Automated Attribution Graph Interpretation via Probe Prompting

Anonymous Authors[1]

Abstract

Even though we know the precise computations that lead from a large language model (LLM) input to its output, this computation is too complicated to interpret. We can try to make it simpler by creating a sparse computational graph that captures most of the model behavior with the smallest number of computational nodes. Cross-layer transcoders (CLT) decompose the dense computations of the MLP but the resulting circuits still contain thousands of nodes even for short prompts. Existing automated interpretation methods label individual features from corpus activations, which are often not validated by causal intervention. We introduce *probe prompting*, a transparent rule-based pipeline that groups the features of an attribution graph into concept-aligned supernodes from their responses on a small set of concept-targeted probe prompts, summarized as Cross-Prompt Activation Signatures (CPAS). Across four factual domains, on Gemma-2-2B with a public CLT dictionary and 44,596 entity-swap interventions, we find that the labeled supernodes beat random and influence matched baselines. Code, datasets, and an interactive demo are released anonymously as a reusable harness for calibrating supernode labels against causal interventions.

1. Introduction

A part of mechanistic interpretability research in large language models (LLMs) is focused in decomposing the representations used by these models and on simplifying the computational graph in such a way as a to make it human understandable, so called reverse engineering. One way researchers have done this decomposition is by training sparse autoencoders (SAEs) and their variants (Bricken et al., 2023; Templeton et al., 2024; Transformer Circuits Team, 2024; Marks et al., 2025). SAEs are trained with the objective of reconstructing model activations, be them the residual stream, the output of MLP or even of attention heads, while using a sparsely activating basis (features) which is hopefully more interpretable.

Automated interpretation of individual features by LLM scoring (Bills et al., 2023; Paulo et al., 2024; Templeton et al., 2024) produces per-feature labels from corpus activations but does not yield circuit-level interaction. Attribution graphs (Ameisen et al., 2025; Lindsey et al., 2025a) trace the influence of features through the computational graph, but most graphs still have too many nodes to manually interpret. A single Gemma-2-2B graph for a short factual prompt routinely contains 600–5,000 cross-layer transcoder (CLT) features and edges between them (Ameisen et al., 2025; Lindsey et al., 2025a), and manual reading by an experienced circuit tracer has been reported to take on the order of two hours per prompt (Lindsey et al., 2025b). As mechanistic interpretability moves from individual case studies to large-scale catalogs of circuits (Hanna & Ameisen, 2026; Marks et al., 2025), automated circuit-level grouping tied to behavior is required.

We propose *probe prompting*, a transparent rule-based pipeline that converts an attribution graph into a compact set of concept-aligned supernodes. For each candidate feature in the graph we run a small set of concept-targeted probe prompts and summarize the feature's responses as a Cross-Prompt Activation Signature (CPAS). Deterministic threshold rules map each signature to one of four functional roles, and features that share role and name are merged into a supernode. Every assignment is traceable to a specific threshold crossing, so a researcher can audit and edit the grouping. We can then produce controlled intervention experiments that can evaluate how good the supernode labels are.

The paper contributes:

- A transparent rule-based pipeline that groups the features of an attribution graph into concept-aligned supernodes from Cross-Prompt Activation Signatures (Section 3).
- A fitness check for supernode labels, with a matched-random-control swap protocol that is grouping-method-

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

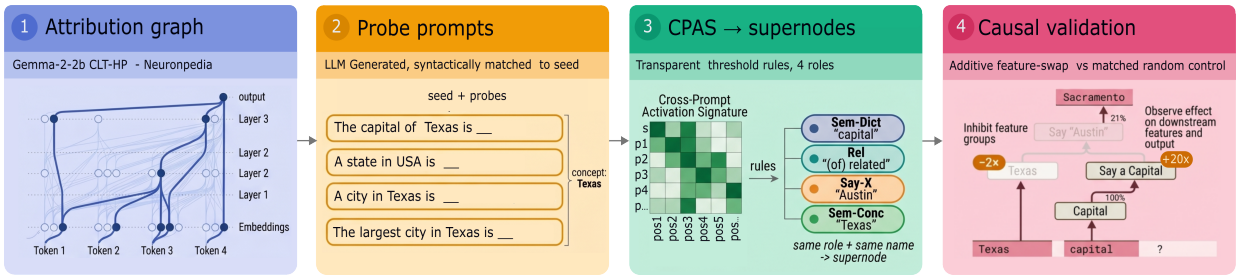

Figure 1. Pipeline overview. Four stages: (1) attribution graph generation (through Neuronpedia); (2) probe prompts (LLM-generated, syntactically matched to seed); (3) Cross-Prompt Activation Signatures followed by transparent rule-based supernode typing; (4) causal validation via additive feature-swap interventions with structurally matched random controls (same feature count, same per-layer histogram, sampled outside concept-matching supernodes). The unit of analysis is the supernode, validated operationally by the Specificity and Matched-Control conditions of §4.1.

agnostic and applicable to any future grouping, including learned ones (Sections 3), as well as a influence matched protocol that swaps the most influential features.

- Evidence that the proposed protocol labeled supernodes pass this test in four different factual-recall domains, showing the generalizability of the technique (Section 4).
- A open release of a 44,596-run intervention dataset, anonymous code, and an interactive demo (Appendix A).

2. Related Work

Attribution graphs and replacement models. Attribution graphs operationalize feature→logit pathways in a local replacement model that linearizes the residual stream through CLT or SAE features and freezes attention (Ameisen et al., 2025; Lindsey et al., 2025a). https://www.neuronpedia.org/ exposes these graphs for selected models and displays per-feature cards at scale (Neuronpedia, 2025; Lindsey et al., 2025c). `Circuit-tracer` provides the open-source reference implementation (Safety Research contributors, 2025) for computing these graphs. Our work operates entirely *downstream* of attribution: we do not propose a new attribution method, but we do produce supernodes that Neuronpedia tooling can render and pin. Our work proposes a clustering technique that reduces the human labor required to produce supernode labels.

Causal circuits with transcoders. Hanna & Ameisen (2026) use cross-layer transcoder feature circuits to study whether language models latently plan rhyming words, structuring their analysis around two falsifiability conditions (predict-token-before-emission, causally-affect-token). Marks et al. (2025) apply sparse-feature circuit analysis to subject–verb agreement; we contribute a domain-general harness and a proposal on how to generate groupings of attribution graphs on scale.

Causal interpretability and matched random controls. Geiger et al. (2025) formalize three levels of interpretive claim and emphasize that *distinguishing* levels requires intervention. Heap et al. (2025) show that SAE auto-interpretability scores (Bills et al., 2023; Paulo et al., 2024) and standard SAE reconstruction metrics can be similar for randomly-initialized and trained transformers, highlighting the need for metrics anchored to causal effect. Our matched-random-control protocol (§3.8) is designed to address this critique: any structural property a random feature set shares with the labeled one (count, layer histogram, eligibility) is held constant; only the concept alignment changes. We also compare our intervention technique with substituting the most influential features, showing that our grouping is working not only by increasing the number of features steered.

