect $26\times$ smaller, indicating that the pattern is a characteristic property of the workspace (§4.2–4.3).
- The workspace never holds the connection between nodes and relations, and the relation leaves the workspace while remaining in the model. Transferring the workspace between paired questions produces only 2.3% of the effect of transferring the entire internal state. When the answer forms, the relation remains readable from the full internal state but is barely readable from the workspace. Interventions outside the workspace can still steer the answer (§4.5–4.7).

2 Background and Related Work

Lenses and the workspace. The logit lens introduced the practice of reading intermediate residual states through the unembedding matrix (nostalgebraist, 2020). Learned affine probes subsequently made these readouts more faithful (Belrose et al., 2023). The J-lens (Gurnee et al., 2026) replaces the learned mapping with the input–output Jacobian averaged over a corpus, $\bar{J}_\ell = \mathbb{E}[\partial h_{\text{final}}/\partial h_\ell]$. Its readout therefore reflects causal influence on the output, rather than only whether information can be decoded linearly. We use this construction as given and investigate what the resulting workspace contains.

Relational structure and binding. Connectionist accounts of variable binding (Smolensky, 1990) anticipate the failure we report: combining contents without preserving their roles loses the information about which item fills which role. Feng & Steinhardt (2024) showed that language models solve in-context binding using *binding ID* vectors attached to entities and attributes, with subsequent

refinements by (Feng et al., 2024). This work locates binding within the residual stream. However, it predates the J-lens and does not examine whether binding IDs lie inside or outside the workspace.

Multi-hop reasoning and graphs. Previous work shows that bridge entities appear in middle-layer representations (Yang et al., 2024), and that the second hop is resolved only after the first (Biran et al., 2024), sometimes too late to be used. These studies track *entities* while treating the relations applied to them as given. A separate line of work studies which graph problems transformers can solve (Sanford et al., 2024; Back de Luca & Fountoulakis, 2024) and documents systematic failures in search and planning (Bachmann & Nagarajan, 2024; Momennejad et al., 2023). However, it does not identify the representations responsible for those failures. We extend the causal analysis previously applied to entities to the edges themselves. Our measurements of workspace capacity also provide a possible mechanism for these failures.

3 METHOD

3.1 J-SPACE, LENS VECTORS, AND INTERVENTIONS

We use the J-lens to read concepts from the model's residual stream and to identify directions along which to intervene. Let $h_\ell \in \mathbb{R}^d$ denote the residual stream at layer ℓ and a given token position, W_U the unembedding matrix, and $\bar{J}_\ell$ the fitted average Jacobian. The J-lens readout is $\mathrm{lens}_\ell(h) = W_U \,\mathrm{norm}(\bar{J}_\ell h)$. For each vocabulary token t, the ***J-lens vector*** identifies the residual-space direction whose inner product drives that token's lens logit:

$$v_t^{(\ell)} = \bar{J}_\ell^\top W_U[t], \qquad \hat{v}_t^{(\ell)} = v_t^{(\ell)} / \|v_t^{(\ell)}\|. \tag{1}$$

We apply three types of intervention using forward hooks at selected layers and token positions.

Lens-coordinate patching replaces a source concept s with a destination concept d. It removes the residual's component along the source direction and adds the same amount along the destination direction, preserving the component orthogonal to both:

$$h \;\mapsto\; h - \langle h, \hat{v}_s\rangle \hat{v}_s + \langle h, \hat{v}_s\rangle \hat{v}_d. \tag{2}$$

Each patch has two matched controls: one replaces the destination with an unrelated relation of the same kind, and the other replaces it with a random unit direction. The random-direction control is especially important near the output, where any large edit to the residual can perturb the logits.

Active-set ablation removes the components of each position's residual that lie in the span of *its own* top-k J-lens directions. We define this active set as the k vocabulary tokens with the highest lens scores at that position. A more literal implementation of the sparse-coding definition would use greedy non-negative pursuit over the full vocabulary, but this procedure is unstable at our scale (Appendix D). The matched control instead removes k randomly drawn dictionary atoms.

Subspace transfer tests which part of the residual carries information between two prompts, A and B, whose token positions are aligned. Let P be the projector onto the active set at layer ℓ. We replace either the active-set component of prompt B with that of prompt A (J-transfer), or the component outside the active set ($J^\perp$-transfer):

$$h^B \mapsto \underbrace{h^B - Ph^B + Ph^A}_{J\text{-transfer}} \quad \text{or} \quad \underbrace{Ph^B + (h^A - Ph^A)}_{J^\perp\text{-transfer}}, \tag{3}$$

with full residual replacement serving as the upper bound. The active subspace contains k of the d_{model} directions, whereas its complement contains all the remaining directions. Because of this difference in size, transfer through the complement is uninformative on its own. We therefore also transfer a *random* k-dimensional subspace drawn from the same dictionary as a matched control.

3.2 MINIMAL EDGE PAIRS

Our design isolates the effect of changing a single relation (Table 1). Each pair of questions refers to the same graph, but changing one relation token changes the correct answer. We verify that every entity name and relation word is a single token when preceded by a space, because the J-lens resolves only single-token concepts. Each prompt includes one worked example using a separate set of names. This raises two-hop accuracy from 56% to 97.5% and removes fill-in-the-blank artefacts from the readout (Appendix C). We also include a one-hop variant as an automatic processing control in the sense of Gurnee et al. (2026). To check whether the results depend on wording, a second family of prompts expresses the same relational structure using finite verbs (*Anna teaches Simon*).

Table 1: A minimal edge pair. Both questions use the same context, including all four entities, the bridge entity (LISA, which is not named in either question), and the filler edges. The questions differ in exactly one relation token.

Context	*Mark* is the **teacher** of *Lisa*. *Paul* is the **doctor** of *Lisa*. *Lisa* is the **banker** of *John*. (+ filler edges, shuffled)
Q_1	Who is the **teacher** of the **banker** of *John*? → *Mark*
Q_2	Who is the **doctor** of the **banker** of *John*? → *Paul*

Position classes. We divide each prompt into three classes of token positions: entity tokens, relation tokens, and all remaining tokens. We assign positions by token identity, matching both the space-prefixed and sentence-initial forms of each word, and verify that the entity and relation classes do not overlap. Applying patches separately within these classes reveals the dissociation reported below. Patching the entire prompt combines the two effects and obscures their difference.

3.3 METRICS

Selectivity. At the final position, the model is about to output an entity. Absolute lens ranks therefore do not provide a fair comparison between entities and relations. Instead, we measure *within-category* selectivity: the paired log ratio of a task-irrelevant foil's rank to the task-relevant item's rank. For entities, we compare a distractor with the answer; for relations, we compare the counterfactual relation with the queried relation. **Workspace load.** Our capacity measure counts how many in-context entity tokens appear in the top-K (K=50) lens readout at a given position.

4 EXPERIMENTS

The results support three claims. First, edges enter the workspace, but their causal effects are restricted to the positions where the relations are written (§4.2–4.3). Second, nodes are the information that the workspace broadcasts and retains for reasoning (§4.4). Third, nodes and edges are never bound together in the workspace. The relation that determines the answer leaves the workspace, although it remains elsewhere in the model, before the answer is formed (§4.5–4.7).

4.1 SETUP

We study 8 open models from the Qwen3, Qwen3.5 and Gemma families (Qwen Team, 2025; Gemma Team, 2024; 2025), using their publicly released J-lenses fitted on WikiText-103. Unless stated otherwise, results in the main text are for Qwen3-1.7B. Results for the other models appear in the appendix, where we compare layers at matched fractions of model depth. Every model answers the two-hop queries above chance ($1/6$; accuracy 0.34 to 1.00) and assigns greater probability to the answer selected by its own question than to the answer selected by the paired question (Appendix H). Because the questions differ in only one relation token, the model must represent which relation was queried somewhere in its internal state. Our main goal is to identify where. Unless stated otherwise, we use n=60 items per condition and report paired comparisons with bootstrap 95% confidence intervals over items. Counts of significant results use Holm correction within each family of tests (Appendix B). We measure behavioral effects as the change from the unedited baseline in $\log P(\text{alt}) - \log P(\text{ans})$. A positive value means that the intervention shifts the model toward the counterfactual answer.

4.2 NODES AND EDGES BECOME READABLE AT DIFFERENT STAGES

It is expected that entities will dominate the readout just before the model outputs an entity. The surprising result is what happens during the first two thirds of the network. Below layer 20, node selectivity is *indistinguishable from zero* (-0.05), whereas edge selectivity is substantial ($+1.27$ on average, peaking at $+2.80$ log rank units at layer 17). For most of the forward pass, the workspace at the final position therefore distinguishes the queried relation from an irrelevant relation, but does not distinguish the answer entity from an irrelevant entity. This ordering reverses only near the end of the network (Figure 2; Appendix P). The pattern is inconsistent with a workspace representation that gradually builds up the answer. It also recurs across models: in every model, the layers before

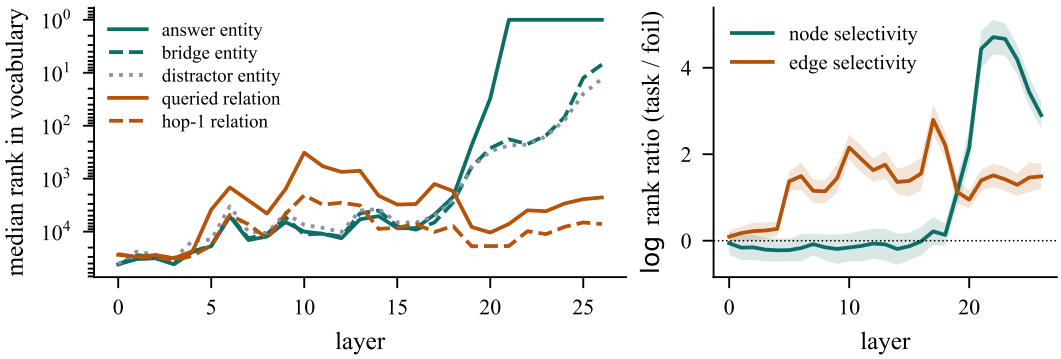


Figure 2: J-lens readout at the final position of Qwen3-1.7B. **Left:** median vocabulary rank across layers, shown on an inverted log scale so that higher values indicate stronger presence in the readout. **Right:** within-category selectivity, measured as the paired log ratio of a task-irrelevant foil's rank to the task-relevant item's rank. Below layer 20, only the relation is selectively represented; above it, only the entity is selectively represented.

