TYPED MECHANISM LIBRARIES: ORGANIZED ROUTING AND WEAK EDIT RESPONSE

Anonymous authors
Paper under double-blind review

ABSTRACT

Does organizing a model's internal state make that state behaviorally editable? We study this question in transformers augmented with a typed mechanism library: an addressable collection of slots with structural payloads and a reversible edit interface. Across 22.6M and 125M configurations, type supervision increases the association between evidence types and routing destinations. At 125M, a replication on fresh training seeds and worlds yields permutation-centered mutual information of 0.1025–0.1275 nats, compared with 0.0471–0.0573 for two unsupervised controls, and every paired contrast against these controls exceeds five permutation standard deviations. Yet structural edits produce little numerical response: the readout change is at most 3.4×10^{-6} in the 125M evaluation, with zero observed collateral change. Language-modeling loss stays within 0.01 nats of a dense baseline, and rollback recovers the saved state exactly. These results separate routing organization, state-level editability, and behavioral control, and an analysis of the architecture shows why they can come apart: when routing ignores payloads, organization is invariant to payload edits, and a null edit response arises either because the edited fields are not read or because the readout does not use them. Evaluations of explicit-state models should therefore report output sensitivity alongside internal organization and calibrate organization claims against unsupervised controls in each configuration.

1 INTRODUCTION

An identifiable representation of knowledge is a potential target for model editing, but identification and behavioral control are different requirements. Locate-and-edit methods seek weight updates that change specific factual predictions (Meng et al., 2022; 2023; Fang et al., 2025), learned editors map an editing signal to a parameter update (Mitchell et al., 2022), and memory-based editors store edits in explicit key–value or side memories (Hartvigsen et al., 2023; Wang et al., 2024). Interpretability methods ask related questions about what internal representations encode and how they participate in computation (Hewitt & Liang, 2019; Elhage et al., 2022; Wang et al., 2023). Across these settings, a central challenge is to establish that a representation is not only identifiable but also an effective intervention target.

Prior work shows that causal localization need not identify the most effective location for a weight edit (Hase et al., 2023), and critiques of the knowledge-neuron thesis (Dai et al., 2022) argue that localized parameter patterns give an oversimplified account of stored knowledge (Niu et al., 2024). We examine a complementary setting in which the editable object is explicitly represented, rather than located after training. Our transformers include a *typed mechanism library*: a finite collection of addressable slots, a routing head, and an interface for editing and restoring slot contents. This design makes the scope of a state update explicit, and it lets us ask directly whether the answer depends on the edited contents.

We separate three properties that such an interface can otherwise conflate. *Routing organization* measures the association between a token's evidence type and its destination slot. *State-level editability* concerns which stored values an operation changes and whether those changes can be undone. *Behavioral control* requires changes in the model's answers, ideally toward the intended edited target and across held-out queries. Our experiments measure these properties separately in

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.

synthetic structural causal worlds, where intervention targets and evidence types are available by construction. The setting is motivated by causal-reasoning benchmarks (Jin et al., 2023; 2024).

At both 22.6M and 125M parameters, type supervision induces slot–type association. The 125M result requires an explicit comparison with unsupervised controls: a block-structured routing prior also develops nonzero association. A replication using fresh training seeds and worlds confirms the increment over both controls, with paired contrasts of 5.0–7.9 permutation standard deviations.

The edit results reveal a different limitation. Structural payload edits produce very small numerical changes in the answer readout, including an upper bound of 3.4×10^{-6} in the 125M evaluation. The edit interface nevertheless restores state exactly, and the language-modeling loss gap stays below 0.01 nats. We use **routing/readout gap** to describe this coexistence of organized routing and weak response to the tested payload edits. The term is operational; Section 5 reduces the gap to two separately testable factors, whether the forward pass reads the edited fields and whether the readout uses what is read.

Our contributions are as follows.

1. We separate routing organization, state-level editability, and behavioral control in a transformer with explicit typed state, and show that organization is invariant to payload edits while the edit response factors into whether the edited fields are read and whether the readout uses them (Section 5).
2. We show that type supervision yields about twice the slot–type organization of the two designated unsupervised controls in a fresh replication, and that the association of the unsupervised baseline itself differs across collections, which makes per-configuration controls necessary for any organization claim.
3. We show that structural payload edits leave answers essentially unchanged, with readout change at most 3.4×10^{-6}, while the edited state is exactly restorable and language-modeling loss stays within tolerance of a dense baseline: explicit, organized, and reversible state is not by itself a lever on behavior.

2 Related Work

Model editing. ROME and MEMIT modify factual associations through structured updates to transformer weights (Meng et al., 2022; 2023); MEND learns to transform gradients into editing updates (Mitchell et al., 2022). Evaluations commonly distinguish success on the edited fact, generalization to related queries, and effects on unrelated predictions, using datasets such as CounterFact and zsRE (Meng et al., 2022; Levy et al., 2017). Subsequent evaluations find that edits often fail to propagate to logically entailed facts (Cohen et al., 2024; Zhong et al., 2023), that teacher-forced metrics overstate edit success relative to realistic generation (Yang et al., 2025), and that editing methods degrade under large streams of sequential edits (Gupta et al., 2024; Thede et al., 2025). Memory-based editors instead write edits to explicit discrete key–value adaptors or routed side memories while leaving the backbone intact (Hartvigsen et al., 2023; Wang et al., 2024); our study likewise concerns explicitly represented state, whose writable extent is specified by an interface. It complements these results rather than competing with established editing methods: state-level locality and rollback are different outcomes from successful factual editing.

Representation analysis and causal interventions. Probes measure accessible information in representations (Adi et al., 2017; Hewitt & Liang, 2019), superposition analyses and sparse dictionary learning study features in distributed activations (Elhage et al., 2022; Bricken et al., 2023), and circuit analyses study how components participate in computation (Olah et al., 2020; Wang et al., 2023). These approaches make different commitments about causality; an association measure alone does not establish the role tested by an intervention. We build a representation with supervised routing and evaluate association and edit response separately. The nonzero association in our unsupervised controls further motivates matching controls to the architecture and evaluation configuration.