3. Method

3.1. Domains and seed prompts

We set four domains spanning two-hop factual recall in different relational structures and answer templates. Each domain has a fixed seed-prompt with three semantic fields (input, intermediate, answer):

- USA: "capital of state containing X"; 50 entities, 2,450 non-identity swap pairs.
- Books: "character X appears in a book by"; 10 entities, 90 non-identity swap pairs.
- Products: "founder of the company that makes X"; 12 entities, 132 non-identity swap pairs.
- Paintings: "first name of the painter of X"; 10 entities, 90 non-identity swap pairs.

3.2. Setting and attribution-graph generation

We use Gemma-2-2B with the public CLT dictionary ($\sim$2.5M features over 26 layers; Hanna & Piotrowski, 2025; Ameisen et al., 2025). Attribution graphs are generated via the Neuronpedia API; exact API parameter names and values are in Section B. The replacement model freezes attention during graph computation.

3.3. Probe prompt generation

Given a prompt used to generate the attribution graph p (e.g., "the capital of the state containing Dallas is"), we use an LLM to generate a small set of concept-targeted probes. First, we extract the K most salient concepts from p (and, when available, from the model's output), each tagged with a short description. The second pass turns the extracted concepts into probe prompts that share the seed's surface structure (same prepositions, same answer position, comparable length) while varying the content entities. A typical seed yields 4–5 probes. The full system prompt and the concept-to-probe template are reproduced in Section B.

3.4. Cross-Prompt Activation Signatures (CPAS)

A peak on a single prompt is ambiguous: a feature that activates on `is` in the seed could be a dictionary detector for the copula, a context-specific feature, or a procedural feature that promotes whatever output the model is about to emit. The CPAS resolves the ambiguity by aggregating peaks across deliberately varied probes (Figure 2).

For each (feature, probe prompt) pair we record a small set of per-probe measures: at each token position we measure the feature's peak activation, how sparsely the feature fires across the probe set, and a robust z-score and cosine similarity of the feature's activations relative to the seed prompt. Aggregating across probes yields a compact per-feature signature (the CPAS): a handful of summary numbers capturing how consistent the peak token is across probes, how many distinct tokens the feature peaks on, and the feature's confidence in its dominant role. Formal definitions and raw-measure specifications are in Section B.

3.5. Functional vs. semantic tokens; target-token mapping

Tokens are labeled *functional* (eg. copulas, articles, prepositions) or *semantic* (content-bearing). When a feature peaks on a functional token, a ± 7-token directional search identifies the nearest semantic peak (e.g., forward for articles; backward for the possessive `'s`).

3.6. Four functional roles

Each feature is assigned one of four functional roles. The four-role vocabulary is an empirical extension of the supernode types that appear in circuit-tracing case studies (Lindsey et al., 2025a; Ameisen et al., 2025); we found that these four roles covered the vast majority of features we encountered during pilot analyses, and we formalize them here as the pipeline's output types.

- **Semantic-Dictionary (SEM-DICT).** The feature fires consistently on the *same* semantic token across probes—a dictionary-like detector for a specific concept (e.g., a feature that reliably peaks on "Texas" whenever Texas is mentioned).
- **Semantic-Concept (SEM-CONC).** The feature peaks on semantic tokens across a small family of related tokens rather than on a single one—a concept-level rather than token-level detector, typical of middle layers.
- **Relationship (REL).** The feature does not concentrate its activity on any particular token; it fires diffusely across the probe, with a comparatively dense activation pattern. These features appear to encode a relation or context rather than a named entity.
- **Say-X (SAY-X).** The feature peaks on functional tokens (e.g., `is`, `the`) in predictable positions relative to a semantic target, and sits in the later half of the network. After the ± 7-token directional search (Section 3.5), the feature is named by the *target* semantic token that follows or precedes the functional peak.

The exact thresholds and priority ordering are in Section B. A post-hoc clustering analysis of the per-feature CPAS metrics confirms that the four-role partition is best read as a deliberate *coarse-graining* of a finer natural geometry; details in Section J.

3.7. Supernode formation and naming

Same-role same-name features form a supernode. Naming is role-specific.

Semantic features (both SEM-DICT and SEM-CONC) are named by the semantic token at which they activate most strongly: a feature that consistently peaks on "Texas" receives the name *Texas*.

Say-X features are named by the semantic token they appear to promote: e.g., *Say (Austin)* for a feature peaking on `is` immediately before "Austin".

Relationship features are named by the highest-activation semantic token . The format is *(token) related*, producing names like *(containing) related* for features that fire diffusely on spatial-relationship phrases.

seed	The	capital	of	the	state	containing	Dallas	is 3.6								
probe 1	A	major	city	in	Texas	,	USA	is	Dallas							
probe 2	The	capital	city	of	Texas	is 26.3	Austin									
probe 3	A	state	in	the	United	States	is	Texas								
probe 4	The	primary	city	serving	as 17.0	the 20.9	seat	of	government	for	a	state	is 29.9	the 62.9	capital	city
probe 5	The	state	in	which	a	city	is	located	is	the	state	containing				

Figure 2. Feature `20-clt-hp:74108` (Say "capital"). Activations on the seed prompt and five concept-targeted probes; peak intensity is shaded in green. On the seed alone the feature looks like a peak on the copula `is`. Across probes it instead peaks on a variety of functional tokens (`is`, `the`, `as`) whose only consistent property is that they sit immediately before the semantic target *capital*; the maximum activation (63.31) lands on the `the` preceding "capital" in probe 4. The cross-prompt pattern identifies this feature as a procedural output-promotion (SAY-X) feature for *capital*, not a semantic detector for `is`.

3.8. Causal validation of grouping

We test whether supernodes have an operational causal consequence. For a source–target pair (e_A, e_B), the intervention asks: if we suppress the features labeled as related to the source entity and amplify the features labeled as related to the target entity, does the model redirect its answer from e_A toward e_B? For each intervened feature, $M \cdot v_{\text{orig}}$ scales the decoder vectors and adds the result to the residual stream at the downstream layers. Attention is not frozen during the intervention; positive results are stronger evidence than in patching of frozen-attention where some direct effects are forced by construction (Ameisen et al., 2025).

We use two primary metrics: **Hit%**, the fraction of pairs whose generated continuation contains the target's first-subword token; **vsMax**, the maximum over the generated trajectory of the target's logit minus the best other answer in the domain (positive means the target is ahead). A small set of secondary diagnostics is described in Section B.

For each swap pair we construct a random feature set with the same feature count and per-layer histogram as the labeled intervention, drawing features from *outside* every concept-aligned supernode in the domain so that the only property that varies is concept alignment. Three deterministic replicates per pair yield Hit-rate and vsMax distributions over random feature sets.

As a second baseline, we also run an influence-matched top-K control. For each labeled swap, this baseline selects the smallest prefix of features ranked by graph node influence whose cumulative source- and target-side influence matches the labeled intervention budget. The top-K control uses the same ablation/amplification signs, but discards CPAS labels, field names, and supernode membership. It therefore tests whether the effect is explained merely by intervening on high-influence graph nodes, rather than by concept-aligned grouping; details are in Section M.