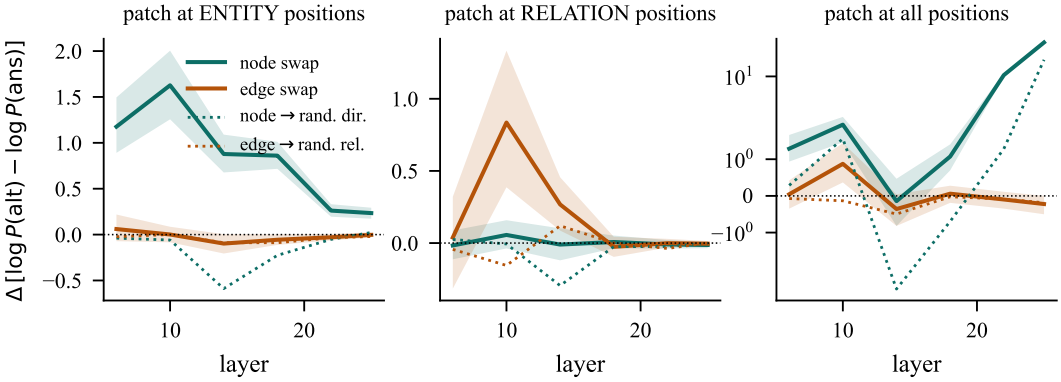


Figure 3: Lens-coordinate swaps across layers, applied separately to each position class. **Left:** at entity positions, only entity swaps change the model's answer preference. **Middle:** at relation positions, only relation swaps change its answer preference, and only within a mid-network window. **Right:** patching all positions combines these effects and also produces large late-layer effects that can be reproduced by a random direction. The dissociation must therefore be assessed using the panels that restrict patches to a position class (Appendix R). Dotted lines show matched random controls.

the crossover are selective for edges but not nodes, and the crossover occurs at roughly two thirds of the model's depth (Appendix J).

One possible objection is that the final-position readout simply reflects the next token: because that token *is* an entity, an output-directed lens may naturally favour entities. We address this concern by comparing items within each category and, more importantly, through the causal experiments below. Reading all token positions, rather than only the final one, also confirms that the lens can detect relations (Appendix U). *Both* categories reach the top of the readout at their own token positions: relations rank 1.0 at relation positions, and entities rank 1.5 at entity positions. The difference is how far these representations propagate. At the final position, the answer entity has rank 1.0, whereas the queried relation has rank 374.5.

4.3 Edges are causal only where they are written

Our central experiment asks whether changing a concept in the workspace changes the model's answer. We use lens-coordinate patching (Eq. 2) to exchange one concept for another while preserving the orthogonal component, and then measure whether the model shifts toward the corresponding alternative answer.

The result is a clear double dissociation: the effect of a swap depends on both what is swapped and where the swap is applied (Figure 3, Table 2). **At entity positions**, replacing the answer entity

Table 2: Causal swaps by position class. Bold values indicate intervals that exclude zero. These cells occur only when the swapped concept matches the position class: node swaps at entity positions and edge swaps at relation positions.

	patch at **entity** positions		patch at **relation** positions	
layer	node swap	edge swap	node swap	edge swap
6	$\mathbf{1.18}$ $[0.89, 1.49]$	0.06 $[-0.07, 0.22]$	-0.02 $[-0.11, 0.09]$	0.04 $[-0.32, 0.32]$
10	$\mathbf{1.63}$ $[1.26, 2.00]$	0.00 $[-0.08, 0.09]$	0.06 $[-0.04, 0.16]$	$\mathbf{0.84}$ $[0.39, 1.33]$
14	$\mathbf{0.88}$ $[0.68, 1.09]$	-0.10 $[-0.21, 0.01]$	-0.01 $[-0.12, 0.10]$	$\mathbf{0.27}$ $[0.09, 0.46]$
18	$\mathbf{0.86}$ $[0.72, 1.01]$	-0.06 $[-0.11, 0.00]$	0.01 $[-0.04, 0.05]$	-0.03 $[-0.10, 0.04]$
22	$\mathbf{0.26}$ $[0.19, 0.33]$	-0.03 $[-0.06, 0.01]$	-0.01 $[-0.04, 0.02]$	-0.00 $[-0.03, 0.03]$
25	$\mathbf{0.23}$ $[0.17, 0.29]$	-0.00 $[-0.03, 0.02]$	-0.01 $[-0.04, 0.01]$	-0.00 $[-0.02, 0.01]$

with the counterfactual entity shifts the model's answer preference at every probed layer ($\Delta = 1.63$ $[1.26, 2.00]$ at its peak). Replacing the decisive relation at those same positions has no effect ($\Delta = 0.00$ $[-0.08, 0.09]$). **At relation positions**, the pattern reverses: relation swaps affect the model ($\Delta = 0.84$ $[0.39, 1.33]$), whereas entity swaps do not ($\Delta = 0.06$ $[-0.04, 0.16]$). Both matched controls, namely replacing a relation with an unrelated relation and replacing an entity with a random unit direction, have effects consistent with zero within these position classes. Thus, edges are present in the workspace and can causally affect the answer, even though patching the whole prompt would obscure this result.

To establish a double dissociation, we must test the interaction directly, rather than assess each effect separately. We therefore calculate the within-item paired contrast $[(\text{node} - \text{edge})_{\text{entity}} - (\text{node} - \text{edge})_{\text{relation}}]$. This contrast is positive at every probed layer, peaking at $\Delta = 2.41$ $[1.88, 2.97]$ at layer 10. The effects differ in how long they persist as well as where they occur. Entity effects are significant at every probed layer and continue through to the output. Relation effects are restricted to layers L10 and L14 and disappear afterwards. The edge therefore leaves the causal path well before the answer is formed. Among the position classes and depths we test, there is no combination at which both nodes and edges can be manipulated to affect the answer.

A swap with no behavioral effect is informative only if it actually changed the representation at the patched positions. We therefore read the J-lens again after each swap and measure how far the *destination* concept has risen in the readout (Appendix Q). Through the middle of the network, edge swaps install their target relation $2.7\times$ more strongly than node swaps install their target entity. This ordering reverses only near the output. Thus, the relation is successfully written into J-space and remains readable there after the intervention. Once the mid-network window has closed, however, downstream processing no longer uses that workspace representation.

Is the Jacobian doing the work? Because our interventions use J-lens vectors, we need to establish whether the dissociation depends on the J-lens or would appear with any unembedding-based readout. In the latter case, the workspace interpretation might not be necessary. We test this by replacing the Jacobian transport with the identity. This changes $v_t = \bar{J}_\ell^\top W_U[t]$ into the plain logit-lens vector $v_t = W_U[t]$. We then repeat the same experiment, with patches restricted to the same position classes.

The dissociation does not survive this substitution. With the J-lens, the interaction is significant at 6 of 6 probed layers and peaks at $\Delta = 2.47$ $[1.96, 2.98]$. With the logit lens, it is significant at 1 of 6 and peaks at $\Delta = 0.10$ $[0.03, 0.17]$, a factor of $26\times$ smaller (Appendix S). We replicate this control in Qwen3-4B: the J-lens interaction is significant at 6 of 6 layers, compared with 2 of 6 for the logit lens, and remains $4.8\times$ larger. The effect is therefore specific to J-space, rather than a general property of the residual stream visible through any lens.

Across surface forms and models. The dissociation remains when every edge is expressed as a finite verb (*Anna teaches Simon*) rather than a nominal relation. The interaction reaches $\Delta = 2.55$ $[1.95, 3.21]$ at layer 10 (Appendix E). Across models, the interaction survives correction in 43 of 48 layer-by-model cells, including cells from the one pretrained-only checkpoint. It therefore does not depend on the post-training reported to install the assistant persona in the workspace (Gurnee et al., 2026) (Appendix J). The node effect survives correction in every model. The edge effect does in 6 of 8, all but Gemma-3-1B and Gemma-2-2B. The node effect is always the larger. In 6 of 8 models, the four cells separate less cleanly. Small cross-effects, such as entity swaps at relation positions, survive correction, except with Gemma-3-270M and Qwen3-1.7B. The dissociation thus appears to

be a reliable interaction between concept type and position class, rather than a pattern that requires the other effects to be exactly zero.

4.4 What the broadcast channel holds: a limited set of candidate entities

If relations are represented locally, how does the shared workspace help the model follow a path through a graph? Our results suggest that it keeps candidate entities available for use across multiple steps. Removing the top-k active J-lens directions late in the network leaves one-hop lookup near ceiling ($1.00 \rightarrow 0.90$) but reduces two-hop traversal accuracy ($0.93 \rightarrow 0.52$ at $k = 5$, versus 0.93 for the random control). Thus, disrupting the workspace has little effect on direct lookup but impairs the ability to combine two lookups. This mirrors the distinction between flexible and automatic processing reported by Gurnee et al. (2026) (Appendix I).