Structured state and routing. External-memory systems (Graves et al., 2014; 2016), trainable key–value memory layers (Lample et al., 2019; Berges et al., 2025; Huang et al., 2025), object-centric slot representations (Locatello et al., 2020), and mixture-of-experts models (Shazeer et al., 2017; Fedus et al., 2022; Zhou et al., 2022; Dai et al., 2024) provide different forms of addressable

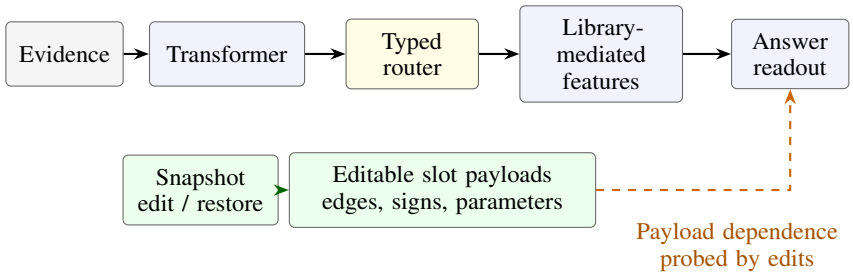


Figure 1: Schematic separating routing, editable payloads, and answer readout. The 125M configuration has 600 slots and 126.21M total parameters, including a 116.66M backbone. The dashed connection marks the dependence of answers on payload contents that the edit experiments probe.

or selectively accessed computation. Analyses of open mixture-of-experts models report domain- and token-level expert specialization (Muennighoff et al., 2025), and some architectures aim to make individual experts interpretable and removable (Park et al., 2025); such specialization is usually established by association between inputs and routing destinations, the kind of evidence our organization measure also provides. Even dense feed-forward layers have been read as implicit key–value memories (Geva et al., 2021), and modular architectures are often motivated by localized, interference-free updates (Pfeiffer et al., 2023); representational structure, however, does not imply identical editing semantics. Our mechanism library combines typed routing with explicit structural payloads in synthetic causal worlds. Causal-reasoning studies motivate the use of known causal structure as an evaluation resource (Jin et al., 2023; 2024; Kiciman et al., 2024); evidence-type supervision also relates to the concurrent study of evidence-type competition (Anonymous, 2026). Here, the focus is the relationship between routing organization and sensitivity to edits of the associated state.

3 Architecture and Setting

The model augments a transformer with typed routing and an editable mechanism library (Figure 1). During edit evaluation, the library is the only state the edit interface may write; the backbone remains fixed. This restriction concerns edit-time writes, not where the trained model may encode causal information.

3.1 Mechanism library and routing

The library contains $N = 200$ slots in the 22.6M configuration and $N = 600$ in the 125M configuration. Slots are partitioned into three and six evidence categories, respectively; the larger configuration uses identity, child, relation, sign, confidence, and a reserved category. Each slot carries a mechanism payload (edge lists, sign bits, and scalar parameters), together with usage counters included in the saved state. Parameter count, slot count, and type count change together between the two configurations.

Three routing modes define the main comparison. **Type** adds an auxiliary type-classification loss with weight $\lambda_g = 0.1$. **Emergent** uses free routing without that loss ($\lambda_g = 0$). **Blocks** uses a block-structured routing prior without type labels ($\lambda_g = 0$). The latter two quantify association that can arise without type supervision. Routing uses Gumbel noise (Jang et al., 2017; Maddison et al., 2017), a temperature schedule over the first 1,500 steps, a routing-mass floor parameter $\beta_{\text{floor}} = 0.3$, and a load-balancing term with weight $\lambda_{lb} = 0.01$.

3.2 Edit interface

An edit session snapshots the library, applies an operation, and either retains the edited state or restores the snapshot. The interface supports four structural operators—sign flip, edge addition, edge removal, and edge swap—and a bounded parameter update using a row mask, learning rate 10^{-3}, and at most 50 steps. Operator sequences are checked against library invariants. Restricting writes and

restoring all saved buffers provide state-level semantics, which we evaluate separately from output locality, efficacy, and generalization. The structural-response experiments test sign flips, and the restoration checks cover all five operators.

3.3 Training and configurations

Models are trained on online-sampled synthetic causal worlds with language-modeling, answer, gating, and load-balancing objectives. The 125M library model contains 126.21M parameters in total, of which 116.66M belong to the backbone. The shared recipe uses 40,000 steps and a learning rate of 6×10^{-4}, with 1,500 warmup steps and cosine decay to 6×10^{-5}; the Gumbel setting is 1.0. Modes share this recipe and differ in the routing prior and type-loss setting. The dense baseline, MONO, has approximately 125M parameters. Table 1 lists the 125M runs.

4 Evaluation Protocol

We measure association, intervention response, and state restoration separately. Table 5 summarizes the evaluation targets, and Appendix B describes the protocol sequence. The larger organization experiment contains an initial collection with 320 queries and a replication with 1,280 queries from 160 fresh worlds. The replication uses fresh training seeds and worlds disjoint from the initial evaluation set.

Routing organization. Let J denote a token's destination slot and T its evidence type. For $B = 2{,}000$ label permutations π_b, define

$$I_{\text{exc}} = \hat{I}(J;T) - \mu_\pi, \qquad \mu_\pi = \frac{1}{B}\sum_{b=1}^{B} \hat{I}(J;T_{\pi_b}), \qquad z = \frac{I_{\text{exc}}}{\sigma_\pi}, \tag{1}$$

where σ_π is the standard deviation of the permutation statistics (Pesarin & Salmaso, 2010). We call I_{exc} *permutation-centered MI*. Centering removes the association expected under the permutation scheme, and z is standardized against this empirical null.

Increment over unsupervised controls. A permutation null and a trained unsupervised control answer different questions. The first destroys slot–type association in a given routing output; the second measures association learned without the type objective. For type and control runs evaluated on matched examples, the paired contrast uses the same label permutation in both runs. Writing $D = \hat{I}_{\text{type}} - \hat{I}_{\text{control}}$ and D_b for its permuted counterpart, we report

$$\Delta_{\text{excess}} = D - \overline{D}_\pi, \qquad z_\Delta = \frac{\Delta_{\text{excess}}}{s_{D,\pi}}. \tag{2}$$

This contrast controls for the association that the baseline itself develops. Because the blocks control develops nonzero association, organization is assessed by these paired contrasts, and a fresh replication fixes its thresholds before data collection: for each type seed, $z \geq 3$ and $I_{\text{exc}} \geq 0.10$, plus $z_\Delta \geq 3$ against each of two controls. An additional blocks run is reported separately as a robustness comparison.

Edit response and collateral change. For library state L, edit ϵ, and targeted query set $\mathcal{Q}_\epsilon$, define

$$|\Delta\hat{y}| = \frac{1}{|\mathcal{Q}_\epsilon|}\sum_{x \in \mathcal{Q}_\epsilon} \left|\hat{y}(x;\epsilon(L)) - \hat{y}(x;L)\right|. \tag{3}$$

Collateral change is the corresponding average on non-target queries. Successful editing also requires the intended direction and generalization, so we report held-out editing probes separately.

Sources of variation. Permutation seeds measure the stability of the organization statistic, and training seeds and fresh worlds measure variation across runs; we report the two collections separately.

Table 1: Routing conditions and evaluation runs at the 125M configuration.

Mode	λ_g	Role	Training seeds
type	0.1	supervised routing	s0–s2 (initial); s3–s5 (replication)
emergent	0	free-routing control	s0 (initial); s1 (replication)
blocks	0	structured control	s0 (initial); s1 (replication); s2 (additional robustness comparison)
MONO	—	dense LM baseline	s0

5 Analysis of the Edit Response

The three properties of the introduction are measured by different functionals of the model. Making this explicit shows why a large organization score places no constraint on the edit response, why a null edit response admits exactly two explanations, and why raw mutual information cannot be compared across library configurations.

