EXACT ROUTING-MARGIN ACCOUNTING IN SHARED-EXPERT TRANSFORMERS

Anonymous authors
Paper under double-blind review

ABSTRACT

Post-hoc circuit tracing can require trained surrogate models whose reconstruction error complicates mechanistic analysis. Building on router-score decompositions and top-k selection margins, we study an alternative using cross-layer shared-expert Transformers whose linear routers read the raw residual stream. The difference in linear-router scores between the weakest selected expert and its strongest excluded competitor decomposes exactly over preceding residual writes. This yields a routing-decision graph without a trained explainer. Unlike contributions to individual router scores, these margin contributions are invariant to common shifts of the router weights. We verify the decomposition in models from 4.9M to 81M parameters, with raw-score relative errors of approximately 10^{-6} and boundary-margin absolute errors at or below 4.2×10^{-4}. We then use this accounting to test when routing evidence supports causal conclusions. Routing-related two-dimensional residual subspaces have large effects under ablation, increasing cross-entropy by 1.31–3.34 nats, but subspaces derived from positive, negative, and total margin contributions largely coincide and have nearly identical effects. Their apparent specificity further weakens under dose-matched controls, and, under the tested interventions, changing expert identity at a single router has little effect on mean cross-entropy. These results establish margin decomposition as an exact, shift-invariant instrument for analyzing expert selection.

1 INTRODUCTION

Many modern language models use mixture-of-experts (MoE) layers, in which a router selects a small subset of feed-forward experts for each token (Shazeer et al., 2017; Fedus et al., 2021). These choices determine which parameters process the token, yet an individual choice is difficult to explain: when expert e is selected while expert j is not, which preceding model components built the score gap between them, which are necessary for the routing decision, and which affect downstream predictions?

Existing methods answer related questions at different levels. Population-level routing statistics can fail to predict token-level intervention effects (Engmann et al., 2026). General circuit-discovery methods either search directly with causal interventions or construct learned feature spaces and attribution graphs using sparse autoencoders and cross-layer transcoders (Bricken et al., 2023; Lindsey et al., 2025). Learned surrogates are powerful, but are typically fitted for the representations under study and introduce reconstruction error. More directly for MoEs, Li et al. (2026) decompose router scores over upstream components, and Zhu et al. (2026) use selection margins as geometric constraints that preserve top-k routing. Together, these results establish that router scores decompose over upstream components, common score shifts preserve expert ordering, and selected–excluded score gaps govern top-k stability. We ask what these properties permit at the level of individual component contributions and whether those contributions survive intervention.

Building on these observations, we construct an exact residual-write accounting of the tight top-k boundary. Our testbed is a cross-layer shared-expert Transformer: attention remains layer-specific, while a bank of expert parameters is reused across depth. A linear router reads the raw residual stream, the additive sum of the embedding and all preceding attention and expert writes. Sharing makes a repeated expert identity refer to the same parameterized function across depth, but the decomposition does not require it. Controls with layer-specific routers and expert banks test this dependence directly.

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.

We compare the weakest selected expert with the strongest excluded competitor under the scores used for routing. Their executed-score difference is the tightest selected–excluded inequality at the observed routing state. Its residual-evidence term decomposes exactly into one contribution per preceding write, while the routing-policy bias difference is recorded separately. A common shift of every router row cancels from every residual-evidence edge, so the accounting is shift-invariant. Representing each preceding write as a source node and its contribution as a weighted edge to the boundary decision gives the routing-decision graph shown in Figure 1.

This graph serves as a diagnostic and experimental-design tool. For an individual token, it shows which earlier attention and MoE writes widen or narrow the closest top-k gap, and separates evidence carried by the residual stream from the effect of the routing policy. Aggregated across tokens and layers, the same accounting identifies source components that repeatedly contribute to tight routing decisions. These edges can prioritize writes for targeted patching or ablation without fitting a surrogate. They describe the observed forward pass; whether a candidate write causally affects routing or model output must be tested by intervention.

Our contributions are:

1. **Exact, shift-invariant routing accounting.** We decompose the residual-evidence term of the tight executed margin over residual writes, with the balancing-policy difference kept separate. The identity requires no trained explainer and is verified to numerical precision on held-out text in models from 4.9M to 81M parameters.
2. **Non-identifiability of raw-score sign explanations.** Common score offsets preserve top-k order but can dominate the signs of individual score contributions. In a controlled example, removing a common component changes apparent negative-evidence mass from 85.9% to 8.5% (Appendix D). Raw-score decompositions still contain the information needed to compare experts, but their signs are not themselves selection-invariant evidence—a caution that applies directly to sign-based promoting–inhibiting readings of decomposed router scores (Li et al., 2026).
3. **An interventional account of scope.** Score-level removal of a dominant supporting term frequently changes the recorded top-k set, but sparse evidence for one inequality does not recover the complete route. Two-dimensional residual subspaces estimated from margin contributions can be behaviorally load-bearing under ablation, yet their effects are not sign-specific and are not uniquely distinguished from dose-matched random controls. Factorized routing interventions further separate expert identity, gate weights, and residual content. Together, these results show that exact accounting generates causal hypotheses but does not itself identify a behavioral circuit.

2 Related work

Router decomposition and selection margins. Using signed score differences to locate decision boundaries is classical in margin-based classification (Cortes & Vapnik, 1995). In mechanistic interpretability, Wang et al. (2023) similarly use the logit difference between competing output tokens as a contrastive target for tracing Transformer components. Our setting moves this idea from the output decision to an internal top-k routing decision.

Li et al. (2026) recursively decompose router inputs in existing MoE language models into upstream component scores, study promoting and inhibiting effects, and note that components constant across experts cannot change their order. Their component analysis is primarily observational, although they also use output-level path patching to identify function heads and compare those heads with routing-score patterns. We build directly on their score decomposition rather than presenting decomposition itself as new.

Selection margins are also established objects in MoE routing. LibMoE uses the top-1–top-2 router-score gap as a descriptive measure of router confidence (Nguyen et al., 2026). For general top-k routing, Zhu et al. (2026) define, for each excluded expert, the minimum gap to any selected expert and constrain router updates not to cross it, thereby preserving the selected set during machine unlearning. Our tight boundary—the weakest selected expert against the strongest excluded expert—is the globally smallest member of this family. Neither work decomposes this boundary over residual writes or uses the resulting terms for circuit analysis. Our contribution relative to these lines is

exact, shift-invariant residual-write accounting of the tight boundary, followed by interventions on the resulting terms.

Routing control and causal evaluation. Ye et al. (2026) split hidden states into router-visible control and router-blind content components across six MoE architectures. Their control channel is derived from the full router row space at each layer, whereas the routing-related subspace we ablate in Section 4 is estimated behaviorally from signed margin contributions and is subjected to sign-specificity, dose-matching, and route-mediation controls they do not test. Their trajectory analysis treats a path as the sequence of layer-specific expert identities, so it does not require a shared expert bank. Yoon et al. (2026) compare executed routes with equal-compute alternatives and show that route quality is token-conditional; importantly, a small router-only update also shifts downstream pass@K, providing an existence result for behaviorally relevant rerouting. Engmann et al. (2026) show that the population-level utilization, norm, and routing-weight statistics they test generally fail to predict token-level expert-ablation effects. These results motivate our separation of observed-forward accounting, route intervention, and downstream behavior; they do not imply that all router interventions must have small effects.