The workspace can hold only a limited set of entities. As the number of in-context entities increases from 4 to 18, workspace load rises from 1.6 to 3.2 and then levels off, while accuracy falls from 0.96 to 0.78. Longer traversal chains reveal a related limitation: accuracy drops from 1.00 to 0.36, but workspace load first rises and then *falls*. At greater depths, the model therefore retains fewer of the entities needed to follow the chain, even though the task requires more steps.

The workspace also helps explain which queries fail within graphs of the same size. On incorrect responses, it holds more candidate entities than on correct responses (2.97 versus 2.03, a difference of -0.93 $[-1.03, -0.84]$), but those candidates are less clearly distinguished. The correct answer is also less likely to be among them (0.39 versus 0.50, a difference of $+0.11$ $[+0.10, +0.12]$). Together, these findings suggest that failure involves crowding: the limited workspace becomes occupied by competing candidates while failing to retain the entity needed for the answer (Appendix F).

4.5 The graph is never bound

If a node and the relation governing it are never available together, the workspace cannot represent them as a bound graph. We test this directly by transferring residual subspaces between the two token-aligned prompts of a minimal edge pair (Eq. 3). We express each transfer's effect relative to the effect of replacing the entire residual.

Within the layers where edge interventions affect the answer, workspace transfer produces 2.3% of the full-replacement effect. This compares with -0.3% for a random subspace of the same dimension and 98.4% for the workspace's complement. At the last probed layer, the pattern reverses: workspace transfer produces 57.3%, compared with 40.9% for its complement. This late increase reflects the arrival of the *conclusion* in the workspace, rather than the arrival of the relation. Attribute binding shows the same pattern. Transferring the workspace between two scenes with different attribute bindings produces 6.4% of the full-replacement effect, compared with 4.9% for the matched control. These results do not support H3: the binding IDs described by Feng & Steinhardt (2024) lie outside the broadcast channel (Appendix G).

4.6 The edge leaves the workspace, not the model

The swap and transfer experiments show that the workspace stops *driving* the relational decision after the middle of the network. However, they do not establish whether the relation is still *present* in the workspace. To answer that question, we train a linear probe to identify which relation was queried.

The probe reads the residual at the final token position, using either the workspace coordinates, a random subspace of the same dimension, or the full residual. Chance performance is 0.08. In early layers, the queried relation can be decoded from all three representations, as expected given that it appears in the prompt. Workspace decoding peaks at 1.00 and then declines. It falls to half its peak by 79% of the model's depth and reaches 0.20 at 93%. At that same depth, the full residual still supports decoding at 0.70 (Figure 9, Appendix O). The relation information has therefore not been destroyed or consumed. It remains in the model, but is no longer readable from the workspace.

A control experiment adds an important qualification: this decline mainly reflects *dimensionality*, rather than a distinctive property of J-space. Over the final fifth of the network, relation decoding is 0.25 from a matched random subspace and 0.28 from a logit-lens basis of the same size, compared with 0.18 from J-space and 0.75 from the full residual. None of these 25-dimensional readouts retains the relation where the answer forms. The distinctive finding about J-space is therefore not that it loses the relation faster. Rather, it never carries the relation *better* than an arbitrary subspace

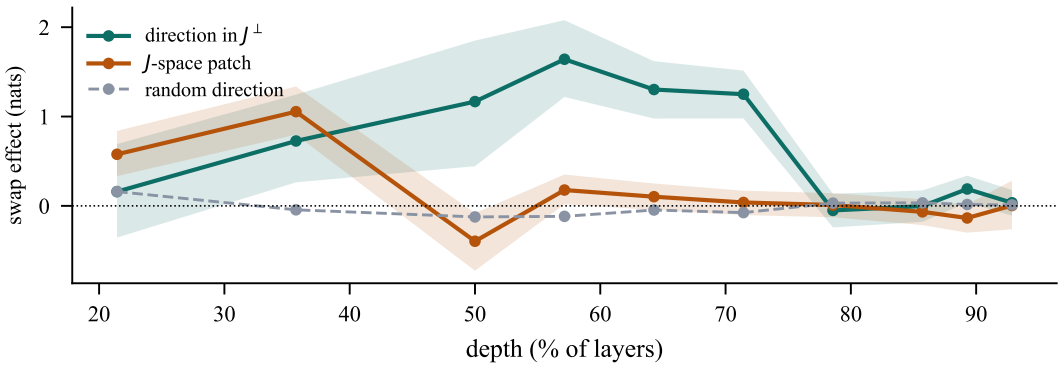


Figure 4: Effects of the same scale-free swap along three directions across the depth of Qwen3-1.7B. The workspace direction has the larger effect in early layers. Through the middle of the network, the larger effect comes from a direction recovered from held-out data and projected off J-space. Beyond 79% of the model's depth, neither direction changes the model's answer preference. A random direction has an effect near zero.

of the same size, at any depth. The causal asymmetry reported in §4.3, by contrast, is specific to the J-lens.

Measuring the auditing blind spot. Regardless of its cause, this loss of readable information matters for auditing. At the depth where the answer forms, a monitor with access to the full residual stream could recover which relation the model is applying. A monitor restricted to J-space, as provided by the J-lens, could not. The gap between 0.70 and 0.20 measures this blind spot. The model's behavior makes the consequence concrete. On token-aligned question pairs, it gives the two different correct answers, answering *oppositely*, on 0.97 of trials. Yet through the middle of the network, the paired top-K workspace readouts have Jaccard overlap 0.51 and cosine similarity 0.90. The readouts separate only when the answer is resolved.

4.7 Is the edge reachable outside the workspace?

After the middle of the network, the workspace no longer carries the edge, but the relation remains decodable from the residual (§4.6). Can an intervention still *reach* this information and change the model's answer? We fit the probe on training items and use it to construct a direction contrasting the queried relation with the counterfactual relation. We then project this direction off J-space, ensuring that it lies entirely in the workspace's complement. On *held-out* items, we apply the same scale-free swap used for the workspace patch. We compare four conditions: the projected direction, the unprojected probe direction, a random direction, and the J-space patch. Each applies the same operation in a different basis, making the effects directly comparable.

In Qwen3-1.7B, the relative strength of the workspace and complementary directions changes across depth (Figure 4). The workspace patch has the larger effect early on, peaking at $\Delta = 1.05$ $[0.80, 1.34]$ at 36% of the model's depth. The projected direction overtakes it by 50% and reaches its maximum effect of $\Delta = 1.64$ $[1.22, 2.08]$ at 57%. At that point, the workspace patch has fallen to $\Delta = 0.18$ $[0.02, 0.35]$, while the random direction produces $\Delta = -0.12$ $[-0.22, -0.01]$. Beyond 79%, neither direction changes the model's answer preference. Thus, the relation information is not consumed when it leaves the workspace. It remains causally reachable for roughly a third of the network, mainly outside the broadcast channel.

Replication. We ran this experiment on 5 models in total. We count a layer as showing an effect only if the perpendicular direction both changes the model's answer preference and has a larger effect than a random direction subjected to the same swap. This control matters because, at some layers, any edit of this size can perturb the output in either direction.

The shift from workspace effects to effects outside the workspace appears in 3 of the 5 models (Qwen3-1.7B, Qwen3.5-2B-Base and Qwen3-4B). Where it appears, it occupies overlapping bands: 36–71% of depth in Qwen3-1.7B; 46–62% of depth in Qwen3.5-2B-Base; 53–64% of depth in Qwen3-4B (Appendix M). It is absent in Gemma-2-2B and Gemma-3-4B: in Gemma-2-2B the J-space route clears random at only 2 layers and reaches at most 5% of the decision margin; in Gemma-

3-4B neither route moves the decision toward the counterfactual answer, though entity swaps still move the model. Every Qwen model we test shows the handoff and no Gemma model does. We therefore claim the handoff as a recurring organization, not a universal one.

Is the split a lens artifact? A poorly fitted lens would mis-specify the subspace we project off. Two checks say no. First, the unprojected probe, which never touches the lens, fails in Gemma-2-2B and Gemma-3-4B just as its projection does (Appendix M). Second, by a direct measure Gemma's lenses fall within the range of Qwen's: the lens's own gradient aligns with the model's true decision gradient at cosine 0.10–0.17 in Gemma against 0.10–0.18 in Qwen between 45% and 75% of depth (random directions: 0.02). In every model the lens begins to track the decision only at 62–71% of depth, where the readout crossover of §4.2 falls (Appendix N).

5 Discussion

The workspace represents nodes and edges separately. Our results rule out both of the most immediate interpretations of what the workspace holds. It is not simply a register of nodes (H1): relations are present in J-space and causally affect the answer at their own token positions. It is also not a labelled graph (H2): broadcast nodes and local edges are never available together. Specifically, we find no layer–position pair (ℓ, p) at which interventions on both nodes and edges affect the answer. This provides an empirical counterpart to Smolensky (1990)'s argument that superposing contents without encoding their roles cannot support binding (Appendix L).

What is general, and what is model-specific. The separation of nodes and edges holds in every model we test. What happens to the edge after it leaves the workspace varies across models. The shift to a causally effective direction outside J-space appears in Qwen3-1.7B, Qwen3.5-2B-Base and Qwen3-4B but not in Gemma-2-2B and Gemma-3-4B (§4.7). Models also differ in how firmly the workspace holds the answer. This differs by $37\times$ between Gemma-3-4B and Qwen3-4B, despite their matching parameter counts and accuracy (Appendix K). Thus, the relation's departure from the workspace is general across the tested models, but where it goes depends on the particular model.