A factorization of the edit response. Write $r(x)$ for the routing distribution over slots, computed from backbone states, $\psi(x; L)$ for the library's contribution to the features that reach the answer readout, which may depend on both the routing weights and the payload contents L, and $\hat{y}(x; L) = f\big(h(x), \psi(x; L)\big)$ for the answer. Routing organization is a functional of r and the type labels alone, whereas the edit response of Eq. (3) is a functional of f and ψ. If routing logits do not depend on payload contents, then, for fixed routing noise and permutation draws, I_{exc} is invariant to every payload edit.

The edit response factorizes as follows. For any edit $L \to L'$, the answer is unchanged only if the forward pass does not read the edited fields, $\psi(x; L') = \psi(x; L)$, or the readout does not distinguish what is read, $f(h, \psi(x; L')) = f(h, \psi(x; L))$. For a differentiable payload and a small edit ΔL, the answer changes by

$$\hat{y}(x; L + \Delta L) - \hat{y}(x; L) = \nabla_\psi f\big(h(x), \psi\big)^\top \frac{\partial \psi(x; L)}{\partial L}\, \Delta L + O(\|\Delta L\|^2), \tag{4}$$

so a negligible response then requires either that the forward pass does not read the edited fields, $\frac{\partial \psi}{\partial L}\Delta L \approx 0$, or that the readout is insensitive to what is read, $\nabla_\psi f \perp \frac{\partial \psi}{\partial L}\Delta L$. For a fixed router, and hence a fixed I_{exc}, both factors can take any value.

Eq. (4) (derivation in Appendix A) locates the edit response in two factors: whether the edited fields are read, which depends on the implementation and on routing, and whether the readout uses what is read, which is a property of training. Organization, by contrast, does not determine either factor, which is why the organization and edit-response results must be reported separately rather than combined into a single editability claim.

Comparability of mutual information. Under independence of J and T, with all $K_J K_T$ joint cells occupied and $N \gg K_J K_T$ independent tokens, the plug-in estimate of $I(J; T)$ has a positive bias of approximately $(K_J - 1)(K_T - 1)/(2N)$ nats, where K_J and K_T are the numbers of occupied slots and types (Paninski, 2003). At a fixed number of tokens, moving from 200 slots and three types to 600 slots and six types multiplies this null bias by $(599 \cdot 5)/(199 \cdot 2) \approx 7.5$ when all slots are occupied. Permutation centering in Eq. (1) subtracts the null-expected part of the estimate within a configuration, but it does not equalize the variance of the null or the association that unsupervised routing develops. A threshold on I_{exc} or z calibrated in one configuration therefore cannot be carried to another, and the unsupervised controls must be re-measured wherever an organization claim is made. Section 6.1 shows that this is not a formality: the association developed by the unsupervised controls differs across collections.

Table 2: The three properties evaluated in this paper, the functional of the model that determines each, and the measurement that tests it.

Property	Determined by	Measured by	Result
Routing organization	router r and type labels; invariant to payload edits	I_{exc} and paired contrasts against unsupervised controls	§6.1
State-level editability	edit interface and saved state	restoration and contamination checks	§6.5
Behavioral control	payload read-out $\partial\psi/\partial L$ and readout sensitivity $\nabla_\psi f$	targeted and collateral answer change, held-out edit generalization	§6.3

Table 3: Initial 125M organization summaries (320 queries; 2,000 permutations per setting).

Setting	z	MI excess (nats)
type s0	6.09	0.141
type s1	7.75	0.164
type s2	5.62	0.134
emergent s0	1.09	0.019
blocks s0	2.38	0.050

6 RESULTS

6.1 ROUTING ORGANIZATION

22.6M configuration. Type-supervised routing produces slot–type association across three training seeds: $z = 4.47/5.99/7.02$, with $I_{\text{exc}} = 0.072$–0.124 nats (mean 0.0998). Both unsupervised controls lie within the original null screen (emergent $z = 0.21$; blocks $z = 0.14$). All type seeds thus exceed $z = 3$ in this configuration.

Initial 125M collection. At 125M, the three type seeds again show strong association: $z = 6.09/7.75/5.62$ and $I_{\text{exc}} = 0.141/0.164/0.134$ nats, respectively (320 queries; 2,000 permutations). The blocks control, however, has $z = 2.38$ and excess 0.050 nats, outside the original $|z| < 2$ screen; emergent has $z = 1.09$ and excess 0.019. Organization at this scale must therefore be measured relative to a control that itself develops association.

Calibration and paired contrasts. The diagnostic checks produce $z = -0.227$ for a random-initialization model and $z \in [-1.72, 1.36]$ across five label-shuffled checks.

Repeating the blocks calculation with five permutation seeds gives $z \in [2.34, 2.42]$ (SD 0.031). These checks are consistent with calibration under the tested nulls and show that the initial blocks elevation is stable to permutation-seed choice (Appendix F). The paired contrasts are positive for all six type-seed–control comparisons, with $\Delta_{\text{excess}} = 0.085$–$0.145$ nats. Five reach $z_\Delta \geq 3$, but type s2 versus blocks reaches only 2.68. Because one contrast falls short of 3 (Table 4, Appendix C), we ran a replication with fresh seeds and worlds, a larger query budget, and thresholds fixed before collection.

Fresh replication. With three fresh type seeds, two control runs, and 1,280 queries from 160 fresh worlds, every type seed exceeds both amplitude thresholds and every paired contrast exceeds $z_\Delta = 5$ (Table 6). The three type runs have excess 0.1025–0.1275 nats; emergent s1 and blocks s1 have excess 0.0471 and 0.0573, respectively, and raw $z = 7.58$ and 10.88. Thus the paired comparisons, rather than raw supervised scores alone, support an increment over these controls.

The additional blocks s2 run has greater association ($z = 12.05$, excess 0.0810). Against this run, the three paired scores are $4.53/2.21/3.86$, all positive, with two of three above 3.