Post-hoc feature and circuit methods. Sparse autoencoders learn overcomplete feature dictionaries (Bricken et al., 2023); cross-layer transcoders support attribution graphs in a learned replacement model (Lindsey et al., 2025); and RouteSAE shares a dictionary across layers (Shi et al., 2025). These methods answer broader semantic and computational questions than our routing graph. Their surrogates are typically fitted for the activations or checkpoints under study, and reconstruction error must be measured when transferring conclusions back to the original model. For one routing boundary, we instead operate directly on the original router.

Sharing computation across depth. Universal and looped Transformers reuse blocks across depth (Dehghani et al., 2019; Yang et al., 2024); MoEUT combines recurrent depth with MoE feed-forwards (Csordás et al., 2024); Jaggi (2026) tie expert parameters across consecutive layers of production-style MoEs while keeping routing and attention layer-wise; and MoRE shares enlarged expert pools within adjacent reuse blocks while retaining layer-specific routers and adding depth embeddings (Qiu et al., 2026). More closely related to global reuse, MoUE combines layer-local experts with a pool of Universal Experts that can be invoked at multiple depths. It constrains their availability through a staggered topology and introduces reuse-aware load balancing and a Universal Router with trajectory state (Chen et al., 2026). These architectures motivate cross-layer reuse as a way to avoid duplicating similar layer-local functions and to compose expert choices across depth, but they optimize parameter efficiency and language-model performance rather than routing interpretability. Our testbed instead exposes the same complete expert bank at every layer and uses a stateless linear router, with layer-specific attention. Reusing the bank makes an expert ID denote the same parameterized function across depth; sharing the router additionally fixes the comparison directions.

3 Exact routing-decision accounting

3.1 Model and routing setup

Let $h_\ell \in \mathbb{R}^d$ denote the raw residual stream immediately before the MoE call at layer ℓ. It is an additive sum of the token embedding and all preceding attention and MoE writes ($i \prec \ell$ indexes the upstream writes in execution order):

$$h_\ell = h_{\text{emb}} + \sum_{i \prec \ell} \Delta h_i. \tag{1}$$

A linear router with rows $w_{\ell,e}$ assigns expert e an evidence score $s_{\ell,e} = w_{\ell,e}^\top h_\ell$. The executed selection score may additionally contain a balancing bias $b_{\ell,e}$,

$$\tilde{s}_{\ell,e} = s_{\ell,e} + b_{\ell,e}, \qquad S_\ell = \text{TopK}(\tilde{s}_\ell, k). \tag{2}$$

In our models, the bias affects expert identity but not the mixture weights: for $e \in S_\ell$,

$$g_{\ell,e} = \text{softmax}(s_\ell|_{S_\ell})_e, \qquad \Delta h_\ell^{\text{moe}} = \sum_{e \in S_\ell} g_{\ell,e} F_e(h_\ell). \tag{3}$$

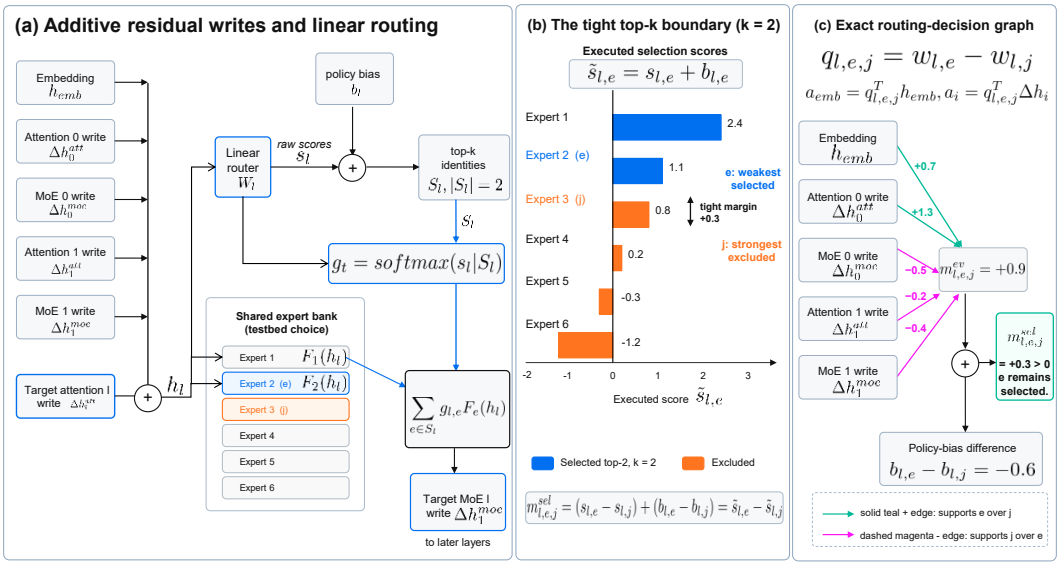


Figure 1: From additive residual writes to an exact routing-decision graph. (a) The residual stream h_ℓ is the sum of the embedding and all preceding attention and MoE writes. A linear router produces raw scores s_ℓ; the executed top-k set S_ℓ is selected using $s_\ell + b_\ell$, while the selected raw scores determine the normalized mixture weights in our models. (b) The weakest selected expert e and strongest excluded expert j define the tightest selected–excluded inequality in the observed routing state. (c) Projecting each preceding residual write onto $q_{\ell,e,j} = w_{\ell,e} - w_{\ell,j}$ exactly decomposes the evidence margin. Adding the policy-bias difference recovers the executed selection margin. Positive and negative edges support e and j, respectively, within this pairwise comparison.

Figure 1 summarizes this forward computation and the accounting construction introduced below.

Our primary models reuse one expert bank across depth while keeping attention blocks layer-specific. Routers may be shared or layer-specific. This design makes repeated expert decisions easy to compare.

3.2 From routing scores to boundary accounting

Following the router-score decomposition of Li et al. (2026), linearity and Equation 1 give an exact decomposition of an individual score,

$$s_{\ell,e} = w_{\ell,e}^\top h_{\text{emb}} + \sum_{i \prec \ell} w_{\ell,e}^\top \Delta h_i. \tag{4}$$

But the sign or magnitude of $s_{\ell,e}$ is not intrinsic to a top-k decision. Adding the same vector u to every router row adds $u^\top h_\ell$ to every score and preserves their ordering (Figure 2).

For a selected expert e and excluded competitor j, let $q_{\ell,e,j} = w_{\ell,e} - w_{\ell,j}$ and define the evidence margin

$$m^{\text{ev}}_{\ell,e,j} = q_{\ell,e,j}^\top h_\ell \tag{5}$$

$$= \underbrace{q_{\ell,e,j}^\top h_{\text{emb}}}_{a_{\text{emb}}} + \sum_{i \prec \ell} \underbrace{q_{\ell,e,j}^\top \Delta h_i}_{a_i}. \tag{6}$$

The selection margin is $m^{\text{sel}}_{\ell,e,j} = m^{\text{ev}}_{\ell,e,j} + b_{\ell,e} - b_{\ell,j}$. We keep the bias difference separate because it is a policy term rather than evidence written into the residual stream.