Consequences for auditing. It is natural to inspect a workspace monitor (Gurnee et al., 2026) where the answer forms. However, this is the wrong place to identify the relation guiding the answer: by that point, the relation has already left the workspace (§4.6). To recover it, read J-space at relation-bearing positions in the middle of the network, or combine the workspace monitor with a probe on the full residual.

Limitations. Our experiments are limited to models up to 4B parameters. The J-lens resolves only single-token concepts (Gurnee et al., 2026), so the results may differ for multi-token relations. Also, we define the active set at each position using the top-k tokens by lens score. Although we validate this choice causally, it does not implement the sparse-coding definition directly (Appendix D). Finally, a null result constrains only what can be reached by an edit along a single direction. Both stimulus families also use templated prompts.

6 Conclusion & Future Work

When a language model reasons over a graph, its workspace holds nodes and edges, but never together: nodes are broadcast, while edges remain local and transient. It therefore represents the graph's parts without binding them. By the time the answer forms, the relation that selected it has left the workspace, though not the model. Workspace monitors should inspect relation-bearing positions, not only where the conclusion appears.

Toward explicit graphs in the workspace. These findings suggest three ways to make graph structure explicit in J-space. First, *read and edit the graph where its parts live*: nodes at entity positions and edges at relation positions in the mid-network window. Token adjacency could supply binding externally, allowing monitors to reconstruct the graph and interventions to target its parts. Second, *give the lens relational atoms*: index the readout by (relation, entity) pairs rather than individual tokens. A low-rank readout bilinear in their lens vectors could make edges addressable where the answer forms, if they remain represented there. Third, *route relations through the workspace*: reward relation decodability from J-space at the decision position through an auxiliary objective. This would test whether binding can be installed and at what cost.

REFERENCES

Bernard J. Baars. *A Cognitive Theory of Consciousness*. Cambridge University Press, 1988.

Gregor Bachmann and Vaishnavh Nagarajan. The pitfalls of next-token prediction. In *ICML*, 2024.

Artur Back de Luca and Kimon Fountoulakis. Simulation of graph algorithms with looped transformers. In *ICML*, 2024. arXiv:2402.01107.

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

Eden Biran, Daniela Gottesman, Sohee Yang, Mor Geva, and Amir Globerson. Hopping too late: Exploring the limitations of large language models on multi-hop queries. In *EMNLP*, 2024. arXiv:2406.12775.

Nelson Cowan. The magical number 4 in short-term memory. *Behavioral and Brain Sciences*, 24 (1):87–114, 2001.

Stanislas Dehaene and Jean-Pierre Changeux. Experimental and theoretical approaches to conscious processing. *Neuron*, 70(2):200–227, 2011.

Jiahai Feng and Jacob Steinhardt. How do language models bind entities in context? In *ICLR*, 2024. arXiv:2310.17191.

Qin Feng, Yusuke Sakai, et al. Representational analysis of binding in language models. In *EMNLP*, 2024. arXiv:2409.05448.

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

Gemma Team. Gemma 3 technical report. *arXiv preprint arXiv:2503.19786*, 2025.

Anirudh Goyal, Aniket Didolkar, Alex Lamb, Kartikeya Badola, Nan Rosemary Ke, Nasim Rahaman, Jonathan Binas, Charles Blundell, Michael Mozer, and Yoshua Bengio. Coordination among neural modules through a shared global workspace. In *ICLR*, 2022.

Wes Gurnee, Nicholas Sofroniew, Adam Pearce, Mateusz Piotrowski, Isaac Kauvar, Runjin Chen, Anna Soligo, Paul Bogdan, Euan Ong, Rowan Wang, Ben Thompson, David Abrahams, Subhash Kantamneni, Emmanuel Ameisen, Joshua Batson, and Jack Lindsey. Verbalizable representations form a global workspace in language models. *Transformer Circuits Thread*, 2026. arXiv:2607.15495.

George A. Miller. The magical number seven, plus or minus two. *Psychological Review*, 63(2): 81–97, 1956.

Ida Momennejad, Hosein Hasanbeig, Felipe Vieira, Hiteshi Sharma, Robert Ness, Nebojsa Jojic, Hamid Palangi, and Jonathan Larson. Evaluating cognitive maps and planning in large language models with CogEval. In *NeurIPS*, 2023.

nostalgebraist. Interpreting GPT: the logit lens. LessWrong, 2020.

Qwen Team. Qwen3 technical report. *arXiv:2505.09388*, 2025.

Clayton Sanford, Bahare Fatemi, Ethan Hall, Anton Tsitsulin, Mehran Kazemi, Jonathan Halcrow, Bryan Perozzi, and Vahab Mirrokni. Understanding transformer reasoning capabilities via graph algorithms. In *NeurIPS*, 2024.

Paul Smolensky. Tensor product variable binding and the representation of symbolic structures in connectionist systems. *Artificial Intelligence*, 46(1-2):159–216, 1990.

Rufin VanRullen and Ryota Kanai. Deep learning and the global workspace theory. *Trends in Neurosciences*, 44(9):692–704, 2021.

Sohee Yang, Elena Gribovskaya, Nora Kassner, Mor Geva, and Sebastian Riedel. Do large language models latently perform multi-hop reasoning? *arXiv:2402.16837*, 2024.

A Reproducibility and compute

We use the reference J-lens implementation released with Gurnee et al. (2026) to read the workspace, and implement all causal interventions ourselves. The lenses are publicly available through Neuronpedia and were fitted on WikiText-103 using 128-token sequences. Reproducing our results therefore requires no additional lens fitting. The full experiment suite takes roughly two hours per model. The submission includes the code, stimuli, raw results, and a script that regenerates every number in the paper from those results.

B Multiple comparisons

For each layer, we test effects using 95% bootstrap intervals over items. We then adjust every claim that *counts* significant layers, cells, or models for multiple comparisons. A test contributes to a count only if it meets its original criterion *and* survives Holm correction within the listed test family, with family-wise α=.05. Corrected counts therefore cannot exceed the original counts. We also report Benjamini–Hochberg correction (q=.05) for comparison.

Where per-item data were saved, as in the swap experiments of §4.3, we use exact percentile-bootstrap p-values. Otherwise, we estimate p-values from the saved intervals using a normal approximation. Where both methods are available, they produce the same corrected decision in 47 of 48 cells.

Some claims require an effect to be significant *and* distinguishable from the random control. For these conjunctions, we use the larger of the two p-values for that layer, following an intersection–union test. All counts in the main text use the corrected results.

Table 3: Counts of significant results before and after multiple-comparison correction.

Counted claim	family	uncorrected	Holm	BH
Interaction contrast > 0, layer $\times$ model cells	48 cells	43	43	43
Node swap at entity positions: models with ≥ 1 layer > 0	layers, per model	8	8	8
Edge swap at relation positions: models with ≥ 1 layer > 0	layers, per model	7	6	6
Cross-effects: models with any significant layer	layers, per model	6	6	6
Lens control, J-lens, Qwen3-1.7B: layers	6 layers	6	6	6
Lens control, logit lens, Qwen3-1.7B: layers	6 layers	1	1	1
Lens control, J-lens, Qwen3-4B: layers	6 layers	6	6	6
Lens control, logit lens, Qwen3-4B: layers	6 layers	3	2	3
Second stimulus family, interaction: layers	6 layers	4	4	4
Qwen3-1.7B: perpendicular beats random	10 layers	5	5	5
Qwen3-1.7B: J-space patch beats random	10 layers	2	1	1
Qwen3.5-2B-Base: perpendicular beats random	9 layers	5	4	4
Qwen3.5-2B-Base: J-space patch beats random	9 layers	4	4	4
Gemma-2-2B: J-space patch beats random	11 layers	2	2	2
Qwen3-4B: perpendicular beats random	11 layers	3	3	3
Qwen3-4B: J-space patch reversed	11 layers	5	4	5
Gemma-3-4B: J-space patch beats random	9 layers	1	0	1
Gemma-3-4B: J-space patch reversed	9 layers	2	2	2
Gemma-3-4B: perpendicular reversed	9 layers	3	3	3
Handoff present: models with ≥ 1 counted layer	layers, per model	3	3	3

C Prompt format

Without a worked example, two-hop accuracy is 56%, and the J-lens readout at the final position is dominated by tokens associated with fill-in-the-blank formatting. Adding one demonstration raises accuracy to 97.5% and places entities at the top of the readout. The demonstration uses names that do not occur in the sampled name pool. All reported experiments use this one-shot format.

D Instability of greedy pursuit at vocabulary scale

J-space is defined as the set of sparse, non-negative combinations of J-lens vectors. A direct implementation would use non-negative orthogonal matching pursuit over the full $|V| \times d$ dictionary.

In our setting, however, this procedure is unstable. At every layer, the selected tokens are dominated by rare tokens and tokens in foreign scripts whose lens vectors are nearly parallel to directions representing content.

The fraction of the residual left unexplained stays above 0.9 throughout. This is consistent with the original report that J-space accounts for under 10% of activation variance. The coefficient participation ratio is roughly 21–23, compared with the nominal $k{=}25$. Despite these numerical properties, the selected tokens are not interpretable.

Table 9 compares pursuit with selection by lens score at the final position of one item. Before the workspace becomes active, both methods return formatting tokens and neither gives an interpretable list of task concepts. This is the expected behavior of a lens in the early sensory layers. Within the workspace layers, however, the methods diverge. The top-k tokens by lens score form a clear list of person names that includes the correct answer, whereas pursuit continues to mix names with unrelated dictionary atoms.

We therefore define the operational active set separately at each position using the top-k tokens by lens score. We validate this choice through causal interventions in §4.3, rather than relying only on how interpretable the token list appears.