Each domain has 3 semantic fields connected to the two-hop logic (input, intermediate, answer). For each pair, 7 variants are run: 3 single-field, 3 two-field, 1 all-three. Each selected field drives both ablation and amplification; this isolates which fields carry the causal signal.

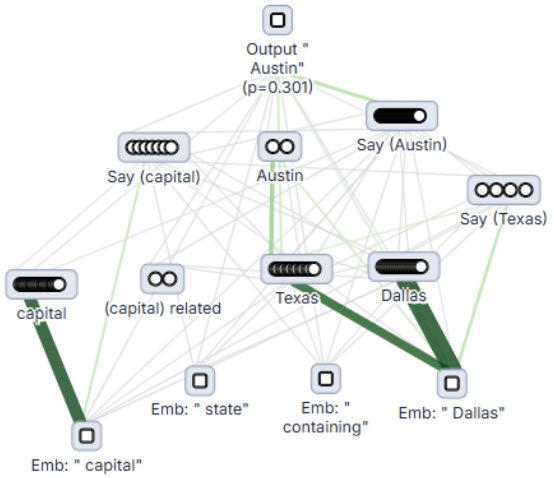

Figure 3. Concept-aligned subgraph for the seed "the capital of the state containing Dallas is". Token embeddings (`Emb`) are at the bottom; the model's predicted continuation (*Austin*, p=0.301) is at the top. Each supernode (rounded box) groups same-role same-name features and is named by its role: semantic (*capital*, *Texas*, *Dallas*, *Austin*), relationship (*(capital) related*), or output-promoter (*Say (capital)*, *Say (Texas)*, *Say (Austin)*). Edge thickness is proportional to influence on the output.

When a swap misses at the default M_{amplify}=20, we search for a better amplifier in two phases. Phase 1 probes a coarse geometric grid and stops at the first hit. If Phase 1 finds no hit, Phase 2 uses the fact that the steering effect typically has a sharp onset: we compute the KL divergence of the steered output against the unsteered baseline at each probed M, locate the interval in which KL rises most steeply, and binary-refine inside that interval for 6 further steps, accepting a hit at any refinement point. Full pseudocode in Section G.

Table 1. **Prompt probes beat matched-random and influence-matched top-K across 4 domains.** Each row reports three methods on the same per-domain entity set. *Ours*: per-pair best field-additivity variant with adaptive M-search on labeled features (Section G, Table 5). *Rand.*: matched-random control under the same per-pair best-of construction (the best across the 3 replicates with adaptive M-search; Section L, Table 11). *Top-K*: per-pair influence-matched top-K-by-graph-influence baseline with adaptive M -search (Section M, Table 12). *Hit%*, the fraction of pairs whose generated continuation contains the target's first-subword token. *vsMax* is the mean per-pair best logit margin against the strongest competing dataset answer

		Ours		Rand.		Top-K	
Domain	N	Hit%	vsMax	Hit%	vsMax	Hit%	vsMax
USA	2,450	72.8	+6.15	0.7	−1.23	4.2	−1.92
Books	90	77.8	+10.38	0.0	+0.19	4.4	+1.57
Products	132	41.7	+5.42	7.6	+1.17	1.1	+0.63
Paintings	90	18.9	+3.45	1.1	+1.27	7.1	+0.16

4. Results

4.1. State capital case study

Lindsey et al. (2025a) use attribution graphs to investigate two-step reasoning with the prompt "The capital of the state containing Dallas is", and we start from the same template. The Gemma 2 CLT attribution graph for this prompt contains 1,182 features at our cumulative-influence threshold. Running these features through the probe-prompt pipeline (Figure 1) places 458 of them into eight concept-aligned supernodes (*capital*, *state*, *Texas*, *Dallas*, *Austin*, *Say (capital)*, *Say (Texas)*, *Say (Austin)*); the resulting subgraph keeps a Neuronpedia completeness of 0.83 and a replacement of 0.53. *Completeness* is the fraction of incoming edges to all nodes of the subgraph that originate from grouped features weighted by influence on the output, while *Replacement* is the fraction of end-to-end influence from input tokens to output logits that flows through grouped features. As a reference point, the human-annotated subgraph featured on Neuronpedia for the same prompt[1] pins 21 features into five named supernodes and reaches a completeness of 0.70 and a replacement of 0.30.

We measure how useful the automated clustering is by entity-swap experiments: for a (source, target) pair we ablate features from the source state and amplify the corresponding features from the target state, and check whether the model's predicted capital is redirected to the target's capital. Because the only human-labeled graph available is the Dallas one, every swap uses Dallas as target and one of the other 49 states as source; the target supernodes come from the human annotation, while the source supernodes always come from the auto pipeline (no human annotation exists for the other 49 graphs). Across the 49 sources, our supernodes redirect the prediction to the target capital on $40/49$ sources versus $38/49$ for the human-labeled supernodes and $0/49$ for the random control; the top-K node-influence baseline saturates at $6/49$ when given $\sim$100 features per call (Figure 4).

[1]Public graph slug `gemma-fact-dallas-austin` on `neuronpedia.org`.

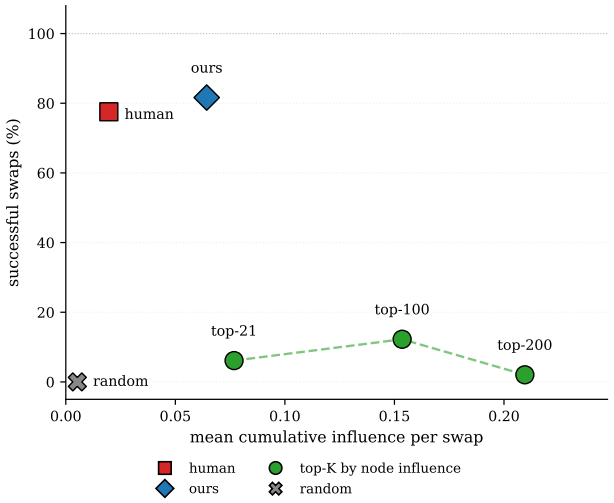

Figure 4. Successful swaps (%) vs. mean cumulative influence per swap. Each point denotes one approach evaluated on the 49 non-Dallas USA states. The target is fixed to Dallas/Texas/Austin, since human annotation is available only for that graph. The y-axis reports the successful-swap rate: the fraction of source states for which at least one field variant redirects the model's predicted capital to *Austin*. The x-axis reports the mean, across swaps, of the cumulative node_influence carried by the amplified features in the target graph. Both labeled methods reach approximately 80% successful swaps, whereas the top-K control family saturates near 10% despite using 1.09×–2.93× more per-swap influence than our method.