Selection margins have previously been used as constraints for preserving top-k routes (Zhu et al., 2026). For each excluded expert, GRIP uses its minimum gap to any selected expert. Our default

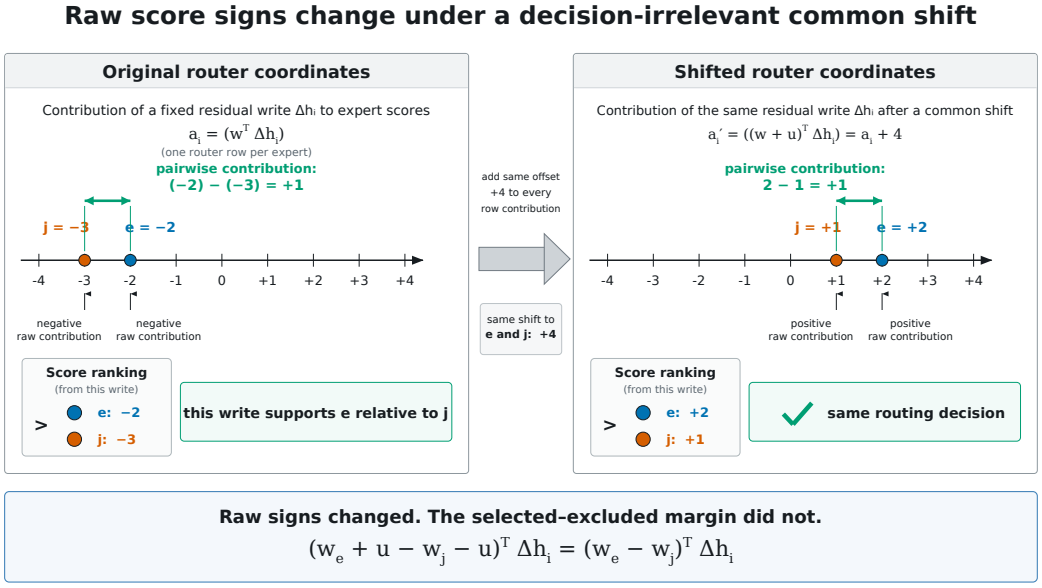


Figure 2: Raw-score signs are not intrinsic to the routing decision. A common shift u of every router row adds the same amount $u^\top \Delta h_i$ to a fixed write's contribution to every expert score, turning negative raw contributions positive, while the pairwise contribution $(w_e - w_j)^\top \Delta h_i$—and every top-$k$ decision—is unchanged. Margin coordinates remove exactly this non-identifiability.

graph uses the tightest inequality at the observed routing state,

$$e = \arg\min_{r \in S_\ell} \tilde{s}_{\ell,r}, \qquad j = \arg\max_{r \notin S_\ell} \tilde{s}_{\ell,r}. \tag{7}$$

Thus e is the weakest selected expert and j the strongest excluded competitor under the scores that actually determine selection. We call the selection margin at this tight pair the *boundary margin*. The resulting object is not the margin by itself, but its exact decomposition over upstream residual writes and its use as an accounting graph. Equation 6 is invariant to common router-row shifts because u cancels before the dot product. It is not invariant to every selection-preserving transformation: jointly rescaling all scores and biases by a positive constant preserves the selected set while rescaling every margin and edge.

3.3 The Routing-Decision Graph

We construct a graph with source nodes for the embedding and each upstream residual write, and a decision node for the boundary pair (e, j). The directed edge from write i has weight

$$a_{i \to (e,j)} = q_{\ell,e,j}^\top \Delta h_i. \tag{8}$$

Positive edges support e relative to j; negative edges support j relative to e. Their sum, together with the embedding term, reconstructs m^{ev} exactly, and adding $b_{\ell,e} - b_{\ell,j}$ reconstructs m^{sel}. The graph has no learned parameters and can be extracted at any checkpoint using the original forward computation.

Figure 3 shows how substantial positive and negative write contributions can nearly cancel at a tight boundary.

3.4 Accounting versus Causal Explanation

An edge in this graph states how a write contributes to a fixed linear decision margin. It does not establish that removing the write changes the route or the model output. A residual intervention can alter routing, expert inputs, and later attention simultaneously; a router intervention can alter both

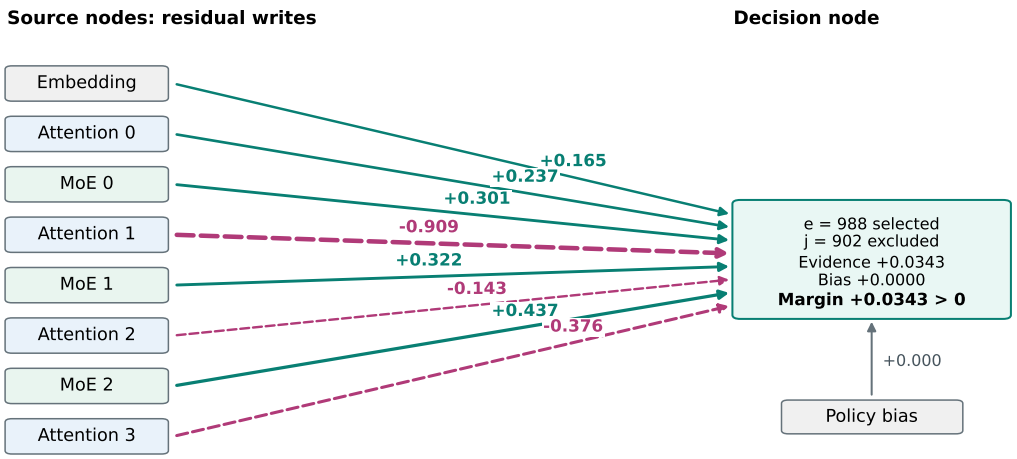


Figure 3: An observed routing-decision graph for one prespecified WikiText-103 test token (offset 4096, position 31) at MoE layer 3 of the six-layer $k = 64$ model. Each edge is the measured contribution of a preceding write to the boundary between selected expert 988 and excluded expert 902. Solid teal edges support the selected expert; dashed magenta edges support its competitor. The writes sum to an evidence margin of $+0.0343$; the policy-bias difference is zero, so the executed selection margin is also $+0.0343$. This is observed-forward accounting, not an estimate of what ablating any write would do.

expert identity and mixture weights. We therefore evaluate three targets separately: the pairwise boundary, the complete top-k set, and downstream cross-entropy. This distinction is central to the experiments that follow.

4 Experiments

Our experiments ask three questions. **(Q1)** Does the graph faithfully account for routing decisions on real text? **(Q2)** Do its largest edges causally control those decisions? **(Q3)** How far does routing accounting extend to downstream language-model behavior?