E SECOND STIMULUS FAMILY

Table 4 repeats the position-restricted swaps using a second stimulus family. It expresses every edge with a finite verb (*Anna teaches Simon*) rather than a nominal relation, changing the wording while preserving the graph. Two-hop accuracy is 0.85, and the interaction contrast peaks at $\Delta = 2.55$ $[1.95, 3.21]$ at layer 10. The dissociation therefore reflects the relational structure rather than one particular way of expressing it.

Table 4: Second stimulus family: the graph is unchanged, but every edge is expressed with a finite verb.

	entity positions		relation positions		
layer	node swap	edge swap	node swap	edge swap	interaction
6	**1.49** [1.07, 1.89]	0.07 [−0.10, 0.24]	−0.16 [−0.36, 0.02]	**0.58** [0.30, 0.89]	**2.15** [1.69, 2.65]
10	**1.83** [1.39, 2.24]	0.13 [−0.02, 0.26]	−0.15 [−0.30, 0.02]	**0.70** [0.32, 1.09]	**2.55** [1.95, 3.21]
14	**1.64** [1.36, 1.90]	−0.18 [−0.33, −0.02]	**0.26** [0.08, 0.43]	0.08 [−0.16, 0.34]	**1.65** [1.30, 1.98]
18	**1.18** [0.97, 1.38]	0.02 [−0.07, 0.10]	0.06 [−0.02, 0.15]	0.04 [−0.04, 0.12]	**1.14** [0.91, 1.36]
22	**0.29** [0.23, 0.36]	0.07 [−0.00, 0.14]	**0.18** [0.14, 0.22]	0.01 [−0.02, 0.04]	0.05 [−0.07, 0.17]
25	**0.28** [0.21, 0.34]	−0.01 [−0.08, 0.06]	**0.40** [0.34, 0.44]	−0.00 [−0.03, 0.03]	−0.11 [−0.20, −0.01]

F CAPACITY SWEEPS

Capacity. Figure 5 in Appendix F shows the graph-size and traversal-depth sweeps. The capacity of J-space is limited: the original work reports a few dozen simultaneously active components that account for under 10% of activation variance. Graph traversal, however, requires maintaining a frontier of candidate vertices that grows with the problem.

As the number of in-context entities increases from 4 to 18, workspace load rises from 1.6 to 3.2 and then levels off. At the same time, two-hop accuracy falls from 0.96 to 0.78. The saturation point is close to classical estimates of human working-memory capacity (Miller, 1956; Cowan, 2001). We treat this similarity as a coincidence worth studying further, rather than as evidence for a shared mechanism.

Failure is crowding. The graph-size sweep establishes an aggregate pattern: larger graphs produce higher load and lower accuracy. A more direct test asks whether the workspace state on *a particular item* predicts whether the model answers *that item* correctly.

Across 240 items, it does, in the direction expected of a capacity-limited store. The workspace holds 2.03 in-context entities on correct responses and 2.97 on errors (-0.93 $[-1.03, -0.84]$). On errors, these entities also receive less of the readout mass (0.36 on correct responses versus 0.30 on errors). The answer entity appears in the top-K readout on 0.50 of correct trials, compared with 0.39 of errors ($+0.11$ $[+0.10, +0.12]$). These differences persist within each graph size, so they cannot be explained by graph size alone. Errors occur when the workspace holds more competing candidates, represents them less strongly, and fails to include the correct one.

The traversal-depth sweep provides a further distinction. Accuracy falls from 1.00 at one hop to 0.36 at four hops. Workspace load increases only up to three hops and then declines. Where traversal fails, the workspace is therefore holding *fewer* of the chain's entities, rather than reaching saturation. This pattern is consistent with a workspace that maintains a small set of currently relevant vertices instead of accumulating a growing frontier. Beyond a certain depth, it loses track of the chain rather than overflowing.

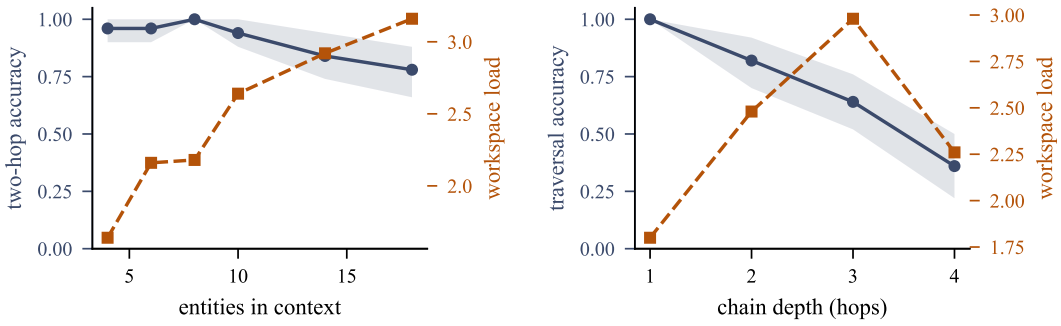


Figure 5: Left: as the graph grows, workspace load levels off while accuracy falls. Right: as traversal depth increases, accuracy falls sharply and load first rises, then *falls*. Where traversal fails, the workspace holds fewer of the chain's entities.

G SUBSPACE TRANSFER, PER LAYER

Subspace transfer: how much of the decision is in the workspace. The swap experiment changes one direction at a time. Subspace transfer asks how the *whole* causal contribution of the residual is divided between the workspace and its complement. The two questions in a minimal edge pair have aligned token positions, allowing us to transfer residual subspaces between them position by position (Eq. 3). Replacing the entire residual transfers the other question's answer. We use that effect as the reference and express all other transfer effects as percentages of it.

Within the window where edge swaps affect the answer, at layer 10, workspace transfer produces 2.3% of the full-residual effect. A random subspace of the same dimension, drawn from the same dictionary, produces -0.3%, while the orthogonal complement produces 98.4%. The workspace projector spans $k{=}25$ directions and contains 0.3% of the residual variance. Its small contribution is therefore partly explained by its size. However, the dimension-matched control provides the appropriate comparison, and the workspace contributes little relative to that control. Almost all of the relational decision is made in $J^{\perp}$.

At the last probed layer, 25, the pattern reverses. The workspace carries 57.3% of the full effect, compared with 3.1% for the matched control and only 40.9% for the complement. This late increase reflects the arrival of the *conclusion*. By this layer, the answer entity is at the top of the J-lens readout (§4.2), so workspace transfer carries the selected entity rather than the relation that selected it.

Together with the swap experiment, these results show that the workspace reports the outcome of relational computation after contributing little to the computation itself. Per-layer values and the corresponding figure are provided in Appendix G.

Binding. We apply the same test to attribute binding. The scenes *the green square and the blue circle* and *the blue square and the green circle* contain the same concept tokens and have aligned token positions. They differ only in which color belongs to which shape. If J-space held role–filler structure (H3), transferring the workspace between the scenes should also transfer these bindings.

Workspace transfer produces 6.4% of the full-residual effect, compared with 4.9% for the matched control and 91.1% for the complement. The workspace contents are thus nearly interchangeable even though the model gives opposite answers for the two scenes. These results do not support H3 and place the binding IDs described by Feng & Steinhardt (2024) outside the broadcast channel.

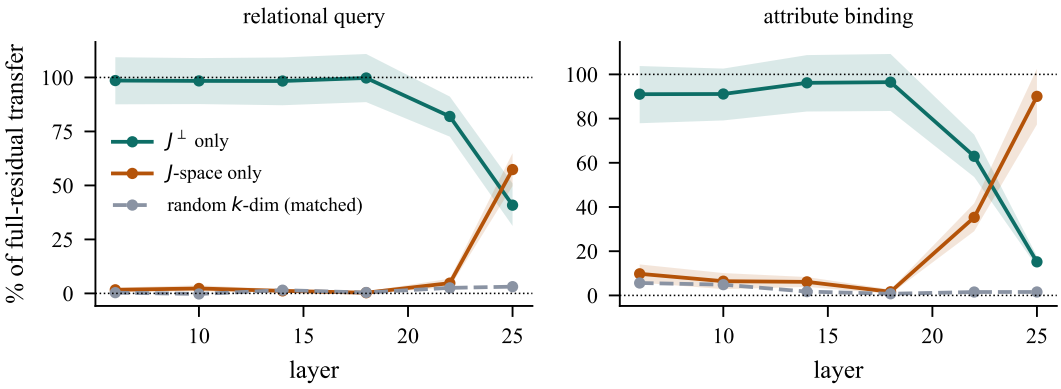


Figure 6: Effect of subspace transfer, expressed as a percentage of the effect of replacing the entire residual.