We find that it is necessary to steer different sets of clusters to correctly induce the entity swap - one can either steer the 'input' field, the city name, the 'intermediate' field, the state, or the 'answer' field, the correct capital, as well as any combination. When performing all possible state swaps, the predicted capital is redirected to the intended target on 72.8% of pairs, see Figure 5. By default we steer on the 3 concepts but if another steering combination is better we color code the cell by the combination used. The vertical striping in the swap matrix further suggests that the optimal subset is largely target-conditioned: for a given target state, the same field combination often transfers to many different

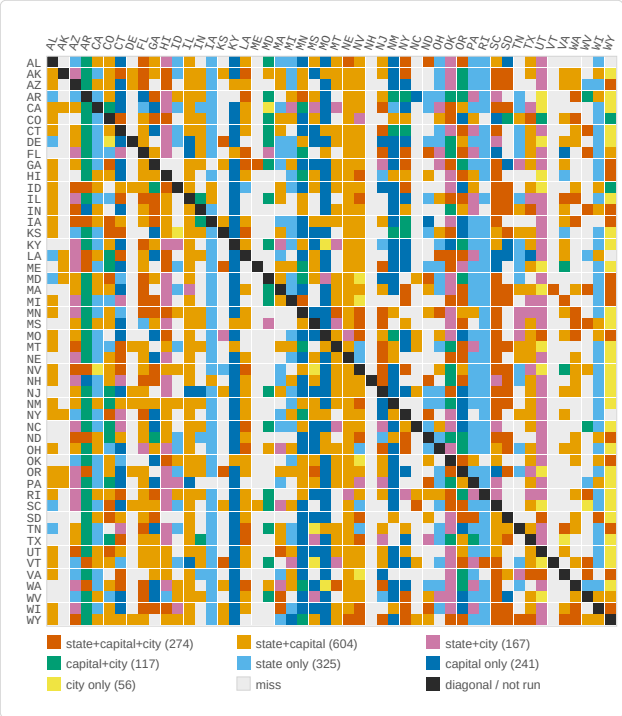

Figure 5. Source → target swap matrix on the 50-state USA panel (n=2,450 pairs, hit-rate 72.8%). By default we steer on all three concept fields (state + capital + city); each cell additionally considers every smaller field subset and reports the winning configuration. The winner is the variant that hits the target capital (steered top-1 = target capital first-token); when several variants hit, we prefer the one that maximises the steered margin of the target logit over the next-best capital in the dataset. Cells are coloured by the winning field subset; the 3-field default wins only 15% of cells, the rest succeed with a strict subset. Grey cells: miss; black on the diagonal: pair not run.

source states.While most states can be used as sources, some of the states can barely be used as targets, like NV, ME, NH, VT AK. This likely reflects the fact that their capitals already have low baseline logit probability and are often not top-logit predictions even in the unsteered setting.

4.2. Generalization to other domains

When developing the probing technique, we focused on the state capital prompt template, which might 'overfit' our decisions to making supernodes that work on this specific setting but don't translate well to other types of prompts and problems. Because of this we create other 3 two-hop datasets, and use our pipeline as is to produce labeled supernodes for graphs computed for these tasks.

We find that in all domains, our labeled supernodes lead to both a higher number of entity swaps (Hit%) as well as a higher average larger logit gap between the target token and other valid answers (Table 1), when compared with randomly selecting features, as well as using top-k features

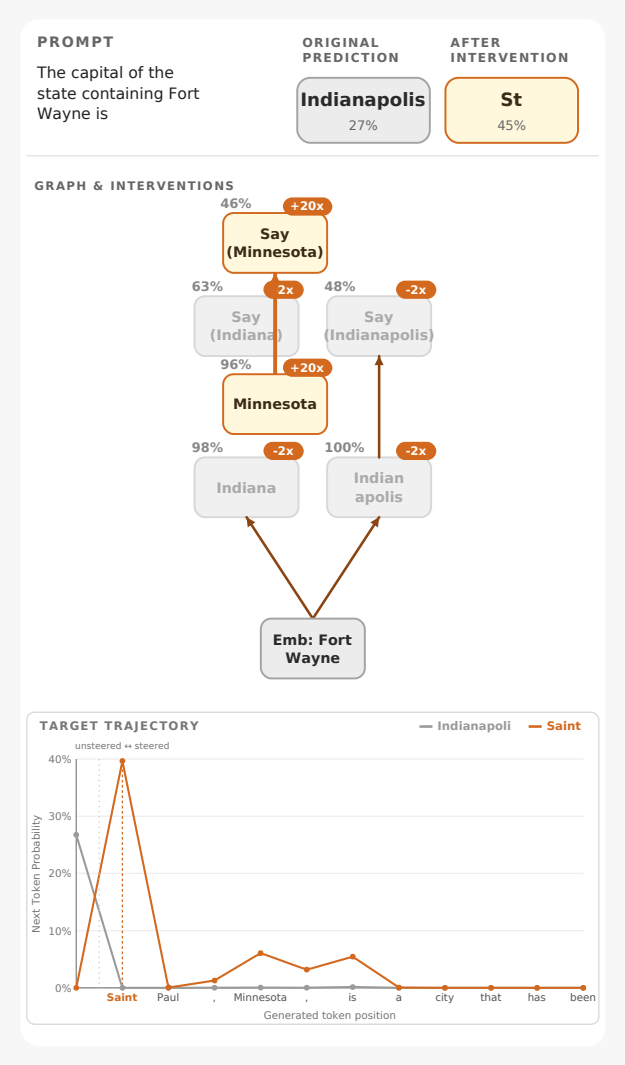

Figure 6. A representative entity-swap intervention on the USA panel: Indiana→Minnesota, probed via "the state containing Fort Wayne". The top row shows the prompt and the pre/post-intervention prediction (`Indianapolis` → `St`, i.e. `Saint Paul`). The middle row shows the labeled supernodes with ablation ($-2\times$) of the source state/capital nodes and amplification ($+20\times$) of the target state/capital nodes. The bottom row shows the next-token probability sweep at M=0 vs. M=20, with the target trajectory (`Saint`) rising at the steered position while the original (`Indianapolis`) collapses.

by graph influence at the same per-pair influence budget (Section M). On the other hand, while both the States and the Book datasets have a majority rate of swapping entities, the other datasets don't have such high steering performances. Because there are no human labeled equivalents, it is hard to establish a good base performance, being unclear if the failure comes from our pipeline, the attribution graphs, or the CLT figures.

Also in these domains we find that correctly choosing the

subset to steer can drastically change the effectivness of steering, with the most effective concepts to steer being the intermediate and answer fields, working better than steering all three fields. (Section F).

We find that a single steering strength for all the experiments is not the most optimal and that there are two different steering strengths, with peaks near $M{\sim}2.4$ (47% of labeled hits) and $M{\sim}6.9$ (40%), that recover most of the swap performance (Section G). Values outside these regimes can fail, with low M leaving the source attractor intact and high M sometimes overshooting the target and disrupting the continuation. These is different from what was reported in the single-feature SAE setting of Templeton et al. (2024), in which clamping at roughly $\pm 10\times$ maximum observed activation saturates the behavior.

5. Limitations

No grouping-method baseline. We compare labeled supernodes against matched random controls rather than against geometric or learned clustering because there is no canonical feature-clustering baseline in the SAE/CLT literature. The harness (§3.8) is method-agnostic and we encourage future groupings to be evaluated on it.

Attention-circuit blindspot. Attribution graphs freeze attention; our swaps run with attention free. The asymmetry plausibly accounts for part of the residual miss rate; attention-aware variants are future work (Section 6).