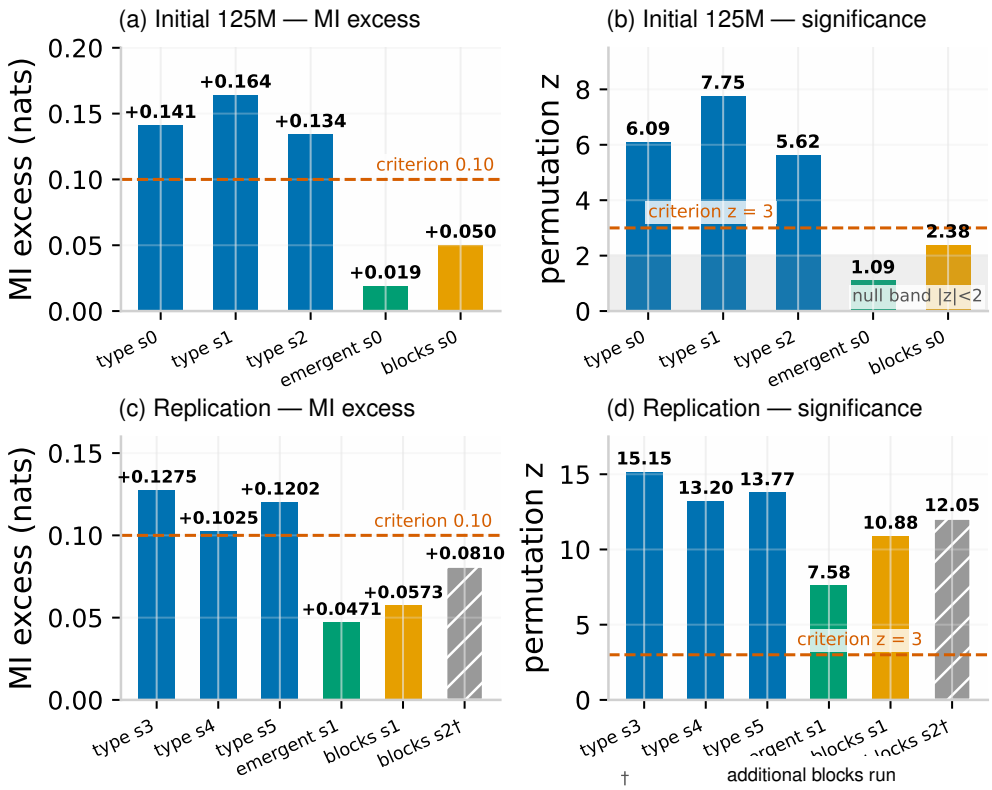


Figure 2: Routing organization in the initial 125M collection and the fresh replication. Type runs exceed the designated controls in both collections. The blocks control's initial $z = 2.38$ motivates the paired comparisons; both designated controls have nonzero association in the replication. Hatching denotes the additional blocks s2 comparison.

Table 4: Initial paired contrasts at 125M (320 queries). Paired excess is in nats; all six contrasts are positive.

Contrast	Δ_{excess}	z_Δ
s0 vs blocks	0.091	3.15
s0 vs emergent	0.122	4.47
s1 vs blocks	0.114	3.89
s1 vs emergent	0.145	5.21
s2 vs blocks	0.085	**2.68**
s2 vs emergent	0.116	3.92

6.2 Stability across seeds

Across the repeated permutation seeds, the largest SD of z is 0.20 (initial type s1). Across training seeds, the initial type excess spans 0.134–0.164 nats, with SD approximately 0.015. The replication range is 0.1025–0.1275. Both collections place the type runs above the measured unsupervised controls; we keep them separate because the query budget, worlds, and training seeds differ (Figure 8, Appendix F).

Table 5: Calibration checks: all null-case scores are inside $|z| < 2$.

Null check	z	MI excess (nats)
random-initialization checkpoint	-0.227	-0.0015
blocks s0 label shuffle 1	1.355	0.0284
blocks s0 label shuffle 2	-0.151	-0.0032
blocks s0 label shuffle 3	-1.720	-0.0365
blocks s0 label shuffle 4	0.205	0.0044
blocks s0 label shuffle 5	-0.924	-0.0196

Table 6: 125M replication on 160 fresh worlds and 1,280 queries, using 2,000 paired permutations. Every score exceeds its threshold.

Measure	s3	s4	s5	Threshold
Amplitude z	15.15	13.20	13.77	≥ 3
I_{exc} (nats)	0.1275	0.1025	0.1202	≥ 0.10
z_Δ vs emergent s1	7.87	5.92	7.29	≥ 3
z_Δ vs blocks s1	7.26	5.02	6.64	≥ 3

6.3 Edit response

Successful state edits need not produce a useful answer change. In the earlier structural-edit probe, all 150 attempted flips are applied, yet the readout change is at most 2×10^{-4}, with zero observed collateral change. In the 125M evaluation, paired flip edits over 150 queries per seed produce a readout-change bound of 3.4×10^{-6} across three seeds, well below the 10^{-3} threshold (Figure 4). These results establish weak sensitivity to the tested structural edits. Comparing the numerical magnitudes of MI (nats) and answer change would not quantify the gap: these quantities have different units.

The parameter-update probes reveal a separate generalization limitation (Figure 3). The 22.6M row-update evaluation gives 0.855 at the edited training point and 0.124 on held-out variants. At 125M, training-point gains are $0.884/0.935/0.917$, while the held-out generalization probe has a three-seed mean of 0.217: parameter edits change the edited point but transfer weakly to variants.

6.4 Locality and language-modeling cost

Off-target collateral is exactly 0.0 at the numerical precision of all three 125M evaluations. Because the intended target also barely changes, this is evidence of output stability under these edits, not selective behavioral editing; state locality follows from the restricted write interface.

The 125M type runs have language-modeling loss gaps of $0.0058/0.0082/0.0071$ nats relative to the approximately parameter-matched dense baseline. All are below 0.01 nats, and the model answers all 50 items of a behavioral battery correctly. These results show limited loss degradation.

6.5 Restoration

For each of three seeds, all 250 single-edit restores, all 1,000 depth-20 rollbacks, and all 1,250 backbone comparisons recover the saved state exactly. All tested failed edits leave the saved state uncontaminated (147/150/150 eligible instances across seeds), and all 250 function-level checks per seed satisfy the 10^{-6} tolerance. Table C2 gives the full counts. Including payloads and usage buffers in snapshots supports complete restoration of the saved library state.

7 Discussion and Limitations

The experiments establish that a library can exhibit organized routing and exact, reversible state operations while its answers barely respond to edits of its structural payloads. Type supervision increases slot–type association over matched unsupervised controls in the 125M replication, and

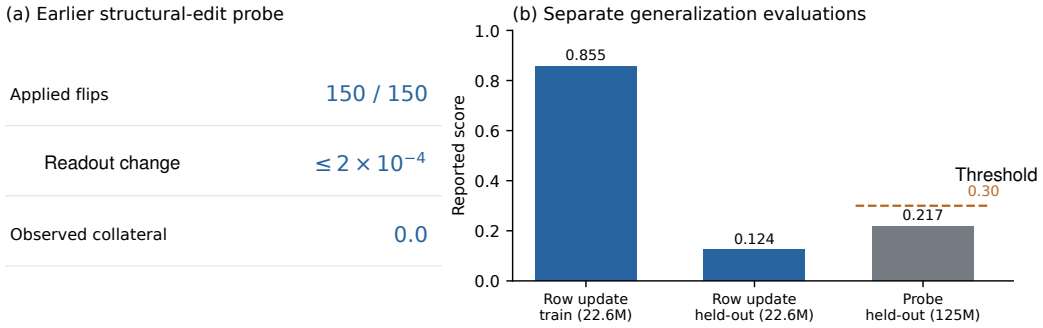


Figure 3: Edit-response and generalization results. (a) All 150 structural flips in the earlier probe are applied, with readout change at most 2×10^{-4} and zero observed collateral. (b) Row-update scores at 22.6M and the 125M held-out probe.