Table 5: Subspace transfer between the prompts of a minimal edge pair and between scenes containing the same concepts with different bindings. Effects are changes in $\log P(\text{alt}) - \log P(\text{ans})$ relative to no intervention. Full residual replacement provides the upper bound. The random k-dimensional control accounts for the workspace projector spanning only k of the d_{model} directions.

task	layer	full residual	$J^{\perp}$ only	random k-dim	J-space only
relational query	6	27.62 [24.55, 30.64]	27.22 [24.18, 30.20]	0.09 [−0.21, 0.42]	0.46 [0.19, 0.75]
	10	27.62 [24.55, 30.64]	27.17 [24.21, 30.09]	−0.07 [−0.41, 0.24]	0.63 [0.27, 1.04]
	14	27.62 [24.55, 30.64]	27.16 [24.06, 30.17]	0.41 [0.19, 0.64]	0.32 [0.11, 0.53]
	18	27.62 [24.55, 30.64]	27.55 [24.46, 30.62]	0.11 [0.01, 0.22]	0.09 [−0.07, 0.25]
	22	27.62 [24.55, 30.64]	22.63 [20.04, 25.17]	0.71 [0.42, 1.00]	1.31 [0.79, 1.89]
	25	27.62 [24.55, 30.64]	11.28 [8.63, 14.06]	0.86 [0.31, 1.41]	15.84 [13.72, 17.94]
attribute binding	6	23.00 [19.98, 25.98]	20.94 [17.92, 23.87]	1.29 [0.57, 1.97]	2.25 [1.34, 3.23]
	10	23.00 [19.98, 25.98]	20.96 [18.21, 23.61]	1.12 [0.72, 1.50]	1.48 [0.65, 2.33]
	14	23.00 [19.98, 25.98]	22.12 [19.14, 25.01]	0.40 [0.10, 0.65]	1.41 [0.89, 1.92]
	18	23.00 [19.98, 25.98]	22.20 [19.19, 25.13]	0.17 [0.03, 0.31]	0.38 [0.15, 0.62]
	22	23.00 [19.98, 25.98]	14.48 [12.32, 16.73]	0.35 [0.15, 0.56]	8.12 [6.68, 9.58]
	25	23.00 [19.98, 25.98]	3.50 [2.96, 4.09]	0.36 [0.21, 0.52]	20.71 [17.79, 23.59]

H TASK PERFORMANCE

I ABLATION SWEEP

Ablation: flexible versus automatic, for graphs. Using the corrected active-set definition, we reproduce the ablation finding of Gurnee et al. (2026) on graph tasks. Removing the top-k active J-lens directions late in the network leaves one-hop edge lookup near ceiling ($1.00 \rightarrow 0.90$) but reduces two-hop traversal performance ($0.93 \rightarrow 0.52$ at $k = 5$, versus 0.93 for the random control). Removing a random subspace of the same dimension leaves both tasks intact.

This effect occurs only at the last probed layer. Ablation in the middle of the network, where edge information is present, does not disrupt either task. Figure 7 in Appendix I shows the full sweep.

The effect does not increase consistently with k: larger active sets cause *less* damage than k=5. This cannot be explained by removing less information. Increasing k removes strictly more residual variance (2.1%, 2.4%, 2.9% for k=5, 10, 25), yet causes less damage. We report this non-monotonic pattern as observed.

Additional diagnostics confirm that the non-monotonic effect of k is not caused by the projection removing too little: k=5 removes 2.1% (condition number 7); k=10 removes 2.4% (condition number 13); k=25 removes 2.9% (condition number 41). Larger k removes more of the residual and produces a worse-conditioned Gram matrix, yet disrupts two-hop traversal less.

Table 6: One-hop and two-hop accuracy across models, with the ratio of the probability assigned to the correct answer to that assigned to the counterfactual answer.

Model	layers	one-hop acc.	$P(\text{ans})/P(\text{alt})$	two-hop acc.	$P(\text{ans})/P(\text{alt})$
Gemma-3-270M	18	0.37 [0.28, 0.46]	1.5	0.34 [0.26, 0.42]	1.3
Qwen3.5-0.8B	24	0.98 [0.96, 1.00]	18.5	0.91 [0.86, 0.96]	7.1
Gemma-3-1B	26	0.66 [0.57, 0.74]	3.0	0.57 [0.47, 0.66]	1.7
Qwen3-1.7B	28	0.97 [0.94, 1.00]	36.0	0.97 [0.95, 1.00]	84.2
Qwen3.5-2B-Base	24	0.99 [0.97, 1.00]	64.4	0.98 [0.96, 1.00]	22.6
Gemma-2-2B	26	0.94 [0.90, 0.98]	14.2	0.92 [0.86, 0.97]	5.7
Qwen3-4B	36	1.00 [1.00, 1.00]	$>10^6$	0.99 [0.97, 1.00]	710
Gemma-3-4B	34	0.98 [0.96, 1.00]	47.8	1.00 [1.00, 1.00]	12.1

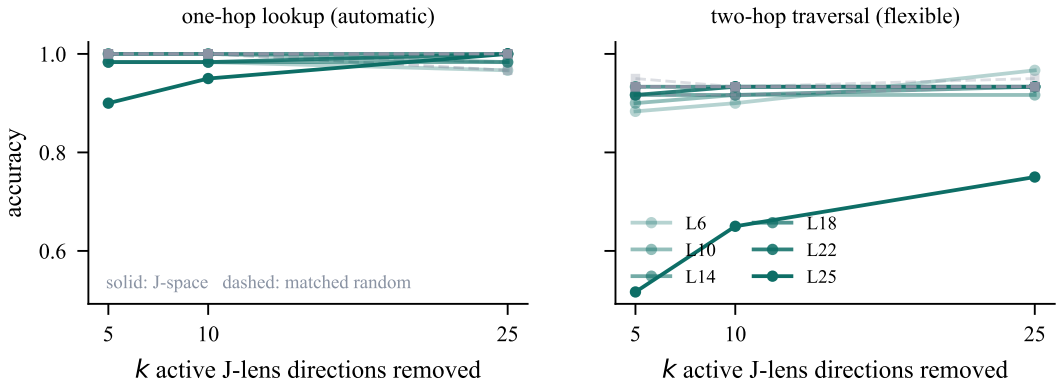


Figure 7: Ablation of the top-k *active* J-lens directions at each position. Solid lines show J-space ablation; dashed lines show removal of k randomly drawn dictionary atoms. One-hop lookup is largely unaffected. Two-hop traversal collapses only under J-space ablation.

J The dissociation across models

K Controllability across models

Controllability depends on the model, not just its size. After expressing each model's peak entity-swap effect relative to its decision margin, the effects span two orders of magnitude across models (0.07 to 17.22). They decrease with model size within every family for which we measure two sizes (Gemma-3 $17.22 \to 2.43$; Qwen3 $0.37 \to 0.07$; Qwen3.5 $4.66 \to 2.22$).

This pattern does not establish a scaling law. Model size and task competence are strongly correlated in our sample ($r = 0.85$). Moreover, Gemma-3-4B and Qwen3-4B have the same parameter count and matching accuracy (1.00 versus 0.99), yet their normalized entity-swap effects differ by $37\times$. How firmly the workspace holds the answer therefore depends on the particular model. A control technique calibrated on one 4B model may have no effect on another.

L Extended discussion

An unbound representation, and why it may be useful. Binding requires a role and the item filling that role to be available simultaneously to the same downstream process. Smolensky (1990) argued that superposing contents without encoding their roles cannot provide this information. Our position-specific results give an empirical counterpart: among the layer–position pairs (ℓ, p) we test, none allows effective interventions on both an entity and its governing relation. This places the binding IDs described by Feng & Steinhardt (2024) outside the broadcast channel.

Such a separation could be a deliberate design choice. Broadcasting individual contents requires $O(n)$ channel capacity, whereas broadcasting relations requires $O(n^2)$ and a shared binding convention among all downstream consumers (Baars, 1988; Cowan, 2001). Our finding that workspace load saturates near three entities, rather than continuing to grow with the graph, is consistent with a channel sized to carry individual contents alone.

Table 7: The dissociation across models: peak node-swap effects at entity positions and peak edge-swap effects at relation positions, with the layer at which each peaks.

Model	layers	node swap	edge swap	interaction	selectivity	pre-crossover sel.	
		at entity pos.	at relation pos.		crossover	edge	node
Gemma-3-270M	18	3.98 [3.13, 4.91]	0.35 [0.10, 0.62]	4.21 [3.36, 5.15]	61%	+0.39	−0.15
Qwen3.5-0.8B	24	9.14 [8.49, 9.75]	0.17 [0.09, 0.26]	8.86 [8.23, 9.46]	62%	+0.86	−0.18
Gemma-3-1B	26	5.95 [5.33, 6.60]	−0.00 [−0.02, 0.01]	5.72 [5.13, 6.33]	65%	+0.57	−0.18
Qwen3-1.7B	28	1.63 [1.26, 2.00]	0.84 [0.39, 1.33]	2.41 [1.88, 2.97]	71%	+1.27	−0.05
Qwen3.5-2B-Base	24	6.93 [6.40, 7.47]	0.36 [0.28, 0.44]	6.71 [6.13, 7.30]	62%	+1.23	+0.12
Gemma-2-2B	26	1.50 [1.36, 1.64]	0.04 [0.00, 0.07]	1.52 [1.35, 1.69]	69%	+1.27	+0.19
Qwen3-4B	36	0.43 [0.36, 0.52]	0.11 [0.02, 0.19]	0.41 [0.33, 0.50]	64%	+0.59	−0.08
Gemma-3-4B	34	6.05 [5.34, 6.72]	0.17 [0.06, 0.28]	6.02 [5.33, 6.70]	68%	+1.10	+0.21

M STEERING REPLICATION, PER MODEL AND LAYER

Each block reports the effects of the same scale-free swap at different depths, using the perpendicular direction, the unprojected probe direction, a random direction, and the J-space patch.

A layer counts as evidence for a handoff outside the workspace only if the perpendicular effect is significant *and* exceeds the random control *and* survives Holm correction across that model's probed layers (Appendix B). We compare with the random control using a paired contrast where it was recorded, and otherwise require non-overlapping intervals. Counted layers are marked and shown in bold. All intervals are bootstrap 95% confidence intervals over held-out items.

In Gemma-3-4B, the perpendicular direction even shifts the model toward the original answer at three layers, spanning 65–79% of depth, where random directions have no effect. None of these layers, however, meets the strict counting criterion.