Threshold sensitivity is unmeasured. Decision-rule thresholds were iteratively refined on pilot circuits; we did not run a formal ± 10–20% sensitivity sweep. Thresholds are configurable defaults in the released code.

Domain scope: two-hop factual recall only. Our four evaluation domains are all two-hop factual-recall prompts. Whether the pipeline transfers to relational tasks with different structure (multi-hop reasoning, antonymy, code, dialogue, instruction-following) is untested.

Single model and language. In the three-level framing of Geiger et al. (2025) we claim Levels 1 and 2 only; we don't think that our current pipeline allows to have a Level 3 (full mechanistic identification of latent variables) understanding. All experiments use Gemma-2-2B-it with CLT-HP on English text(Section 3.1); cross-architecture, cross-dictionary, and multilingual generalization are open.

6. Discussion and Conclusion

Methodological lessons. Three findings in this paper may inform intervention practice in related settings. First, matched random controls need to be *structurally* matched—same feature count, same per-layer histogram, and sampled outside every concept-aligned supernode—because otherwise the labeled–random gap is easy to misstate: in three of our four domains, matched random feature sets produce *more* source-suppression than the labeled intervention does, even though they redirect the model nowhere in particular. Second, the $M{=}20$ amplification default that is standard for single-feature steering does not carry over to the multi-feature CLT setting studied here: our winning values of M are bimodal at roughly 2.4 and 6.9, and a coarse sweep over M rescues a substantial fraction of misses. Third, semantic fields are not interchangeable for intervention purposes: restricting to intermediate- and answer-field features gains 14–33 pp of Hit-rate over using all three fields.

Polysemanticity and cross-prompt activation. CLT and SAE features are not guaranteed to be monosemantic in practice. Templeton et al. (2024) document *feature splitting*—a coarse feature in a smaller dictionary splits into several finer features as dictionary capacity grows—indicating that "one feature" is often a bundle of related meanings. Lindsey et al. (2025a) note this explicitly: some member features are polysemantic in a dictionary-wide sense, and are kept in the supernode only on account of the facet relevant to the traced prompt. Balcells et al. (2024) document related pass-through and specialization phenomena layer-by-layer. Probe prompting formalizes this ad-hoc practice. By comparing a feature's activation across concept-targeted probes, CPAS assigns each feature a *prompt-contextual* role: the feature is monosemantic enough, in the context of the prompt family under study, to serve as the unit of a supernode, even when it is polysemantic across the dictionary as a whole. The same logic suggests that the method is not strictly tied to a trained sparse dictionary and can in principle be applied to raw MLP neurons, which are known to be substantially more polysemantic: when neurons are inspected on a small set of carefully chosen probes instead of on corpus-wide activations, a usable functional signature can emerge even without the sparsity prior.

Future work. Several directions seem natural. An obvious first extension is to make the intervention attention-aware, either by incorporating edge-attribution information or by also swapping attention-head outputs along with CLT feature activations; replicating the main findings on Gemma-2-9B, Llama-3, and alternative feature dictionaries would test how much of the current picture is specific to Gemma-2-2B-it and CLTs. A third direction is multilingual extension: our current prompt templates are English-only, and we have observed (Section C) that the functional-token vocabulary and the directional search it drives do not transfer cleanly to, for example, French.

Two further directions deserve specific mention. The first concerns the *probe set* itself. Our current probes are positive variations on the seed prompt ("the ⟨entity⟩ is"), and we treat the set as fixed across features. There is no reason to do so. Contrastive or near-miss probes—probes that differ minimally from the seed along a concept dimension the model might confuse—should be considerably more discriminating. At longer context lengths, standard for frontier models, one can imagine agentic variants of the harness that treat probe design as a detective game: an agent with a bounded action budget chooses which substring or position to intervene on next, based on what the previous probe revealed. The second direction concerns the feature-role taxonomy itself. The four-role vocabulary (Semantic-Dictionary, Semantic-Concept, Relationship, Say-X) is an empirical extension of the roles that appear in circuit-tracing case studies (Lindsey et al., 2025a), and the fact that it generalizes across four unrelated domains suggests there is a useful "model-biology" object hiding behind it; sharpening this taxonomy—adding roles, subdividing existing ones, or reducing them—is a research question in its own right. Finally, the grouping machinery itself is open: learned CPAS classifiers trained on the rule-based decisions could replace the hand-authored decision tree with a smoother map from signatures to roles.

Conclusion. Probe prompting shows that circuit-level interpretability of attribution graphs can be partially automated with a transparent, rule-based pipeline, and that the resulting labels can be used to perform entity swaps, an intervention technique that can be used to quickly evaluate whether supernode labels correctly cluster features into their actual behavior in the computational graph.

References

Ameisen, E., Lindsey, J., Pearce, A., Gurnee, W., Turner, N. L., Chen, B., Citro, C., Abrahams, D., Carter, S., Hosmer, B., Marcus, J., Sklar, M., Templeton, A., Bricken, T., McDougall, C., Cunningham, H., Henighan, T., Jermyn, A., Jones, A., Persic, A., Qi, Z., Thompson, T. B., Zimmerman, S., Rivoire, K., Conerly, T., Olah, C., and Batson, J. Circuit tracing: Revealing computational graphs in language models, 3 2025. URL https://transformer-circuits.pub/2025/attribution-graphs/methods.html. Transformer Circuits Thread.

Balcells, D., Lerner, B., Oesterling, A., Bloom, J., and Nanda, N. Evolution of SAE features across layers in LLMs. arXiv:2410.08869, 2024. URL https://arxiv.org/abs/2410.08869.

Bills, S., Cammarata, N., Mossing, D., Tillman, H., Gao, L., Goh, G., Sutskever, I., Leike, J., Wu, J., and Saunders, W. Language models can explain neurons in language models, 2023. URL https://openaipublic.blob.core.windows.net/neuron-explainer/paper/index.html. OpenAI.

Bricken, T., Templeton, A., Batson, J., Chen, B., Jermyn, A., Conerly, T., Turner, N., Anil, C., Denison, C., Askell, A., Lasenby, R., Wu, Y., Kravec, S., Schiefer, N., Maxwell, T., Joseph, N., Hatfield-Dodds, Z., Tamkin, A., Nguyen, K., McLean, B., Burke, J. E., Hume, T., Carter, S., Henighan, T., and Olah, C. Towards monosemanticity: Decomposing language models with dictionary learning, 2023. URL https://transformer-circuits.pub/2023/monosemantic-features/. Transformer Circuits Thread.

Geiger, A., Ibeling, D., Zur, A., Chaudhary, M., Chauhan, S., Huang, J., Arora, A., Wu, Z., Goodman, N., Potts, C., and Icard, T. Causal abstraction: A theoretical foundation for mechanistic interpretability. *Journal of Machine Learning Research*, 26, 2025.

Hanna, M. and Ameisen, E. Cross-layer transcoder feature circuits reveal latent planning in language models. In *International Conference on Learning Representations (ICLR)*, 2026. arXiv:2604.12493.

Hanna, M. and Piotrowski, M. CLT-HP: Cross-layer transcoders for Gemma-2-2B-it. Neuronpedia transcoder dictionary, 2025. URL https://www.neuronpedia.org/transcoders-hp.