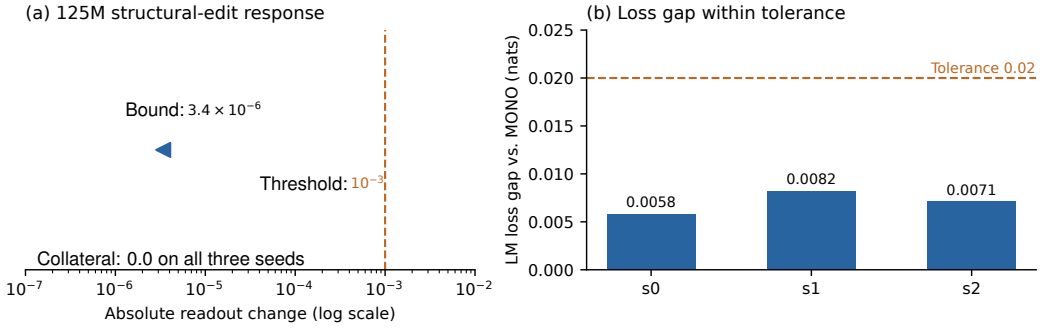


Figure 4: 125M response and language-modeling evaluations. (a) The bound on structural-edit response is below the 10^{-3} threshold; zero collateral is stated separately because zero is not representable on a logarithmic axis. (b) All three loss gaps are below 0.01 nats.

the edit interface restores state exactly, yet neither property makes the payload a lever on behavior. The factorization of Section 5 explains why these outcomes are separable and reduces the weak edit response to two measurable accounts: the forward pass may not read the edited fields, or the readout may not use what is read.

Implications for explicit-state models. Evaluation should report three outcomes separately: organization relative to configuration-specific controls, successful application and restoration of state edits, and the effect of those edits on targeted and held-out answers. The unsupervised runs develop different levels of association across configurations and collections, so an organization threshold calibrated in one setting cannot be reused in another. Explicit state becomes a behavioral lever only when the readout depends on it: sparse updates to memory-layer slots change model knowledge with limited forgetting (Lin et al., 2025), whereas the typed library here shows organized routing without such dependence, and Eq. (4) makes this dependence the quantity an explicit-state design must secure.

Limitations. The study uses synthetic worlds and two configurations of one architecture family, which give exact control over evidence types and edit targets and leave naturalistic knowledge editing for future work. The auxiliary type objective directly encourages the association that slot–type MI measures, which is why organization is assessed against unsupervised controls rather than in absolute terms. Parameter count, slot count, and type count vary together between the two configurations.

Ethics Statement

The experiments use synthetic causal-world data and involve no human participants or personal data. Model-editing tools can be used both to correct and to insert information, and auditable, reversible state is one natural safeguard for them. Our results caution that exact state restoration and organized routing do not by themselves provide control over model answers, so claims about safe model maintenance should be supported by behavioral evaluations.

Reproducibility Statement

The main text describes the model configurations, the edit interface, and the shared training recipe, defines the organization measure, the paired contrasts, and the edit-response metrics, and states the analysis and its assumptions. The appendix gives the derivation of the edit-response factorization, the protocol sequence and continued-training evaluation, the paired contrasts and diagnostics, the numerical summaries, and the evaluation thresholds and results.

AI Use Statement

We used generative AI tools to assist with writing and editing the manuscript, searching for and verifying related literature, and developing and debugging training and evaluation code. AI tools also assisted in deriving and writing up the edit-response factorization and its derivation; the authors checked every derivation and its connection to the data. We take responsibility for the final content of this paper, including text, claims, proofs, and code produced with the aid of generative AI.

References

Yossi Adi, Einat Kermany, Yonatan Belinkov, Ofer Lavi, and Yoav Goldberg. Fine-grained Analysis of Sentence Embeddings Using Auxiliary Prediction Tasks. In *International Conference on Learning Representations*, 2017.

Anonymous. When Observational Context Overrides Interventional Evidence in Transformers, 2026. Concurrent ICLR 2027 submission.

Vincent-Pierre Berges, Barlas Oguz, Daniel Haziza, Wen tau Yih, Luke Zettlemoyer, and Gargi Ghosh. Memory Layers at Scale. In *International Conference on Machine Learning*, 2025.

Trenton Bricken, Adly Templeton, Joshua Batson, Brian Chen, Adam Jermyn, Tom Conerly, Nicholas L. Turner, Cem Anil, Carson Denison, Amanda Askell, Robert Lasenby, Yifan Wu, Shauna Kravec, Nicholas Schiefer, Tim Maxwell, Nicholas Joseph, Alex Tamkin, Karina Nguyen, Brayden McLean, Josiah E. Burke, Tristan Hume, Shan Carter, Tom Henighan, and Chris Olah. Towards Monosemanticity: Decomposing Language Models With Dictionary Learning. Transformer Circuits Thread, 2023. URL `https://transformer-circuits.pub/2023/monosemantic-features/index.html`.

Roi Cohen, Eden Biran, Ori Yoran, Amir Globerson, and Mor Geva. Evaluating the Ripple Effects of Knowledge Editing in Language Models. *Transactions of the Association for Computational Linguistics*, 12:283–298, 2024.

Damai Dai, Li Dong, Yaru Hao, Zhifang Sui, Baobao Chang, and Furu Wei. Knowledge Neurons in Pretrained Transformers. In *Annual Meeting of the Association for Computational Linguistics*, 2022.

Damai Dai, Chengqi Deng, Chenggang Zhao, R. X. Xu, Huazuo Gao, Deli Chen, Jiashi Li, Wangding Zeng, Xingkai Yu, Y. Wu, Zhenda Xie, Y. K. Li, Panpan Huang, Fuli Luo, Chong Ruan, Zhifang Sui, and Wenfeng Liang. DeepSeekMoE: Towards Ultimate Expert Specialization in Mixture-of-Experts Language Models. In *Annual Meeting of the Association for Computational Linguistics*, 2024.

Nelson Elhage, Tristan Hume, Catherine Olsson, Nicholas Schiefer, Tom Henighan, Shauna Kravec, Zac Hatfield-Dodds, Robert Lasenby, Dawn Drain, Carol Chen, Roger B. Grosse, Sam McCandlish, Jared Kaplan, Dario Amodei, Martin Wattenberg, and Christopher Olah. Toy Models of Superposition. *arXiv preprint arXiv:2209.10652*, 2022.

Junfeng Fang, Houcheng Jiang, Kun Wang, Yunshan Ma, Jie Shi, Xiang Wang, Xiangnan He, and Tat-Seng Chua. AlphaEdit: Null-Space Constrained Knowledge Editing for Language Models. In *International Conference on Learning Representations*, 2025.

William Fedus, Barret Zoph, and Noam Shazeer. Switch Transformers: Scaling to Trillion Parameter Models with Simple and Efficient Sparsity. *Journal of Machine Learning Research*, 23(120):1–39, 2022.

Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. Transformer Feed-Forward Layers Are Key-Value Memories. In *Conference on Empirical Methods in Natural Language Processing*, 2021.