Qwen3-1.7B (interval separation).

depth	perpendicular	unprojected probe	random	J-space	counts
21%	0.16 [−0.36, 0.69]	0.19 [−0.34, 0.74]	0.16 [−0.05, 0.35]	0.58 [0.33, 0.84]	
36%	**0.73** [0.26, 1.25]	0.81 [0.34, 1.33]	−0.04 [−0.25, 0.14]	1.05 [0.80, 1.34]	yes
50%	**1.17** [0.44, 1.85]	1.03 [0.21, 1.81]	−0.12 [−0.26, 0.02]	−0.40 [−0.73, −0.08]	yes
57%	**1.64** [1.22, 2.08]	1.70 [1.23, 2.17]	−0.12 [−0.22, −0.01]	0.18 [0.02, 0.35]	yes
64%	**1.30** [0.98, 1.62]	1.31 [0.97, 1.65]	−0.05 [−0.16, 0.07]	0.10 [−0.05, 0.25]	yes
71%	**1.25** [0.98, 1.51]	1.29 [1.01, 1.55]	−0.08 [−0.13, −0.02]	0.04 [−0.10, 0.17]	yes
79%	−0.05 [−0.24, 0.14]	−0.08 [−0.28, 0.12]	0.03 [−0.02, 0.08]	0.01 [−0.13, 0.14]	
86%	−0.00 [−0.18, 0.17]	0.11 [−0.08, 0.30]	0.03 [−0.01, 0.08]	−0.07 [−0.22, 0.07]	
89%	0.19 [0.04, 0.34]	0.20 [0.01, 0.39]	0.02 [−0.01, 0.04]	−0.14 [−0.30, 0.02]	
93%	0.03 [−0.11, 0.18]	0.34 [0.03, 0.66]	0.01 [−0.01, 0.03]	0.00 [−0.26, 0.28]	

Qwen3.5-2B-Base (interval separation).

depth	perpendicular	unprojected probe	random	J-space	counts
25%	−0.02 [−0.10, 0.06]	−0.03 [−0.11, 0.06]	−0.01 [−0.03, 0.01]	0.50 [0.41, 0.59]	
38%	0.14 [0.02, 0.27]	0.16 [0.03, 0.29]	−0.01 [−0.03, 0.01]	0.30 [0.23, 0.37]	
46%	**0.26** [0.15, 0.38]	0.26 [0.14, 0.38]	−0.00 [−0.02, 0.02]	0.17 [0.11, 0.23]	yes
50%	**0.29** [0.17, 0.42]	0.28 [0.15, 0.41]	0.00 [−0.02, 0.02]	0.14 [0.07, 0.22]	yes
54%	**0.49** [0.36, 0.63]	0.46 [0.32, 0.60]	0.00 [−0.01, 0.02]	0.09 [0.02, 0.15]	yes
62%	**0.33** [0.26, 0.42]	0.33 [0.25, 0.41]	−0.01 [−0.03, 0.00]	0.01 [−0.02, 0.04]	yes
71%	−0.02 [−0.08, 0.05]	−0.01 [−0.07, 0.05]	−0.01 [−0.03, 0.01]	−0.01 [−0.03, 0.01]	
79%	−0.01 [−0.06, 0.04]	0.00 [−0.05, 0.06]	−0.00 [−0.01, 0.01]	0.00 [−0.02, 0.03]	
88%	−0.03 [−0.08, 0.03]	−0.00 [−0.06, 0.05]	0.00 [−0.00, 0.01]	0.01 [−0.01, 0.04]	

Qwen3-4B (paired contrast).

depth	perpendicular	unprojected probe	random	J-space	counts
22%	0.18 [−0.07, 0.44]	0.17 [−0.09, 0.42]	−0.08 [−0.19, 0.02]	0.03 [−0.08, 0.14]	
39%	0.13 [−0.04, 0.32]	0.12 [−0.06, 0.31]	0.05 [−0.04, 0.15]	0.04 [−0.11, 0.16]	
47%	0.16 [−0.09, 0.40]	0.14 [−0.13, 0.39]	0.04 [−0.03, 0.13]	−0.17 [−0.36, 0.02]	
53%	**0.44** [0.20, 0.69]	0.46 [0.21, 0.73]	−0.00 [−0.09, 0.07]	−0.18 [−0.47, 0.11]	yes
58%	**0.88** [0.62, 1.15]	0.96 [0.67, 1.25]	0.06 [−0.02, 0.14]	0.04 [−0.26, 0.35]	yes
64%	**0.67** [0.48, 0.87]	0.66 [0.46, 0.87]	−0.00 [−0.07, 0.06]	−0.22 [−0.43, −0.00]	yes
67%	0.11 [−0.02, 0.24]	0.14 [0.01, 0.26]	0.05 [−0.02, 0.11]	−0.36 [−0.61, −0.09]	
69%	0.11 [0.01, 0.21]	0.12 [0.03, 0.22]	0.02 [−0.04, 0.07]	−0.38 [−0.64, −0.11]	
75%	−0.04 [−0.12, 0.04]	0.01 [−0.07, 0.09]	0.05 [−0.01, 0.11]	−0.55 [−0.88, −0.22]	
81%	−0.01 [−0.11, 0.10]	−0.04 [−0.15, 0.08]	0.03 [−0.03, 0.09]	−0.72 [−1.12, −0.33]	
89%	0.13 [−0.02, 0.28]	0.16 [0.00, 0.32]	0.06 [−0.00, 0.13]	−1.39 [−1.92, −0.86]	

Gemma-2-2B (interval separation).

depth	perpendicular	unprojected probe	random	J-space	counts
23%	$0.06\ [-0.00, 0.12]$	$0.06\ [-0.00, 0.12]$	$0.01\ [-0.02, 0.03]$	$0.09\ [0.06, 0.13]$	
38%	$-0.02\ [-0.07, 0.04]$	$-0.02\ [-0.07, 0.04]$	$0.02\ [-0.00, 0.05]$	$-0.02\ [-0.06, 0.01]$	
50%	$0.06\ [-0.03, 0.15]$	$0.06\ [-0.03, 0.15]$	$-0.00\ [-0.02, 0.01]$	$-0.00\ [-0.05, 0.05]$	
54%	$-0.02\ [-0.10, 0.06]$	$-0.03\ [-0.11, 0.05]$	$0.00\ [-0.01, 0.02]$	$-0.04\ [-0.08, 0.01]$	
62%	$0.05\ [-0.01, 0.11]$	$0.05\ [-0.01, 0.11]$	$0.01\ [-0.01, 0.02]$	$0.01\ [-0.03, 0.04]$	
65%	$-0.02\ [-0.07, 0.03]$	$-0.01\ [-0.06, 0.04]$	$0.01\ [-0.01, 0.03]$	$-0.03\ [-0.06, 0.01]$	
69%	$-0.01\ [-0.07, 0.05]$	$-0.00\ [-0.06, 0.06]$	$-0.00\ [-0.02, 0.01]$	$-0.05\ [-0.09, -0.02]$	
77%	$0.01\ [-0.05, 0.07]$	$0.02\ [-0.04, 0.09]$	$0.01\ [-0.00, 0.03]$	$-0.03\ [-0.06, -0.00]$	
85%	$0.00\ [-0.06, 0.06]$	$0.04\ [-0.02, 0.10]$	$-0.01\ [-0.02, 0.00]$	$0.00\ [-0.02, 0.02]$	
88%	$-0.02\ [-0.08, 0.05]$	$0.02\ [-0.05, 0.10]$	$-0.00\ [-0.01, 0.01]$	$0.01\ [-0.01, 0.03]$	
92%	$-0.02\ [-0.08, 0.04]$	$0.01\ [-0.06, 0.08]$	$-0.00\ [-0.01, 0.01]$	$0.03\ [0.02, 0.05]$	

Gemma-3-4B (interval separation).

depth	perpendicular	unprojected probe	random	J-space	counts
24%	$0.10\ [0.04, 0.17]$	$0.09\ [0.02, 0.16]$	$0.17\ [0.00, 0.31]$	$0.35\ [0.23, 0.47]$	
38%	$0.08\ [0.02, 0.15]$	$0.11\ [0.04, 0.17]$	$-0.01\ [-0.16, 0.13]$	$0.10\ [0.06, 0.15]$	
47%	$0.04\ [-0.04, 0.12]$	$0.02\ [-0.06, 0.10]$	$-0.21\ [-0.33, -0.09]$	$0.08\ [0.02, 0.13]$	
53%	$-0.12\ [-0.20, -0.04]$	$-0.15\ [-0.23, -0.07]$	$-0.20\ [-0.32, -0.08]$	$-0.04\ [-0.11, 0.03]$	
59%	$-0.16\ [-0.24, -0.08]$	$-0.19\ [-0.27, -0.11]$	$-0.20\ [-0.31, -0.10]$	$-0.08\ [-0.15, -0.01]$	
65%	$-0.22\ [-0.31, -0.12]$	$-0.25\ [-0.36, -0.15]$	$-0.09\ [-0.20, 0.01]$	$-0.06\ [-0.13, 0.00]$	
71%	$-0.12\ [-0.19, -0.05]$	$-0.15\ [-0.22, -0.07]$	$-0.01\ [-0.12, 0.09]$	$-0.04\ [-0.10, 0.03]$	
79%	$-0.09\ [-0.15, -0.03]$	$-0.10\ [-0.17, -0.04]$	$0.08\ [-0.02, 0.18]$	$-0.18\ [-0.31, -0.07]$	
88%	$0.00\ [-0.03, 0.03]$	$0.01\ [-0.03, 0.04]$	$0.06\ [0.03, 0.09]$	$-0.36\ [-0.59, -0.15]$	

N LENS FIDELITY ON THE TASK

We evaluate how faithfully the lens reflects the model on the held-out items from §4.7. Measurements use the final token position at every layer probed in that experiment.

Output fidelity compares the lens's predicted output with the model's actual output. We measure $\mathrm{KL}(p_{\text{model}} \,\|\, p_{\text{lens}})$ and top-1 agreement with the model's own prediction. We evaluate both the J-lens and, as a reference, the logit lens.