Heap, T., Lawson, T., Farnik, L., and Aitchison, L. Sparse autoencoders can interpret randomly initialized transformers. In *International Conference on Machine Learning (ICML)*, 2025.

Lindsey, J., Ameisen, E., Hanna, M., Lewis, O., Pearce, A., Gurnee, W., Turner, N. L., Chen, B., Citro, C., Abrahams, D., Carter, S., Hosmer, B., Marcus, J., Sklar, M., Templeton, A., Bricken, T., McDougall, C., Cunningham, H., Henighan, T., Jermyn, A., Jones, A., Persic, A., Qi, Z., Thompson, T. B., Zimmerman, S., Rivoire, K., Conerly, T., Olah, C., and Batson, J. On the biology of a large language model, 3 2025a. URL https://transformer-circuits.pub/2025/attribution-graphs/biology.html. Transformer Circuits Thread.

Lindsey, J., Ameisen, E., McGrath, T., and Nanda, N. Attribution graphs for dummies – Part 1: What are attribution graphs? Neuronpedia Podcast (video panel), 2025b. URL https://www.youtube.com/watch?v=ruLcDtr_cGo.

Lindsey, J., Ameisen, E., Nanda, N., Shabalin, S., Piotrowski, M., McGrath, T., Hanna, M., Lewis, O., Tigges, C., Merullo, J., Watts, C., Paulo, G., Batson, J., Gorton, L., Simon, E., Loeffler, M., McDougall, C., and Lin, J. The circuits research landscape: Results and perspectives, 2025c. URL https://www.neuronpedia.org/graph/info. Neuronpedia.

Marks, S., Rager, C., Michaud, E. J., Belinkov, Y., Bau, D., and Mueller, A. Sparse feature circuits: Discovering and editing interpretable causal graphs in language models. In *International Conference on Learning Representations (ICLR)*, 2025.

Neuronpedia. Neuronpedia: Open interpretability platform and apis, 2025. URL https://www.neuronpedia.org/. Documentation: https://docs.neuronpedia.org/.

Paulo, G., Mallen, A., Juang, C., and Belrose, N. Automatically interpreting millions of features in large language models. arXiv:2410.13928, 2024. URL https://arxiv.org/abs/2410.13928.

Safety Research contributors. circuit-tracer: Tools for finding circuits with transcoders. https://github.com/safety-research/circuit-tracer, 2025.

Templeton, A., Conerly, T., Marcus, J., Lindsey, J., Bricken, T., Chen, B., Pearce, A., Citro, C., Ameisen, E., Jones, A., Cunningham, H., Turner, N. L., McDougall, C., MacDiarmid, M., Freeman, C. D., Sumers, T. R., Rees, E., Batson, J., Jermyn, A., Carter, S., Olah, C., and Henighan, T. Scaling monosemanticity: Extracting interpretable features from Claude 3 Sonnet, 2024. URL https://transformer-circuits.pub/2024/scaling-monosemanticity/. Transformer Circuits Thread.

Transformer Circuits Team. Sparse crosscoders for cross-layer features and model understanding, 2024. URL https://transformer-circuits.pub/2024/crosscoders/. Transformer Circuits Note.

A. Reproducibility Statement

To support the workshop's emphasis on reproducibility and code/data access, we release the following anonymously for the review period.

- **Code.** Anonymous repository at https://anonymous.4open.science/r/attribution-graph-probing-anon (commit hash and license in `README.md`).
- **Datasets.** The full 44,596-run swap dataset (graphs, prompts, activations, supernode groupings, swap dumps for all five domains) at the same anonymized URL, with a per-domain manifest in Section R.
- **Interactive demo.** An anonymized HuggingFace Space at https://huggingface.co/spaces/anonymous-agp-review/attribution-graph-probing-anon that showcases concept swaps across 82 entities in 4 domains.
- **Per-domain CLI examples.** Section R gives end-to-end commands for the USA, Books, Products, Paintings, Sounds pipelines, including the exact τ, M, and seed used for each table and figure.
- **Determinism.** The pipeline is fully deterministic (sha256-seeded random controls; fixed temperature/seed/penalty for generation; no random initialization); per-feature checkpoints permit resumption from partial runs.
- **Hardware.** A single L4 GPU is sufficient to reproduce CPAS for any single entity in 14–24 minutes (graph generation + activation measurement + grouping + subgraph), excluding the swap sweep; the full swap sweep is documented per-domain in Section R.

We commit to de-anonymizing the URLs at camera ready and to maintaining the artifacts indefinitely.

Appendix

The appendix is organized to track the main paper: §B–§D give a detailed reference for the pipeline, its functional vocabulary, and emblematic feature examples per role; §E collects worked steering case studies; §F–§H back up the field-additivity, M-search, and steering-strength discussions of §4; §I gives the logit-shift regime taxonomy used throughout; §J validates the four-role partition against the natural cluster geometry of the CPAS metric space; §K is the human-curated Dallas case study with comparison with our pipeline and a fixed-K saturation control; §L–§M give per-domain labeled-vs-random and per-pair influence-matched top-K results; §N details the observation that suppression-is-easy, steering-is-hard §O–§P back up the cross-prompt and scaffold analyses; §Q discusses the weakest of our four domains (Paintings); §R is the reproducibility manifest; §S is the glossary. Every appendix section is self-contained and cross-referenced from the main text.

B. Pipeline Reference

B.1. End-to-end pipeline

The pipeline takes an attribution graph for a seed prompt and returns a collection of concept-aligned supernodes together with the data needed to run matched-control swap interventions. The pseudocode below summarizes the steps; the paragraphs that follow describe each step in plain language.

```
Input:  attribution graph G_p, seed prompt p, target logit y
Output: concept-aligned supernodes {S_i}, swap harness I

1. V_tau  = select_features_by_cumulative_influence(G_p, tau=0.95)
2. C      = llm_generate_concepts(p, K)
3. Probes = synthesize_probes(C, template_constraints)
4. for each feature f in V_tau, each probe q in Probes:
     A[f, q] = measure_activation(f, q)
5. CPAS[f] = aggregate_cross_prompt(A[f, *])
6. for each feature f in V_tau:
     role[f] = classify(CPAS[f], thresholds)
     name[f] = assign_name(f, A, blacklist, target_token_rules)
7. Supernodes = group_by((role, name)), stability >= 0.6
8. for each swap pair (e_A, e_B):
     I_lab  = build_swap(S(e_A), S(e_B), M_ablate=-2, M_amplify=20)
```

```
    I_rand = matched_random_control(I_lab)
    run(I_lab); run(I_rand)
9. adaptive_M_search(missed_pairs)
```

Step 1 (feature selection). Attribution graphs from Neuronpedia are generated with a relatively permissive node-inclusion threshold (`nodeThreshold=0.8`), a high edge-influence threshold (`edgeThreshold=0.85`), and a cap of `maxFeatureNodes=5000`. The graph typically contains 600–5,000 nodes; we then retain only the nodes whose cumulative influence sums to at least τ=0.95, which reduces each circuit to 200–700 features for batch experiments.