Alex Graves, Greg Wayne, and Ivo Danihelka. Neural Turing Machines. *arXiv preprint arXiv:1410.5401*, 2014.

Alex Graves, Greg Wayne, Malcolm Reynolds, Tim Harley, Ivo Danihelka, Agnieszka Grabska-Barwinska, Sergio Gomez Colmenarejo, Edward Grefenstette, Tiago Ramalho, John P. Agapiou, Adrià Puigdomènech Badia, Karl Moritz Hermann, Yori Zwols, Georg Ostrovski, Adam Cain, Helen King, Christopher Summerfield, Phil Blunsom, Koray Kavukcuoglu, and Demis Hassabis. Hybrid computing using a neural network with dynamic external memory. *Nature*, 538(7626): 471–476, 2016.

Akshat Gupta, Anurag Rao, and Gopala Anumanchipalli. Model Editing at Scale leads to Gradual and Catastrophic Forgetting. In *Findings of the Association for Computational Linguistics: ACL 2024*, 2024.

Tom Hartvigsen, Swami Sankaranarayanan, Hamid Palangi, Yoon Kim, and Marzyeh Ghassemi. Aging with GRACE: Lifelong Model Editing with Discrete Key-Value Adaptors. In *Advances in Neural Information Processing Systems*, 2023.

Peter Hase, Mohit Bansal, Been Kim, and Asma Ghandeharioun. Does Localization Inform Editing? Surprising Differences in Causality-Based Localization vs. Knowledge Editing in Language Models. In *Advances in Neural Information Processing Systems*, 2023.

John Hewitt and Percy Liang. Designing and Interpreting Probes with Control Tasks. In *Conference on Empirical Methods in Natural Language Processing*, 2019.

Zihao Huang, Qiyang Min, Hongzhi Huang, Yutao Zeng, Defa Zhu, Ran Guo, and Xun Zhou. Ultra-Sparse Memory Network. In *International Conference on Learning Representations*, 2025.

Eric Jang, Shixiang Gu, and Ben Poole. Categorical Reparameterization with Gumbel-Softmax. In *International Conference on Learning Representations*, 2017.

Zhijing Jin, Yuen Chen, Felix Leeb, Luigi Gresele, Ojasv Kamal, Zhiheng Lyu, Kevin Blin, Fernando Gonzalez Adauto, Max Kleiman-Weiner, Mrinmaya Sachan, and Bernhard Schölkopf. CLadder: Assessing Causal Reasoning in Language Models. In *Advances in Neural Information Processing Systems*, 2023.

Zhijing Jin, Jiarui Liu, Zhiheng Lyu, Spencer Poff, Mrinmaya Sachan, Rada Mihalcea, Mona T. Diab, and Bernhard Schölkopf. Can Large Language Models Infer Causation from Correlation? In *International Conference on Learning Representations*, 2024.

Emre Kiciman, Robert Osazuwa Ness, Amit Sharma, and Chenhao Tan. Causal Reasoning and Large Language Models: Opening a New Frontier for Causality. *Transactions on Machine Learning Research*, 2024.

Guillaume Lample, Alexandre Sablayrolles, Marc'Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. Large Memory Layers with Product Keys. In *Advances in Neural Information Processing Systems*, 2019.

Omer Levy, Minjoon Seo, Eunsol Choi, and Luke Zettlemoyer. Zero-Shot Relation Extraction via Reading Comprehension. In *Conference on Computational Natural Language Learning*, 2017.

Jessy Lin, Luke Zettlemoyer, Gargi Ghosh, Wen tau Yih, Aram Markosyan, Vincent-Pierre Berges, and Barlas Oğuz. Continual Learning via Sparse Memory Finetuning. *arXiv preprint arXiv:2510.15103*, 2025.

Francesco Locatello, Dirk Weissenborn, Thomas Unterthiner, Aravindh Mahendran, Georg Heigold, Jakob Uszkoreit, Alexey Dosovitskiy, and Thomas Kipf. Object-Centric Learning with Slot Attention. In *Advances in Neural Information Processing Systems*, 2020.

Chris J. Maddison, Andriy Mnih, and Yee Whye Teh. The Concrete Distribution: A Continuous Relaxation of Discrete Random Variables. In *International Conference on Learning Representations*, 2017.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and Editing Factual Associations in GPT. In *Advances in Neural Information Processing Systems*, 2022.

Kevin Meng, Arnab Sen Sharma, Alex J. Andonian, Yonatan Belinkov, and David Bau. Mass-Editing Memory in a Transformer. In *International Conference on Learning Representations*, 2023.

Eric Mitchell, Charles Lin, Antoine Bosselut, Chelsea Finn, and Christopher D. Manning. Fast Model Editing at Scale. In *International Conference on Learning Representations*, 2022.

Niklas Muennighoff, Luca Soldaini, Dirk Groeneveld, Kyle Lo, Jacob Morrison, Sewon Min, Weijia Shi, Evan Pete Walsh, Oyvind Tafjord, Nathan Lambert, Yuling Gu, Shane Arora, Akshita Bhagia, Dustin Schwenk, David Wadden, Alexander Wettig, Binyuan Hui, Tim Dettmers, Douwe Kiela, Ali Farhadi, Noah A. Smith, Pang Wei Koh, Amanpreet Singh, and Hannaneh Hajishirzi. OLMoE: Open Mixture-of-Experts Language Models. In *International Conference on Learning Representations*, 2025.

Jingcheng Niu, Andrew Liu, Zining Zhu, and Gerald Penn. What does the Knowledge Neuron Thesis Have to do with Knowledge? In *International Conference on Learning Representations*, 2024.

Chris Olah, Nick Cammarata, Ludwig Schubert, Gabriel Goh, Michael Petrov, and Shan Carter. Zoom In: An Introduction to Circuits. Distill, 2020. URL `https://distill.pub/2020/circuits/zoom-in/`.

Liam Paninski. Estimation of Entropy and Mutual Information. *Neural Computation*, 15(6):1191–1253, 2003.

Jungwoo Park, Young Jin Ahn, Kee-Eung Kim, and Jaewoo Kang. Monet: Mixture of Monosemantic Experts for Transformers. In *International Conference on Learning Representations*, 2025.

Fortunato Pesarin and Luigi Salmaso. *Permutation Tests for Complex Data: Theory, Applications and Software*. Wiley, 2010.

Jonas Pfeiffer, Sebastian Ruder, Ivan Vulić, and Edoardo M. Ponti. Modular Deep Learning. *Transactions on Machine Learning Research*, 2023.

Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc V. Le, Geoffrey E. Hinton, and Jeff Dean. Outrageously Large Neural Networks: The Sparsely-Gated Mixture-of-Experts Layer. In *International Conference on Learning Representations*, 2017.

Lukas Thede, Karsten Roth, Matthias Bethge, Zeynep Akata, and Thomas Hartvigsen. WikiBigEdit: Understanding the Limits of Lifelong Knowledge Editing in LLMs. In *International Conference on Machine Learning*, 2025.