4.1 Experimental setup

We evaluate autoregressive language models with primary models ranging from 4.9M to 81M parameters, including 4-, 6-, and 16-layer models trained on character data, WikiText-103, and FineWeb-Edu. Their MoE blocks use fixed top-k routing and linear routers over the raw residual stream. We evaluate held-out contiguous windows. Subspace directions and dose calibrations use samples disjoint from the intervention evaluation; per-token component rankings use the clean forward pass of the evaluated token. Architecture controls replace the shared router with layer-specific routers, and additionally replace the shared expert bank with per-layer experts.

We distinguish fixed-pair boundary reversals from changes in the selected top-k set. Route Jaccard measures agreement of the complete selected sets; change in next-token cross-entropy (ΔCE, in nats) measures downstream behavior. The archived M2 readout is a top-k set-change statistic, not an independently verified fixed-pair reversal (Appendix C.1). Random controls are either equal-dimensional or scaled to match the intervention's induced change in router margins ("dose matched"). Further configuration, sampling, and uncertainty details are provided in Appendix A.

4.2 Faithfulness of the routing graph

Numerical reconstruction. Table 1 reports the largest absolute discrepancy between the boundary margin reconstructed from graph edges and a high-precision reference. Across the four principal

Table 1: Numerical reconstruction on held-out text. Margin error is the maximum absolute boundary-margin discrepancy.

Model	Layers	Parameters	Max margin error
Character LM	4	4.9M	3.34×10^{-5}
WikiText	6	46.0M	2.22×10^{-5}
WikiText	16	48.2M	4.20×10^{-4}
FineWeb-Edu	16	81.3M	1.13×10^{-4}

models it remains at or below 4.2×10^{-4}. Raw router scores reconstruct to approximately 10^{-6} relative to their scale. These are different error definitions: many boundary margins are close to zero, so a relative error normalized by the margin itself would be unstable. The decomposition is algebraically exact; the observed error arises from comparison with the model's finite-precision forward pass.

Why margins rather than score signs? We add a common vector to every router row. In exact arithmetic this preserves all pairwise margins and every top-k decision, but individual score signs and raw selected-score rankings change substantially. Margin edges are unchanged. At artificially large offsets, fp32 subtraction can change near-tied selections through cancellation; computing the row difference before the dot product avoids this failure. Thus margins remove a genuine non-identifiability of raw-score explanations while making their numerical implementation explicit.

4.3 Causal validation of routing decisions

Route-set sensitivity to supporting evidence. For each token and layer, we identify the largest positive edge supporting the selected expert and remove it with an antisymmetric boundary-score intervention (operator details in Appendix C.1). The archived intervention runs report values of roughly 70–100% across three language models; their Jaccard measurements are consistent with a one-member top-k set-change rate. The aggregate records do not identify the exchanged experts, so this is not a verified reversal of the original boundary pair or evidence that the source write is necessary. Removing a negative edge increases that pair's margin, yet the archived negative-edge set-change readout is nonzero. We therefore do not interpret the positive–negative contrast as a fixed-pair causal asymmetry.

A sparse boundary need not imply a sparse route. We next retain the largest N upstream write contributions and resample the remainder. For an individual selected–excluded boundary, retaining only the largest supporting term preserves approximately 57–82% of decisions. In contrast, the complete top-k set remains distributed: in deeper layers, retaining eight whole-write vectors yields route Jaccard no greater than about 0.41. Large values at the first one or two layers are less informative because few upstream writes are available there. These results distinguish sparse evidence for one local inequality from sufficiency for an entire route.

Selection and gates. Top-k changes alone do not imply a downstream effect. We therefore replay expert identities and gate weights in a 2×2 design. Changing identity while retaining baseline gates has approximately zero effect on cross-entropy; conditions that alter gate weights carry the measured router-level effect. In a separate negative-edge whole-write ablation, mean cross-entropy rises by 0.003–0.013 nats across models. This intervention also changes the inputs to experts and later layers, so its loss effect cannot be assigned to routing alone. Routing identity, gating, and residual content are distinct pathways.

4.4 From routing structure to model behavior

A load-bearing two-dimensional subspace. We estimate routing-related residual directions from signed margin contributions on a calibration split and ablate the leading two directions on held-out text (estimation and dose protocol in Appendix C). Table 2 shows a large loss increase in all three models at one layer per model; per-layer values appear in Appendix D. Equal-dimensional random subspaces have little effect.

Table 2: Held-out cross-entropy increase after ablating a two-dimensional routing-related residual subspace and an equal-dimensional random subspace (seed mean at the same layer; per-seed values span 0.0007–0.011 across these layers; per-layer and dose-matched results in Appendix D).

Model	Layer	Routing-related	Random
6-layer LM	2	2.06	0.006
16-layer WikiText	0	1.31	0.005
81M FineWeb-Edu	0	3.34	0.004

Two controls substantially narrow this result. First, bases estimated from negative, positive, and all routing evidence have subspace overlap 0.990–0.999 in the principal models (≥ 0.94 across architecture-control seeds, excluding one non-replicating seed-0 run disclosed in Appendix B) and produce nearly identical loss changes. The effect is therefore not specific to negative evidence. Second, the routing-derived ablation changes router margins more strongly than an unscaled random projection. Once random bases are scaled to match this dose, their loss effects can be comparable or larger, especially in the 81M model. The robust conclusion is that the estimated directions are low-dimensional and behaviorally load-bearing—not that their effect is sign-specific or established to be mediated by routing.

The large loss effect is not explained by route identity alone. Projecting the same directions only at the router flips 77–96% of tested boundaries. In the two 16-layer models the loss effect stays within ± 0.007; in the 6-layer model it is $+0.005$ at layer 0 but grows with depth, reaching $+0.22$ at the final layer, where the residual-stream ablation itself reaches $+0.81$ (at layer 2 the same contrast is $+0.02$ versus 2.06). By contrast, residual-stream projection changes every downstream consumer. The large ΔCE in Table 2 therefore establishes behavioral importance of the residual subspace, most of which is not mediated by route identity alone; the final-layer exception shows that router mediation can contribute in some configurations.

Accounting is not causal ranking. Finally, we rank upstream writes using signed margin, absolute margin, raw and centered score contributions, write norm, aligned gradients, and random baselines, then predict held-out donor-replacement outcomes. Signed margin is near chance; absolute margin is more useful but variable, and norm or gradient baselines often perform better. Raw selected-score rankings can change under a decision-irrelevant common router shift, whereas margin rankings cannot. Margin is therefore the appropriate invariant coordinate for accounting for a routing boundary, but not a universally superior measure of downstream causal importance. Donor replacement depends on the counterfactual change $(w_{\ell,e} - w_{\ell,j})^\top (\Delta h_i^{\mathrm{donor}} - \Delta h_i)$, rather than on the observed contribution $(w_{\ell,e} - w_{\ell,j})^\top \Delta h_i$ alone. This experiment therefore evaluates the off-the-shelf use of observed accounting edges as causal scores; it does not rule out intervention-aligned margin predictors.