Step 2–3 (probe generation). The seed prompt is handed to a separate LLM together with a small specification that (a) lists the *concepts* we expect to find in the graph (entities, relations, answer tokens) and (b) constrains the probe templates to be syntactically close to the seed (same prepositions, same answer position, same length). The LLM returns a handful of concept-targeted probe prompts. Typical probe counts are 4–5 per seed.

Step 4–5 (activation measurement and CPAS). Each feature in V_τ is re-run on each probe and its activation measured with the Neuronpedia API. The per-(feature, probe) record stores the peak token, the peak position in the probe, the activation density, a sparsity ratio, a robust z-score, and a cosine similarity to the seed activation. These raw measures are then aggregated across probes into a compact per-feature signature (the CPAS).

Step 6 (classification and naming). A strict-priority decision tree (§3.6; thresholds in Table 3) assigns each feature one of four functional roles (Semantic-Dictionary, Semantic-Concept, Relationship, Say-X) or a fifth REVIEW bucket that is excluded from downstream analysis. Naming is role-specific (§3.7).

Step 7 (supernode formation). Same-role same-name features are merged into a supernode; features whose role or name is inconsistent across probes ($< 60\%$ probe-level stability) are marked *ungrouped* and excluded.

Step 8–9 (causal-validation harness). For every entity pair in the domain we build an additive entity-swap intervention (§3.8), run it at $M_{\text{ablate}}=-2$ / $M_{\text{amplify}}=20$ with attention *not* frozen, and compare it to three deterministic matched-random-control replicates (§3.8). Pairs that miss at the defaults enter adaptive M-search (§G).

B.2. CPAS measures

The CPAS is the seven-number summary that the decision tree operates on. Table 2 gives the formal definitions and the intuition for each field.

Field	Definition and intuition
`peak_consistency_main`	Fraction of probes whose peak token matches the feature's modal peak. High (≥ 0.8) means the feature is a dictionary-like detector for one particular token.
`n_distinct_peaks`	Number of distinct peak tokens across *active* probes. $= 1$ for a Semantic-Dictionary feature; larger for Semantic-Concept or Say-X.
`share_F` / `conf_F`	Share of active probes whose peak falls on a functional token / classifier confidence on that share. A feature with `share_F`≈ 1 and high `conf_F` is a Say-X candidate.
`func_vs_sem_pct`	$100 \cdot \left(\max_{\text{func}} - \max_{\text{sem}}\right) / \max_{\text{overall}}$. Positive when the peak is on a functional token; used to break ties between Say-X and semantic roles.
`sparsity_median`	Median of $(\text{peak} - \text{mean})/\text{peak}$ across active probes. Low values indicate a dense, diffuse activation pattern characteristic of Relationship features.
`conf_S`	$1 - \text{share_F}$, i.e. confidence that the peak is on a semantic token.

Table 2. CPAS fields used by the classification rules in §3.6.

B.3. Decision-rule thresholds

The decision tree is a strict-priority cascade: features are evaluated against the rules in the order below, and the first matching rule wins. Thresholds were iteratively refined on a small set of pilot circuits and then frozen before the 44,596-run sweep.

The layer-based asymmetry (Say-X requires layer ≥ 7, Sem-Conc accepts layer ≤ 3) encodes a rough prior observed in Gemma-2-2B: early layers are overwhelmingly semantic-content features; the "Say X" behavior of functional-token features promoting a specific answer is observed only in the later half of the network.

Priority	Role	Rule
1	SEM-DICT	peak_consistency_main ≥ 0.80 AND n_distinct_peaks ≤ 1
2	SAY-X	func_vs_sem_pct ≥ 50 AND conf_F ≥ 0.90 AND layer ≥ 7
3	REL	sparsity_median < 0.45
4	SEM-CONC	layer ≤ 3 OR conf_S ≥ 0.50 OR func_vs_sem_pct < 50
5	REVIEW	anything not matching the above (excluded)

Table 3. Classification thresholds used for the four functional roles (§3.6). Strict priority: the first matching rule wins.

Dictionary subtype: strict and fallback. The semantic role is split internally into three *subtypes* that determine downstream naming. Rule 1 (peak_consistency_main ≥ 0.80 and n_distinct_peaks ≤ 1) produces the strict *Dictionary* subtype: stable single-token detectors. Rule 4 produces either *Concept* (the conf_S or func_vs_sem_pct branch) or, when fired through its low-layer branch (layer ≤ 3), the *Dictionary (fallback)* subtype: early-layer semantic features whose peak token is not stable enough to satisfy Rule 1 but whose layer prior pulls them into a dictionary-like reading as the safest default. The two dictionary subtypes are reported together as SEM-DICT in every per-domain table; only the cluster-geometry analysis (§J) keeps them separate, which is where the *Dictionary (fallback)* bucket reveals additional internal structure.

C. Functional Vocabulary and Target-Token Mapping

The pipeline distinguishes *functional* tokens (copulas, articles, prepositions, conjunctions, relative pronouns, auxiliaries) from *semantic* tokens (content-bearing), and uses this distinction both in the Say-X rule and during supernode naming.

English functional vocabulary. We use a fixed list of 87 English functional tokens. The groups and their members are:

- **Copulas and copular contractions**: is, was, are, were, be, been, being, am, 's, 're, 'm.
- **Articles and demonstratives**: the, a, an, this, that, these, those.
- **Prepositions**: of, in, on, at, to, for, with, by, from, as, into, onto, upon, about, above, below, between, among, through, during, before, after.
- **Conjunctions**: and, or, but, nor, so, yet, for.
- **Relative pronouns and wh-words**: that, which, who, whom, whose, where, when, why, how.
- **Auxiliaries**: do, does, did, have, has, had, will, would, shall, should, can, could, may, might, must.

Directionality of the ± 7-token search. When a feature peaks on a functional token, we search the probe for the nearest *semantic* target in a window of ± 7 tokens. The direction of the search depends on the functional token's syntactic role: copulas, auxiliaries, articles, and most prepositions look *forward* (the semantic target is typically the noun phrase they introduce); of and possessive 's look *backward* (the semantic target is the head noun that governs the functional token); conjunctions and punctuation are searched *bidirectionally*, nearest-first. Multi-token semantic targets resolve to the first subword.

Cross-lingual behavior and a documented failure mode. We observed one clean cross-lingual failure during development, on a French version of the antonym seed "le contraire de 'petit' est". Running the English pipeline on a French prompt produced incoherent cross-prompt activations: functional tokens were often peaked on the English contraction 's or on French tokens (de, est) that were not in our vocabulary, so the Say-X rule either never fired or fired on the wrong side. Adding a small French functional vocabulary and expanding the blacklist (bridging tokens that appear in both languages) restored reliable grouping. We flag this as a concrete instance of the cross-architecture-and-language limitation discussed in §5.