Kevin Ro Wang, Alexandre Variengien, Arthur Conmy, Buck Shlegeris, and Jacob Steinhardt. Interpretability in the Wild: a Circuit for Indirect Object Identification in GPT-2 Small. In *International Conference on Learning Representations*, 2023.

Peng Wang, Zexi Li, Ningyu Zhang, Ziwen Xu, Yunzhi Yao, Yong Jiang, Pengjun Xie, Fei Huang, and Huajun Chen. WISE: Rethinking the Knowledge Memory for Lifelong Model Editing of Large Language Models. In *Advances in Neural Information Processing Systems*, 2024.

Wanli Yang, Fei Sun, Jiajun Tan, Xinyu Ma, Qi Cao, Dawei Yin, Huawei Shen, and Xueqi Cheng. The Mirage of Model Editing: Revisiting Evaluation in the Wild. In *Annual Meeting of the Association for Computational Linguistics*, 2025.

Zexuan Zhong, Zhengxuan Wu, Christopher D. Manning, Christopher Potts, and Danqi Chen. MQuAKE: Assessing Knowledge Editing in Language Models via Multi-Hop Questions. In *Conference on Empirical Methods in Natural Language Processing*, 2023.

Yanqi Zhou, Tao Lei, Hanxiao Liu, Nan Du, Yanping Huang, Vincent Y. Zhao, Andrew M. Dai, Zhifeng Chen, Quoc V. Le, and James Laudon. Mixture-of-Experts with Expert Choice Routing. In *Advances in Neural Information Processing Systems*, 2022.

A Derivation of Eq. (4)

The invariance of I_{exc} holds because the slot–type association is computed from routing destinations and labels, neither of which a payload edit changes when routing does not read payloads and the routing noise and permutation draws are held fixed. The dichotomy follows from $\hat{y} = f(h, \psi)$: if ψ is unchanged the answer is unchanged, and otherwise the answer is unchanged only if f takes the same value at both arguments. Equation (4) is the chain rule applied to $f \circ \psi$. For the last statement, f and ψ enter the answer but not the routing distribution, so they can be chosen freely once r is fixed; in particular, a readout that ignores ψ gives zero response at any organization level.

B Protocol Sequence and Continued-Training Evaluation

The 125M organization evaluation proceeds in four stages.

- **Initial collection.** Three type seeds and one run of each unsupervised control are evaluated on 320 queries with 2,000 permutations (Table 3). The edit-response and loss-gap thresholds are satisfied, and the blocks control reaches $z = 2.38$, outside the $|z| < 2$ screen for unsupervised controls.
- **Paired comparisons.** Because the blocks control develops association, each type seed is compared with each control under synchronous label permutations (Appendix C). Five of the six contrasts reach $z_\Delta \geq 3$; type s2 versus blocks reaches 2.68.
- **Fresh replication.** Thresholds are fixed before data collection (Section 4). Three fresh type seeds and two designated controls are evaluated on fresh worlds with 1,280 queries, and every score exceeds its threshold (Table 6). An additional blocks s2 run is reported separately as a robustness comparison.
- **Continued training.** Type s3/s4/s5 are trained from 40,000 to 45,000 steps (an additional 12.5%) with a two-segment learning-rate profile that includes a 17.9% restart increase; the $k = 0$ baselines are those of the replication.

The restoration checks (Table C2) cover single and stacked restores, backbone preservation, function-level restoration, empty-stack behavior, and failed-edit behavior on three seeds.

Continued training. After 5,000 additional steps, organization retention is 0.7986, 0.9579, and 1.1370 for type s3, s4, and s5, respectively; s4 and s5 exceed the 0.80 retention threshold, and s3 falls short of it by approximately 0.00018 nats. The corresponding amplitude scores remain $z = 11.29/11.09/13.90$. Each seed answers 50/50 capability items in both context modes, and language-modeling loss drifts are $-0.0008/+0.0061/-0.0023$, all within the $+0.02$ tolerance. Continued training therefore preserves capability, language-modeling loss, and strong organization amplitude on all three seeds. At 1,000 additional steps, retention ratios are $0.938/0.953/1.132$. At 5,000 steps, the control summaries give emergent s1 $z = 3.37$, excess 0.0614, and blocks s1 $z = 3.51$, excess 0.0529. A separate control reference gives $z = 3.89$ and excess 0.065; it is not pooled with the 1,280-query replication summary.

C Paired Contrasts and Diagnostic Sequence

The initial paired comparison uses 320 queries and 2,000 synchronous label permutations. All six paired excesses are positive (Table 4). Increasing the budget to 1,280 queries in the replication was projected to reduce the permutation SD from approximately 0.029 to 0.015.

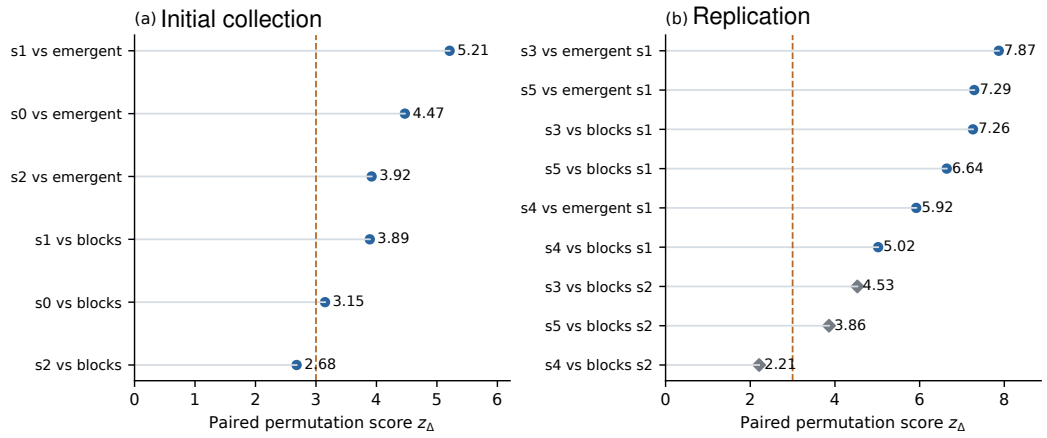


Figure 5: Paired standardized permutation scores z_Δ in the initial collection and the replication. All six replication contrasts against the two designated controls exceed $z_\Delta = 3$; against the additional blocks s2 run, two of three scores exceed 3.

D Numerical Summaries

Table C1: Permutation-seed sensitivity: original and four additional permutation seeds, conditional on each evaluated run.

Setting	Original z	Four additional z scores	SD	Range
type s0	6.09	6.06 / 6.13 / 6.00 / 6.03	0.048	[6.00, 6.13]
type s1	7.75	7.25 / 7.79 / 7.52 / 7.65	0.200	[7.25, 7.79]
type s2	5.62	5.56 / 5.30 / 5.55 / 5.53	0.105	[5.30, 5.62]
emergent s0	1.09	1.13 / 1.13 / 1.05 / 1.11	0.032	[1.05, 1.13]
blocks s0	2.38	2.41 / 2.42 / 2.34 / 2.39	0.031	[2.34, 2.42]

Table C2: Restoration checks: successful checks / eligible checks per seed.