4.5 Robustness and learned-surrogate baselines

The exact identity and the qualitative boundary/subspace results persist with layer-specific routers and with per-layer expert banks (Appendix B). In a matched pilot, sparse autoencoders obtain relative reconstruction MSE of 0.16–0.17 and route Jaccard of 0.41–0.51, while a write-aggregated cross-layer transcoder obtains MSE of 0.059–0.066 and route Jaccard of 0.60–0.72 (configurations and full tables in Appendix E). Native margins require no auxiliary fit and reconstruct the routing decision up to numerical precision. This comparison establishes the advantage for the narrowly defined routing-decision object; it does not claim that native graphs replace learned features or full transcoder graphs.

5 Discussion and limitations

The main result is a separation between three notions that are easily conflated. *Accounting* asks how upstream writes sum to the margin used by a router; for a linear router over the raw residual stream, our answer is exact. *Routing causality* asks which writes or score changes alter the selected experts or gates; this requires intervention. *Behavioral causality* asks which changes affect language-model

predictions; it additionally depends on expert content, later attention, and other residual-stream pathways. Our interventions show that success at the first level does not guarantee success at the other two.

This distinction also clarifies what is obtained "for free." No post-hoc model is trained to construct the upstream routing edges, so the same analysis is available at every training checkpoint without accumulating surrogate reconstruction error. Graph extraction, storage, interpretation, and causal validation still incur cost. Sparse autoencoders and transcoders also address broader questions, including semantic feature discovery, that the routing margin does not answer.

Our study has several limitations. The evaluated models are modest in scale and were designed to expose routing cleanly; larger production MoEs may introduce nonlinear routers, normalization, capacity constraints, and distributed implementation details. All checkpoints are also early in training (3000–9000 steps); the token-level language models have held-out perplexities of 33–112, while the character model is evaluated in bits per character. Our longitudinal study shows that routing geometry can reorganize sharply over training (Appendix **??**); the geometry characterized here may differ in converged models. The central language-model interventions use a small number of base-model seeds. The low-dimensional residual ablation is not fully mediated by routing and should not be interpreted as a routing-only mechanism. The graph leaves attention internals and expert semantics opaque, and its edge weights need not be sparse or human-interpretable. Finally, the surrogate comparison is a routing-focused pilot rather than a benchmark of complete circuit recovery.

Despite these limits, the broader design principle is useful: interpretability can partly be an architectural property. When a model exposes a decision-relevant quantity as a sum of identifiable internal writes, exact accounting can replace a learned approximation for that quantity. The resulting graph is best viewed as a constrained interface for causal investigation, not as a finished explanation.

6 Conclusion

We introduced architecture-native routing-decision graphs for linear routers over additive residual streams. Pairwise selected–excluded margins decompose exactly over upstream writes and remain invariant to common router-score shifts, allowing routing evidence to be inspected without training an explainer. Interventions reveal sparse supporting evidence for some individual boundaries and behaviorally load-bearing two-dimensional residual directions estimated from routing contributions, while also showing that sign specificity, route mediation, and downstream causal importance do not follow from exact decomposition alone. Architecture can therefore make a useful subset of model computation directly auditable, with causal testing—rather than algebra alone—determining what that audit explains.

Reproducibility statement

Appendix material specifies model configurations, datasets, evaluation windows, intervention operators, controls, and additional results. We will release the training, routing-attribution, and intervention code.

AI use statement

The authors used generative AI to provide feedback on drafts of the paper, assist with the additional experiments reported in the appendix, improve the manuscript, and assist with programming. The authors reviewed and revised the AI-assisted text and code, verified the code used in the reported experiments, and take responsibility for the final content of this work.

References

Trenton Bricken, Adly Templeton, Joshua Batson, Brian Chen, Adam Jermyn, Tom Conerly, Nick Turner, Cem Anil, Carson Denison, Amanda Askell, Robert Lasenby, Yifan Wu, Shauna Kravec, Nicholas Schiefer, Tim Maxwell, Nicholas Joseph, Zac Hatfield-Dodds, Alex Tamkin, Karina Nguyen, Brayden McLean, Josiah E. Burke, Tristan Hume, Shan Carter, Tom Henighan, and Christopher Olah. Towards monosemanticity: Decomposing language models with dictionary

learning. Transformer Circuits Thread, 2023. URL https://transformer-circuits.pub/2023/monosemantic-features.

Yilong Chen, Naibin Gu, Junyuan Shang, Zhenyu Zhang, Yuchen Feng, Jiawei Sheng, Tingwen Liu, Shuohuan Wang, Yu Sun, Hua Wu, and Haifeng Wang. Mixture of universal experts: Scaling virtual width via depth-width transformation, 2026.

Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine Learning*, 20(3):273–297, 1995. doi: 10.1007/BF00994018.

Róbert Csordás, Kazuki Irie, Jürgen Schmidhuber, Christopher Potts, and Christopher Manning. MoEUT: Mixture-of-experts universal transformers. In *Advances in Neural Information Processing Systems 37*, pp. 28589–28614, 2024. doi: 10.52202/079017-0897.

Mostafa Dehghani, Stephan Gouws, Oriol Vinyals, Jakob Uszkoreit, and Lukasz Kaiser. Universal transformers. In *International Conference on Learning Representations*, 2019.

Leonard Engmann, Christian Medeiros Adriano, and Holger Giese. From observation to intervention: A causal audit of expert importance in mixture-of-experts models, 2026.

William Fedus, Barret Zoph, and Noam Shazeer. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity, 2021.

Martin Jaggi. Tying the loop – tied expert layers in mixture-of-experts language models, 2026.

Wengang Li, Lingqi Zhang, Toshio Endo, and Mohamed Wahib. Understanding cross-layer contributions to mixture-of-experts routing in llms. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/3efdd3cd3d70da09ec6ed3b3d17d2fd5-Abstract-Conference.html.

Jack Lindsey, Wes Gurnee, Emmanuel Ameisen, Brian Chen, Adam Pearce, Nicholas L. Turner, Craig Citro, David Abrahams, Shan Carter, Basil Hosmer, Jonathan Marcus, Michael Sklar, Adly Templeton, Trenton Bricken, Callum McDougall, Hoagy Cunningham, Tom Henighan, Adam Jermyn, Andy Jones, Andrew Persic, Zhenyi Qi, T. Ben Thompson, Sam Zimmerman, Kelley Rivoire, Thomas Conerly, Christopher Olah, and Joshua Batson. On the biology of a large language model. Transformer Circuits Thread, 2025. URL https://transformer-circuits.pub/2025/attribution-graphs/biology.html.

Nam V. Nguyen, Thong T. Doan, Luong Tran, Van Nguyen, and Quang Pham. LibMoE: A library for comprehensive research on mixture of experts in large language models. *Transactions on Machine Learning Research*, 2026. URL https://openreview.net/forum?id=PB2ju8tq0n.

Eric S. Qiu, Utku Umur Acikalin, Justin Lovelace, Christian Belardi, Arjun B. Mulchandani, Carla P. Gomes, and Kilian Q. Weinberger. MoRE: Mixture of reused experts. In *Conference on Language Modeling*, 2026.

Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer, 2017.

Wei Shi, Sihang Li, Tao Liang, Mingyang Wan, Guojun Ma, Xiang Wang, and Xiangnan He. Route sparse autoencoder to interpret large language models, 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. URL https://openreview.net/forum?id=NpsVSN6o4ul.