Decision-gradient fidelity asks whether the lens identifies the directions that actually change the model's preference between answers. One backward pass gives the model's true decision gradient, $g = \partial[\log P(\text{alt}) - \log P(\text{ans})]/\partial h_\ell$. We measure its cosine similarity with two directions: the lens's own local gradient, computed through $W_U \,\mathrm{norm}(\bar{J}_\ell h)$, and the intervention vector $v_{\text{alt}} - v_{\text{ans}}$. We also measure the share of $\|g\|^2$ contained in the 25-dimensional active subspace removed by the projection in §4.7. We divide this share by k/d, the expected share for a random subspace, to express it relative to chance.

Across models, cosine similarity between the lens gradient and the true decision gradient is 0.01–0.04 before 45% of depth, compared with 0.02 for random directions. It is 0.10–0.18 between 45% and 75%, and 0.38–0.72 after that. The J-lens therefore tracks how the final position reaches its decision only in the last third of the network. We define the onset of tracking as the first probed layer where this cosine exceeds 0.2.

O RELATION DECODABILITY BY DEPTH

P READOUT BY LAYER

Q MANIPULATION CHECK

After each swap, we measure the improvement in the *destination* concept's log rank in the J-lens readout. This checks how strongly the intervention actually introduces its target concept into the workspace.

R WHOLE-PROMPT PATCHING

Patching every token position at once (Figure 3, right) produces a very large late-layer entity effect ($\Delta = 32.7\ [30.2, 35.2]$). A matched random direction reproduces much of this effect ($\Delta = 17.8\ [16.8, 18.8]$), because near the output any large residual edit can perturb the logits. Whole-prompt patching also dilutes the edge effect, since relation tokens account for only a minority of positions.

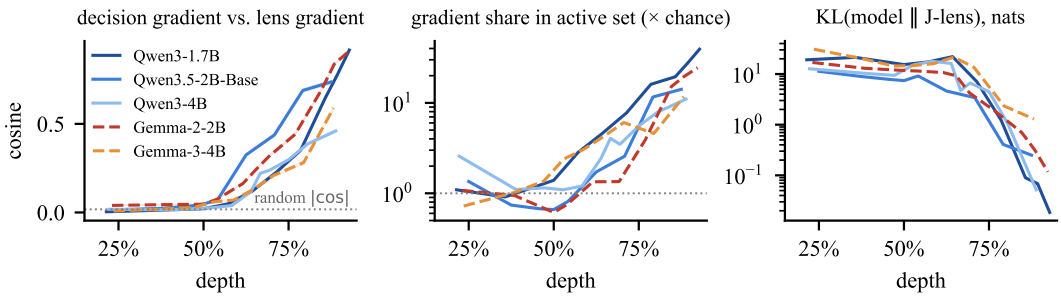


Figure 8: Lens fidelity across depth at the final position of held-out items. **Left:** cosine similarity between the model's true decision gradient and the J-lens gradient. Bands show 95% CIs; the dotted line shows the typical $|\cos|$ for a random direction. **Middle:** share of the decision gradient contained in the 25-dimensional active subspace, relative to chance. **Right:** KL divergence from the model's output distribution to the J-lens distribution. Solid lines represent Qwen models; dashed lines represent Gemma models.

Table 8: Lens fidelity averaged over depth bands (early $< 45\%$, middle 45–75%, late $> 75\%$). g: the model's decision gradient; v: the intervention vector $v_{\text{alt}} - v_{\text{ans}}$.

model	depth	$\cos(g, \nabla\text{lens})$	$\cos(g, v)$	share / chance	KL, J-lens	KL, logit lens	top-1, J-lens
Qwen3-1.7B	early	+0.008	+0.007	1.0	20.2	11.4	0.00
	middle	+0.104	+0.097	4.2	15.5	12.5	0.01
	late	+0.672	+0.462	25.5	0.4	0.9	0.91
Qwen3.5-2B-Base	early	+0.020	+0.020	1.0	10.1	15.0	0.00
	middle	+0.183	+0.179	1.3	6.5	13.8	0.15
	late	+0.712	+0.707	12.9	0.3	1.9	0.95
Qwen3-4B	early	+0.017	+0.015	1.8	11.5	13.9	0.00
	middle	+0.110	+0.099	2.2	11.7	12.9	0.04
	late	+0.381	+0.330	8.2	1.9	6.2	0.64
Gemma-2-2B	early	+0.042	+0.036	1.0	15.0	30.0	0.00
	middle	+0.168	+0.142	1.1	9.5	19.5	0.09
	late	+0.718	+0.631	15.8	0.7	1.8	0.92
Gemma-3-4B	early	+0.014	+0.008	0.9	25.4	11.5	0.00
	middle	+0.099	+0.076	3.4	16.0	7.1	0.15
	late	+0.435	+0.345	8.1	1.8	0.9	0.90

Both artefacts disappear when patches are restricted to the relevant position class. All claims in §4.3 therefore rely on these position-restricted interventions.

S Lens-specificity control

We repeat the position-restricted swap experiment with the Jacobian transport replaced by the identity, yielding the logit-lens control. Bold values mark intervals that exclude zero.

T Order of processing across models

Order of processing. Combining the causal and readout measurements across models reveals a consistent sequence. On average, relation swaps stop affecting the answer at 60% of the model's depth. The answer entity reaches the top of the readout only at 74%, and the bridge entity does so later still.

The edge thus stops being causally effective in the workspace before either entity it relates has been resolved. This timing is consistent with a system that reads a relation from the broadcast channel, applies it elsewhere, and returns only the result.

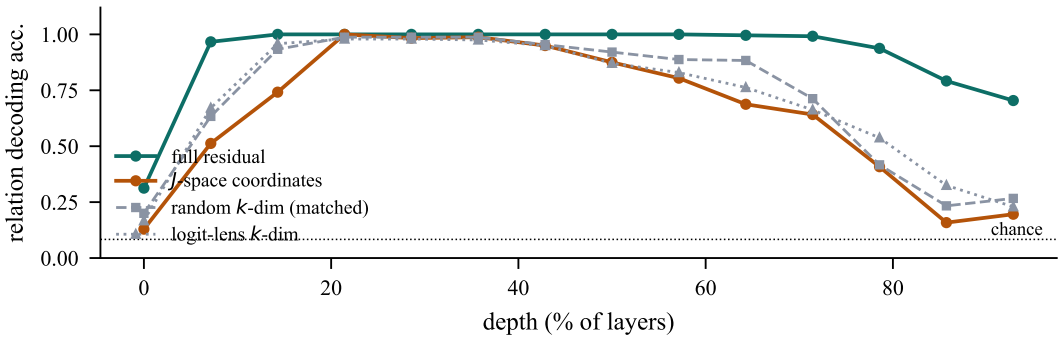


Figure 9: Accuracy of decoding the queried relation from the final token position across model depth. The relation remains decodable from the full residual through to the output, while decoding from workspace coordinates falls to near chance. The dimension-matched control shows that the workspace never carries relation identity better than an arbitrary subspace of the same size.

layer	edge swap installs rel.	node swap installs ent.
6	1.34 [1.09, 1.63]	0.64 [0.46, 0.84]
10	2.16 [1.87, 2.47]	0.47 [0.30, 0.64]
14	1.51 [1.25, 1.78]	0.68 [0.45, 0.89]
18	2.34 [2.15, 2.53]	0.94 [0.77, 1.11]
22	1.66 [1.48, 1.87]	4.22 [3.78, 4.65]
25	1.87 [1.60, 2.15]	2.51 [2.17, 2.86]

U POSITION MAP

We read the J-lens at every token position and every fitted layer. For each concept and position class, the table reports the best (lowest) median rank reached anywhere in that class.

Both entities and relations reach the top of the readout at their own token positions, confirming that the lens can detect relations. Their difference lies in propagation: entities reach the final position where the answer forms, whereas relations do not.

concept	relation positions	entity positions	final position
answer entity	1110	**1**	**1**
bridge entity	421	**2**	**6**
distractor entity	1878	**1**	**19**
queried relation	**1**	342	374
hop-1 relation	**1**	734	2938
counterfactual relation	**1**	358	2590

layer	interaction, J-lens	interaction, logit lens
6	$\mathbf{1.76}$ [1.23, 2.33]	-0.07 [−0.29, 0.13]
10	$\mathbf{2.47}$ [1.96, 2.98]	0.03 [−0.14, 0.19]
14	$\mathbf{0.88}$ [0.62, 1.16]	0.01 [−0.13, 0.16]
18	$\mathbf{0.85}$ [0.69, 1.01]	0.04 [−0.04, 0.14]
22	$\mathbf{0.27}$ [0.20, 0.33]	0.04 [−0.02, 0.10]
25	$\mathbf{0.26}$ [0.20, 0.32]	$\mathbf{0.10}$ [0.03, 0.17]

Table 9: Dictionary atoms selected by non-negative OMP over the full vocabulary, compared with the top-k tokens by lens score at the same position and layer.

layer	resid.	NN-OMP atoms	top-k by lens score
4	0.88	.cont, end, <U+6984>, <U+0970>	. \n, , \n,). \n, .\n
10	0.92	<U+FFFD>, belong, <U+018B>, <U+6CD5><U+5B98>	<U+FF02>, <U+300D>, <U+FF01><U+300D>, <U+00AD>
16	0.92	None, <U+B985>, uesto, <U+FFFD>	."\n\n\n, .\n\n\n, .\n, .\n
22	0.92	Mary, _; \n, Maya, <U+6CD5><U+5B98>	Maria, Mary, Mar<U+00ED>a, Maya