Check	s0	s1	s2
Single-edit bit-exact restore (50 worlds $\times$ 5 operators)	250/250	250/250	250/250
Depth-20 stacked rollback	1000/1000	1000/1000	1000/1000
Unchanged-backbone comparisons	1250/1250	1250/1250	1250/1250
Function-level deviation $\leq 10^{-6}$	250/250	250/250	250/250
Empty-stack no-op	50/50	50/50	50/50
Failed-edit state preservation	147/147	150/150	150/150

Failed-edit preservation has 147 eligible instances for s0 because three constructed candidate edits were legal: the edge intended to be absent was present in those worlds. These instances therefore fall outside the failed-edit condition.

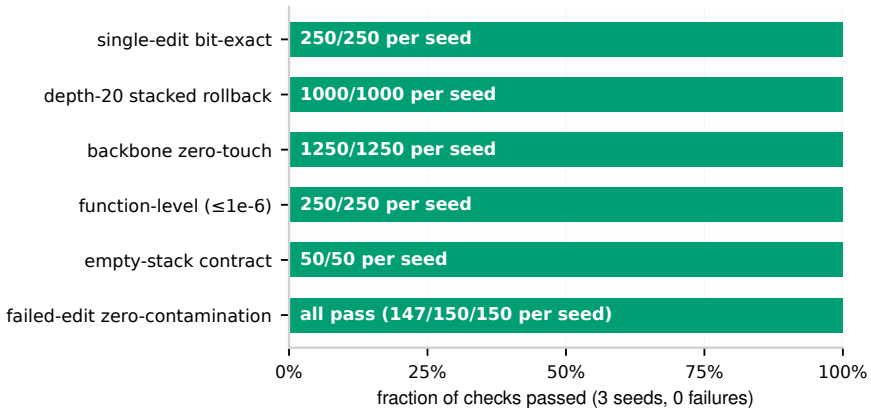


Figure 6: Restoration checks: fraction of eligible checks that succeed on each seed. Exact restoration covers the full saved library state.

E Evaluation Targets and Results

Table 5: Evaluation targets, thresholds, and results.

Evaluation target	Threshold	Result
22.6M routing organization	per-seed $z \geq 3$	$z = 4.47/5.99/7.02$; mean excess 0.0998 nats.
125M initial paired contrasts	every $z_\Delta \geq 3$	Five of six contrasts reach 3; s2 vs blocks reaches 2.68.
125M replication	each amplitude $z \geq 3$, excess ≥ 0.10; each designated contrast $z_\Delta \geq 3$	All scores exceed thresholds; $z_\Delta = 5.02$–7.87. Additional blocks s2: $z_\Delta = 4.53/2.21/3.86$.
125M structural response	readout change $\leq 10^{-3}$	Bound 3.4×10^{-6} across three seeds.
125M collateral	change < 0.01	0.0 on three seeds.
LM loss and battery	gap ≤ 0.02; battery $\geq 95\%$	Gap ≤ 0.0082 nats; 50/50 items.
Saved-state restoration	exact restore; function-level deviation $\leq 10^{-6}$	All checks succeed on three seeds (Table C2).
Unsupervised controls	descriptive comparison	Nonzero association at 125M, differing across collections.
Parameter-update editing	edited point and held-out variants	125M training point $0.884/0.935/0.917$; held-out mean 0.217.
Continued-training retention	each ratio ≥ 0.80	$0.7986/0.9579/1.1370$; capability 50/50 items, loss drift within $+0.02$.

F Supplementary Organization Analyses

Figure 7 displays the calibration and permutation-seed checks in Tables 5 and C1. They cover the tested null cases and the Monte Carlo sensitivity to the permutation seed. Figure 8 places the two collections side by side; each collection is analyzed separately.

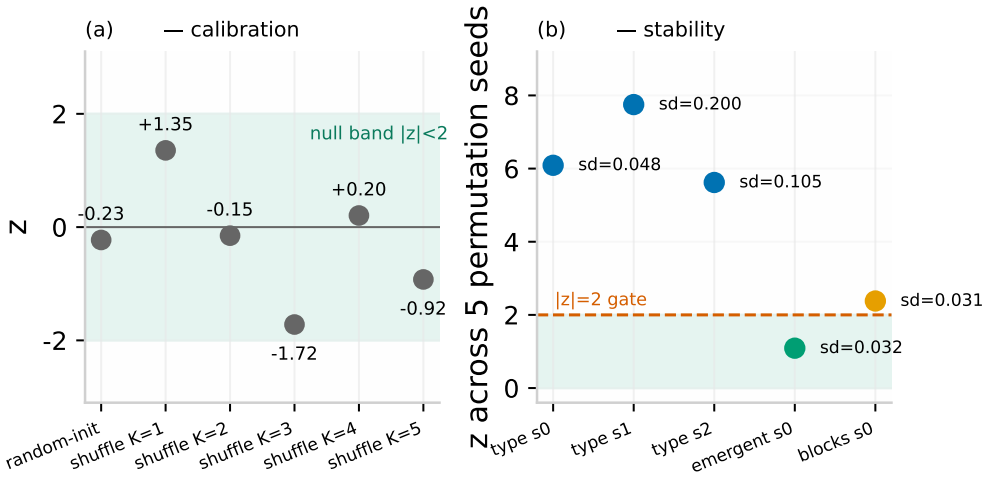


Figure 7: Calibration cases and sensitivity to permutation seed. (a) The tested null cases lie within $|z| < 2$. (b) The initial blocks score remains above 2 across five permutation seeds.

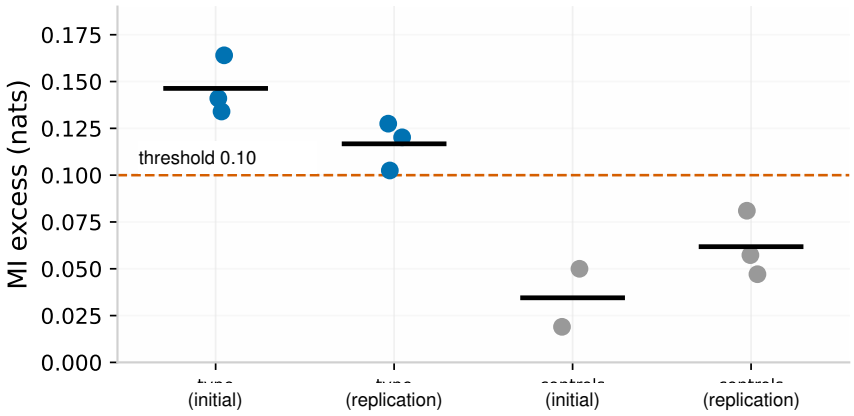


Figure 8: MI excess in the two collections, shown side by side. Each point is one run. The initial type range is higher than the replication range, and the unsupervised controls also differ across collections.