Liu Yang, Kangwook Lee, Robert Nowak, and Dimitris Papailiopoulos. Looped transformers are better at learning learning algorithms. In *International Conference on Learning Representations*, 2024.

Charles Ye, Bo Yuan, and Lee Sharkey. Polysemantic experts, monosemantic paths: Routing as control in moes, 2026.

Youngsik Yoon, Siwei Wang, Wei Chen, and Jungseul Ok. When are experts misrouted? counterfactual routing analysis in mixture-of-experts language models, 2026.

Andy Zhu, Rongzhe Wei, Yupu Gu, and Pan Li. GRIP: Algorithm-agnostic machine unlearning for mixture-of-experts via geometric router constraints, 2026.

A Experimental details

A.1 Models and data

Table 3 lists the primary models. All use fixed top-k routing over the raw residual stream and layer-specific attention. The WikiText and FineWeb-Edu models share a bank of 1024 experts and a router across depth, executing $k = 64$ experts per token. They use 8 attention heads, $d_{\text{model}} = 384$ and 512 respectively, and bias-based load balancing (bias update rate 10^{-3}, router z-loss coefficient 10^{-3}). Architecture controls vary router and bank sharing (Appendix B); further seeds, a $k = 16$ variant, and a 32-layer model appear in Appendix D.

Table 3: Evaluated models. All except the character model use a 50257-token vocabulary; the character model is character-level. Training steps refer to the evaluated checkpoint.

Model	Layers	Parameters	Training data	Steps	Held-out level
Character LM	4	4.9M	text8	5000	1.84 bpc
WikiText	6	46.0M	WikiText-103	5000	33.1 ppl
WikiText	16	48.2M	WikiText-103	3000	111.6 ppl
FineWeb-Edu	16	81.3M	FineWeb-Edu	9000	66.0 ppl

The character model uses seed 0. WikiText models use the native WikiText-103 splits; FineWeb-Edu uses a contiguous 98/1/1 split of a 600M-token cache.

A.2 Evaluation protocol

All evaluations run single-seed forward analyses on held-out contiguous text windows disjoint from training data (evaluation token offsets $2\text{M}/3\text{M}/4\text{M}/5\text{M}$; sequence length 256; batch size 4 for intervention studies and 8 for baseline evaluations). For the subspace studies, discovery, dose calibration, and test windows occupy disjoint offsets (100k–400k, 20k–40k, and 2M–3.5M tokens respectively). Per-token component rankings are computed from each evaluated token's clean forward pass, then tested by intervention on that token; they are not fitted on a disjoint sample. Interventions are applied in parallel over all positions of a batch, not one token at a time. Unless noted otherwise, uncertainty is reported at window level; bootstrap confidence intervals use 16 evaluation windows (95% intervals), and the surrogate study of Appendix E uses 200 block-bootstrap resamples over disjoint sequence blocks. All runs were executed on a single NVIDIA L20Y (torch 2.7.1, cu126).

A.3 Outcome measures

Fixed-pair boundary flip rate. Where measured directly, the fraction of evaluated (token, layer) positions at which an intervention reverses the baseline selected–excluded ordering of the pair in Equation 7. The archived M2 field named `boundary_flip` is treated separately in Appendix C.1.

Route Jaccard. The intersection-over-union between the complete top-k expert sets of the intervened and baseline forward passes.

Change in cross-entropy (ΔCE). The change in mean next-token cross-entropy, in nats, relative to the baseline forward pass on the same windows.

Subspace overlap. The overlap between two estimated bases, computed from their principal angles.

Dose. The magnitude of an intervention's induced change in router margins, measured on the calibration split. The routing-derived ablation at unit strength ($\alpha = 1$) defines a target dose; dose-matched random controls are rescaled by a fitted factor α^* (capped at 50) so that their induced margin change matches this target. Unscaled equal-dimensional random controls apply the same projection dimension at $\alpha = 1$ without rescaling.

B ARCHITECTURE CONTROLS

Tables 4 and 5 compare three WikiText architectures with $k = 64$. Configuration A, the primary testbed, shares both the router and expert bank across layers. Configuration B retains the shared expert bank but uses a separate router at each layer. Configuration C also uses separate expert banks at each layer, raising the parameter count to 173.8M; it is a qualitative control, not a parameter-matched comparison. In all three cases, the router reads the additive residual stream linearly, which is sufficient for the exact margin decomposition. Subscripts on A denote seeds.

Table 4: Architecture controls: routing measurements. Margin error is the maximum absolute boundary-margin discrepancy. Positive/negative columns give the archived top-k set-change readout at $\alpha = 1$, not verified fixed-pair flips (Section C.1). For C, readouts are from layer 0 (pooled values: 0.901/0.072).

Config.	Margin err.	Positive	Negative
A_0	2.2×10^{-5}	0.964	0.033
A_1	$\sim 2.5 \times 10^{-5}$	0.939	0.058
A_2	$\sim 2.5 \times 10^{-5}$	0.936	0.054
B	2.9×10^{-5}	0.907	0.088
C	3.0×10^{-5}	0.741	0.197

Table 5: Architecture controls: two-dimensional subspace ablation at $\alpha = 1$ (layer 2 for 6-layer models). Overlap compares negative- and positive-evidence subspaces; ΔCE is held-out loss increase.

Config.	Overlap	ΔCE	Test ppl
A_0	0.995–0.997	2.06	33.1
A_1	0.942	1.44	33.0
A_2	0.993	1.59	33.0
B	0.988–0.992	4.00	38.0
C	0.965–0.976	3.5–4.3	37.0

One seed-0 run of configuration C showed a depressed layer-2 overlap (0.66); this did not replicate in two further seeds (0.965, 0.976; layer 5 ≥ 0.989 in all seeds) and we treat it as single-seed noise. Exact reconstruction (Section 4.2), supporting-edge route-set sensitivity (Section 4.3), and the load-bearing low-dimensional subspace (Section 4.4) all persist in every configuration.

C INTERVENTION OPERATORS AND CONTROLS

C.1 ANTISYMMETRIC BOUNDARY-SCORE INTERVENTION

For a boundary pair (e, j) and an upstream write i with edge weight $a_i = q_{\ell,e,j}^{\top} \Delta h_i$, the logit-level removal of that edge subtracts $a_i/2$ from the executed score of e and adds $a_i/2$ to the executed score of j, so the executed selection margin changes by exactly $-a_i$—the edge's contribution is cancelled without modifying the residual stream. A dose parameter $\alpha \in \{0.25, 0.5, 1, 2, 4\}$ scales this adjustment (margin change $-\alpha\, a_i$), so $\alpha = 1$ removes the edge's contribution and $\alpha = 2$ reverses its sign. Conditions remove the largest positive (supporting) edge, the largest negative (opposing) edge, a random edge, the largest absolute edge, or reverse the sign of the supporting edge; a norm-matched variant pins the intervention norm while keeping its direction. The recorded mean dose scales linearly with α and has the expected sign in both edge conditions. For $k = 64$, a single-member exchange has route-set Jaccard $63/65$; across the archived M2 records, $\overline{J} \simeq 1 - 2f/65$ for the

reported $f = \texttt{boundary_flip}$ (maximum discrepancy 1.1×10^{-7} across the 1088 records with both fields). This is consistent with f measuring the fraction of positions with a one-member top-k set change, but the aggregate records do not identify which experts were exchanged. For $a_i < 0$, the declared operator increases the original pair's margin and cannot reverse that pair; the nonzero negative-edge readout therefore cannot be a fixed-pair flip. Without the runner source, we do not assign the positive-edge set changes to verified (e, j) exchanges or infer sign asymmetry. Evaluated layers are all layers of the 6-layer model and $\{0, 4, 8, 12, 15\}$ of the 16-layer models. This score-level operator tests route-set sensitivity, not the necessity of a named residual write.

C.2 Retain-top-N resampling

For each evaluated position, we retain the N largest upstream write contributions ($N \in \{0, 1, 2, 4, 8\}$, under positive-, negative-, absolute-margin, gradient, and random rankings) and replace all remaining writes with donor writes taken from the same position in a sequence shifted by one batch index. Part A measures pairwise-boundary outcomes (sufficiency: the baseline boundary is preserved; necessity: survival after dropping the top edge). Part B measures full top-k route Jaccard and ΔCE.

C.3 Selection-and-gate replay (2×2 design)

To separate expert identity from gating, we replay the baseline selected identities S_0 and/or baseline gate weights G_0 (Equations 2 and 3) in an otherwise perturbed forward pass, yielding four conditions: baseline S_0G_0, gate-only S_0G_1, identity-only S_1G_0, and full router change S_1G_1. Cross-entropy differences between these conditions attribute the router-level effect to identity or gating pathways.

C.4 Routing-related subspace estimation and dose matching

On the discovery split, we collect at each (token, layer) position the upstream write vectors Δh_i of Equation 1, weight each write by the magnitude of its margin contribution, $|a_i| = |q_{\ell,e,j}^\top \Delta h_i|$ for the position's boundary pair, and estimate the routing-related subspace as the top two directions of the weighted centered covariance (weighted PCA). The negative-, positive-, and total-evidence bases restrict this collection to writes with negative, positive, or all contributions, respectively. Ablating the subspace projects the residual stream to remove these components at every position of the test split, with strength $\alpha \in \{0.5, 1, 2\}$. Random controls draw orthonormal 2D bases (3 seeds), either applied at $\alpha = 1$ (unscaled) or rescaled by α^* fitted on the calibration split to match the routing-derived ablation's induced margin change (dose-matched). Bases estimated from negative-only, positive-only, and all signed evidence are compared by subspace overlap and by their induced loss changes.

C.5 Router-only projection

To test route mediation, the same projection is applied only to the router's input at the target layer, leaving the residual stream itself unchanged for all other consumers. This flips most tested boundaries while isolating the routing pathway from content pathways.

C.6 Donor replacement and ranking validity

For each target write, donor replacement substitutes the whole write vector with the write at the same position of a sequence shifted by one batch index, and measures the realized boundary outcome in a full forward pass. We compare rankings of upstream writes by observed signed margin, observed absolute margin, raw selected-score contribution, centered score contribution, write norm, cross-entropy-aligned gradients, and a random baseline, scoring each ranking by AUROC against held-out boundary-flip labels and by Spearman correlation with realized $|\Delta\text{CE}|$. A row-shift control recomputes the rankings after adding a common vector to every router row. Because the outcome depends on the counterfactual change $q_{\ell,e,j}^\top(\Delta h_i^{\text{donor}} - \Delta h_i)$ rather than on the observed contribution $q_{\ell,e,j}^\top \Delta h_i$ alone, this experiment tests the off-the-shelf use of observed accounting edges as causal scores; it does not rule out intervention-aligned margin predictors.

D FULL NUMERICAL RESULTS

Numerical reconstruction. Table 6 extends the main-text reconstruction check to all evaluated checkpoints. Raw-score errors are reported relative to the raw logit scale; margin errors are absolute (many boundary margins are near zero, so margin-relative errors are unstable). Under an artificially large common offset ($\approx 100\times$ the largest logit magnitude), computing row differences before the dot product keeps errors at 1.6×10^{-5}–5.5×10^{-4}, while a naive fp32 implementation degrades to 1.9×10^{-3}–1.2×10^{-1} with top-k set agreement falling to 0.34–0.89; this motivates the implementation note in the main text. Row-shift stress tests change the top-k set at 0.1–0.9% of positions and margin edges by at most 1.2×10^{-3}. Routing the router input through layer normalization instead of reading the raw stream breaks the identity, as expected (scaled raw-score error grows from 1.05 at layer 0 to 12.84 at layer 5).

Table 6: Numerical reconstruction on held-out text, all checkpoints. Margin errors are maximum absolute discrepancies against a high-precision reference; p99 in parentheses. Raw max is the maximum raw-score error relative to the raw logit scale.

Checkpoint	Layers	Max margin error (p99)	Raw max
Character LM	4	3.34×10^{-5} (1.76×10^{-5})	1.15×10^{-6}
WikiText, $k = 16$	6	3.78×10^{-5}	2.19×10^{-6}
WikiText, $k = 64$ (s0/s1/s2)	6	$(2.22, 2.15, 2.81) \times 10^{-5}$	$(1.68, 1.61, 2.01) \times 10^{-6}$
WikiText	16	4.20×10^{-4} (2.10×10^{-4})	1.31×10^{-6}
WikiText	32	5.72×10^{-4} (2.86×10^{-4})	1.47×10^{-6}
FineWeb-Edu	16	1.13×10^{-4} (5.05×10^{-5})	1.85×10^{-6}

Controlled shift example. The failure case behind Contribution 2 is a controlled study in a small model (4 layers, width 128, 256 experts, $k = 8$, bias balancing, router z-loss 10^{-3}) trained on an induction task, evaluated on one held-out batch of 32 sequences. We subtract the across-expert mean row from every router row while keeping balancing biases fixed, which adds a common term to every score; expert selections, gate weights, and model outputs change by less than 10^{-3}. In the raw basis, 69.6% of score-contribution edges are negative and negative edges carry 85.9% of contribution mass; after this centering, the two statistics drop to 32.3% and 8.5%, and the fraction of positions whose largest edge is negative falls from 99.6% to 12.3%. The apparent negative-evidence mass is therefore dominated by a removable common component (a shared negative baseline, for which the router z-loss is a plausible but unproven source), and raw-score sign statistics are not intrinsic to the routing decision.

Boundary interventions. Table 7 gives the archived set-change readout at $\alpha = 1$; the supporting-edge values range from 0.687 to 1.000 across layers and windows. Routing-only logit overrides have small mean cross-entropy effects ($\leq 2 \times 10^{-3}$ for $\alpha \leq 2$). A separate negative-edge whole-write ablation increases mean cross-entropy by 0.003–0.013 nats across the three models. In the 2×2 replay at $\alpha = 4$, identity-only effects are negligible (-3×10^{-6} to $+1.3 \times 10^{-4}$ nats); conditions changing gates carry the router-level effect.

Table 7: Archived `boundary_flip` readout under the reported antisymmetric boundary-score intervention at $\alpha = 1$ (mean over layers and windows). It is consistent with a one-member top-k set-change rate, not a verified fixed-pair flip rate; see Section C.1.

Model	Supporting edge	Opposing edge	Random edge
WikiText 6-layer	0.964	0.033	0.491
WikiText 16-layer	0.936	0.035	0.465
FineWeb-Edu 16-layer	0.958	0.028	0.470
Range across layers and windows: supporting $[0.687, 1.000]$, opposing $[0.000, 0.199]$.			

Sparse boundary versus sparse route. Retaining only the largest supporting contribution per position preserves 57–82% of baseline boundaries (rising with depth; the opposing-edge variant falls to 20–43%, below the resample-everything floor). Dropping the largest supporting edge leaves the boundary intact in only 17% of positions on average (6-layer; range 0.11–0.33 across layers and windows), versus 84% when dropping the largest opposing edge. For the full route, retaining eight whole writes yields route Jaccard past layer 0 of at most 0.61 (6-layer, at layer 1; later layers ≤ 0.40), 0.36 (16-layer WikiText; deep layers ≤ 0.19), and 0.41 (FineWeb-Edu; deep layers ≤ 0.25); layer 0 is trivially high because only the embedding and first attention write are available there.

Subspace ablation. Table 8 gives the full subspace results (estimation protocol in Appendix C.4). Across all evaluated layers and seeds, unscaled random controls span $\Delta\text{CE} = -0.007$–0.023, with per-layer seed means spanning 0.004–0.017; at the headline layers of Table 2, per-seed values span 0.0007–0.011 with seed means of 0.004–0.006. Estimated subspaces are genuinely low-dimensional: the participation ratio of the weighted contribution distribution (the effective direction count, $(\sum_i \lambda_i)^2 / \sum_i \lambda_i^2$ over PCA eigenvalues) is 1.4–5.3 in mid and deep layers, and the top two directions capture 73–77% of weighted variance at the 6-layer model's layer 2 (35–40% at the 16-layer models' layer 0). Dose-matched random controls reach ΔCE comparable to or larger than the routing-derived ablation in the 16-layer models (FineWeb-Edu layer 0: random seed mean 3.60, seed range 3.19–3.84, versus routing-derived 3.34) but remain below it in the 6-layer model ($\alpha^* = 4.7$–14.4, ΔCE 0.53–1.21 versus 2.06). Router-only projection flips 77–96% of boundaries; its loss effect is within ± 0.007 in the 16-layer models but grows with depth in the 6-layer model ($+0.005$, $+0.019$, $+0.222$ at layers 0, 2, 5), as discussed in the main text.

Table 8: Two-dimensional routing-subspace ablation at $\alpha = 1$ (negative basis; positive and total bases are within 0.02 nats at the 6-layer model and within 0.08 nats at the 16-layer models). Overlap is negative- versus positive-evidence subspace overlap. Random is the unscaled equal-dimensional control at the same layer (seed mean). Dose-matched random uses the calibration-fitted α^*.

Model	Layer	ΔCE	Random	Dose-matched random	Overlap
WikiText 6-layer	2	2.06	0.006	0.53–1.21	0.995–0.997
WikiText 16-layer	0	1.31	0.005	0.91–3.45	0.990–0.999
FineWeb-Edu 16-layer	0	3.34	0.004	0.99–4.03	0.992–0.999

Across all evaluated layers, the routing-derived ΔCE at $\alpha = 1$ spans 1.96 / 2.06 / 0.81 (6-layer, layers 0/2/5), 0.82–1.37 (16-layer WikiText, layers 0–15), and 0.10–3.34 (FineWeb-Edu, layers 0–15), with no single consistent depth profile across models.

Ranking validity. Table 9 reports ranking quality against held-out donor-replacement outcomes (protocol in Appendix C.6). Observed signed margin is at chance in every model; absolute margin is somewhat better but inconsistent; write norm dominates at layer 0 of every model and overall in the 16-layer WikiText model. Raw selected-score rankings are actively misleading for loss effects (Spearman versus realized $|\Delta\text{CE}|$ down to -0.76) and unstable under decision-irrelevant row shifts (mean before/after rank correlation 0.09, 0.28, and -0.02 across the three models, with per-window values spanning -0.78 to 0.94), whereas signed-margin rankings are exactly invariant (correlation 1.0).

Table 9: AUROC of upstream-write rankings against held-out boundary-flip labels under donor replacement (mean over layers and windows).

Ranking	WikiText 6L	WikiText 16L	FineWeb-Edu
Signed margin	≈ 0.50	≈ 0.50	≈ 0.50
Absolute margin	0.540	0.630	0.549
Centered score	0.559	0.540	0.516
Gradient	0.539	0.592	0.594
Write norm	0.781	0.996	0.500
Random	≈ 0.50	≈ 0.50	≈ 0.50

E LEARNED-SURROGATE PILOTS

Native accounting uses observed residual writes and router weights without an auxiliary fit, while sparse autoencoders and cross-layer transcoders fit learned replacements that can address broader feature-level questions.

We fit sparse autoencoder (SAE) and write-aggregated cross-layer transcoder (CLT) pilots at layer 3 of the 6-layer WikiText model, using checkpoints at training steps 1000, 3000, and 5000. Both surrogates use width 768 with 32 active features, 1000 fitting steps (learning rate 3×10^{-4}, batch 256 over 32 training windows), and are evaluated on 8 held-out windows with 200 block-bootstrap resamples over disjoint sequence blocks (Table 10). The intervention operator is target-layer router-only whole-write donor replacement; the estimand is the selection margin at the boundary pair (weakest selected minus strongest excluded, including the balancing-bias term). Native accounting reconstructs this margin with mean absolute error 2.9×10^{-7}.

Table 10: Surrogate pilots. Retrain fits and evaluates at the same training stage; transfer fits at step 1000 and evaluates at later stages. Ranges span training stages and auxiliary seeds. Route Jaccard is agreement of the induced top-k set with the native model; flip AUROC scores surrogate rankings against native route-flip labels. Native margin error is reported separately in the text as an absolute error, not in the relative-MSE column.

Surrogate	Relative MSE	Route Jaccard	Flip AUROC
SAE (retrain)	0.16–0.17	0.41–0.51	0.939–1.000
SAE (transfer)	0.33–0.34	0.34–0.35	—
CLT (retrain)	0.059–0.066	0.60–0.72	0.936–0.999
CLT (transfer)	0.09–0.11	0.50–0.52	—
Native margin edges	—	exact	0.947–1.000

F INDUCTION CASE STUDY

We train a small model to 100% accuracy on a controlled induction task. Occluding each component independently identifies two attention components as necessary: removing them drops induction accuracy by 0.981 and 0.975, while every other candidate (embedding, four expert writes, two further attention writes) shows no measurable drop. Ranking the eight margin contributors at the target routing decision places exactly these two components first and second, and removing them disrupts 97.8–99.3% of evaluated route sets. We report this as component prioritization, not recovery of a complete circuit.