D. Feature examples per Functional Role

To make the four functional roles of §B concrete, we hand-pick one canonical feature per role—five panels, since SEM-DICT is shown both in its strict and fallback subtype. Each panel stacks the seed prompt (top, bordered) with the five probe

prompts of the entity's CPAS dump; cell shading is the per-token activation normalized by the feature's global maximum across all probes, numbers are the raw activations, and values below the visibility threshold are left blank. Candidates were selected from the cross-dataset feature manifest and inspected manually; the metrics quoted in each caption are the raw values that placed the feature inside its branch of the decision tree of Table 3.

seed	The 4.6	company	that	makes	the	iPhone 13.8	was	founded	by			
probe 1	A	famous	product	is	the	iPhone 8.8						
probe 2	A	major	technology	company	is	Apple 0.6						
probe 3	The	founder	of	Apple 0.6	is	Steve	Jobs					
probe 4	The	person	who	started	and	created	a	company	is	called	the	founder
probe 5	The	person	who	established	a	company	is	its	founder			

Figure 7. SEM-DICT (strict) — *iPhone* entity, products domain, layer-0 feature 1285. The feature lights up on the literal token `iphone` in every probe and is silent almost everywhere else: the same surface token, every time, with no spread to morphological variants or related concepts. This is a canonical single-token detector that sits at the strict end of the dictionary branch of the decision tree.

seed	The	company	that	makes	Oracle	Database	was	founded	by 22.4			
probe 1	A	famous	product	is	Oracle	Database						
probe 2	A	major	technology	company	is	Oracle						
probe 3	The	founder	of	Oracle	is 2.6	Larry	Ellison					
probe 4	The	person	who	started	and	created	a	company	is 3.1	called	the	founder
probe 5	The	person	who	established	a	company	is 2.3	its	founder			

Figure 8. SAY-X — *Say (founder)* entity, products domain, layer-11 feature 9574. A late-layer feature whose only strongly-active position in each probe is the copula `is` immediately preceding `selected` next-token prediction. Because every active position is a functional token, the ± 7-token search of §C resolves the actual semantic target by looking forward from the copula—in this case, into the answer slot.

seed	The	first	name	of	the	painter 7.9	of 8.3	N 1.8	ighth 10.1	aw	ks	is
probe 1	A	famous	painting 13.5	is	N	ighth 7.7	aw	ks				
probe 2	A	famous	painter 11.1	is	Edward	Hopper 21.4						
probe 3	The	first	name	of	Edward	Hopper 20.7	is	Edward				
probe 4	A	person	who	creates	paintings 10.1	and 0.1	artworks 6.1	is	called	a	painter 7.5	
probe 5	The	given	name	of	a	painter 5.3	is	their	first	name		

Figure 9. SEM-CONC — *Hopper* entity, paintings domain, layer-4 feature 47039. The same feature fires on *Edward*, *Hopper* and on the painting tokens (*Nighthawks*) across all five probes, distributing its activation across several distinct semantic peaks anchored on a single coherent concept (the artist Edward Hopper). Multi-peak coverage of a concept is the visual signature that distinguishes SEM-CONC from the strict-dictionary case above.

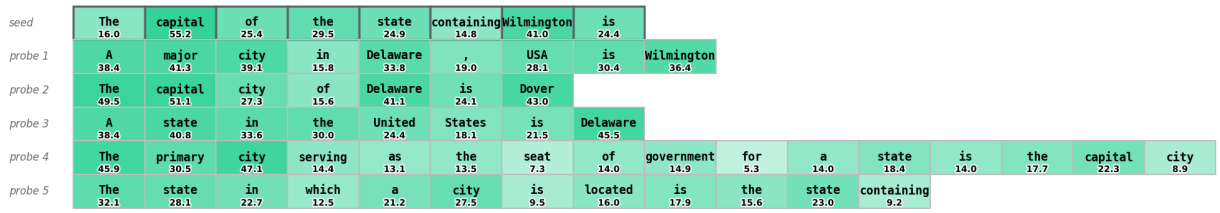

Figure 10. RELATIONSHIP — *(Capital) related* entity, U.S. states domain, layer-7 feature 66851. The feature is active on a substantial fraction of every probe: a diffuse green band runs across each row, with several peaks spread between functional and semantic positions and no single token dominating the activation budget. This combination—low sparsity, no clear peak target—is the visual signature of RELATIONSHIP features. Major peaks (*capital*, *Delaware*, *Wilmington*, *state*) inform possible main links.

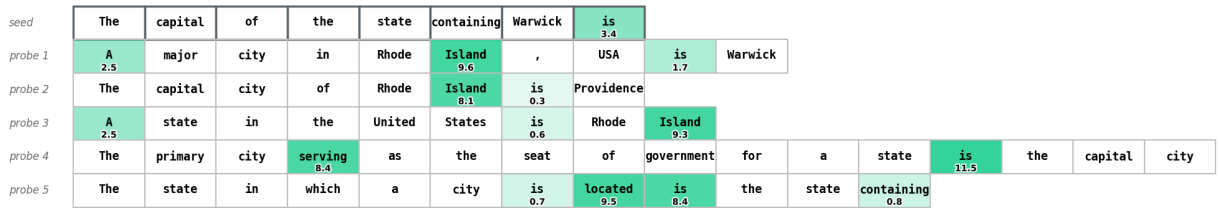

Figure 11. SEM-DICT (fallback) — *Island* entity, U.S. states domain, layer-3 feature 81381. The feature fires on every probe and behaves like a Rhode-Island detector, but its activation does not collapse onto a single surface token: it lights up both on a subword of the multi-token state name (`island`) and re-fires on functional tokens and general concepts. The fallback rule of §B is meant to capture this regime—dictionary-like specificity at a layer too shallow for the semantic concept criterion, with a peak that drifts across morphologically related tokens before the network has had room to collapse them onto a single representative.

E. Steering Case Studies

This section collects worked case studies that would be too discursive for the main text. Each entry gives the source and target entity, the field-additivity variant used, the intervention size, the vsMax achieved, and the first ∼10 generated tokens of the model's continuation.

E.1. Five success cases

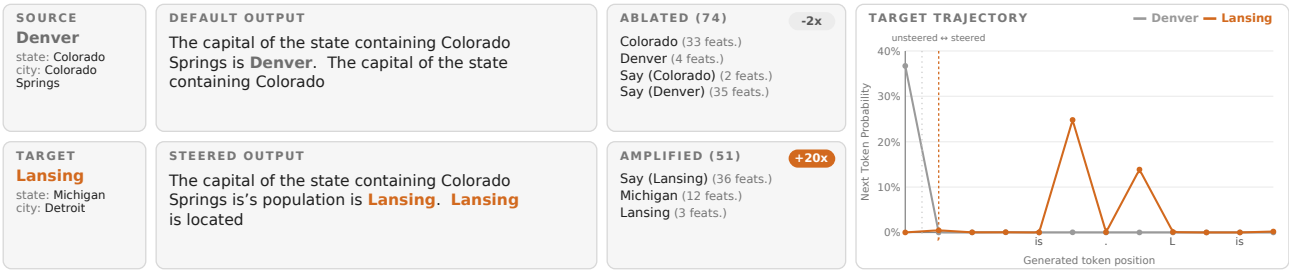

Figure 12. Colorado Springs → Detroit (`state+capital` variant). Steered "Lansing" wins at trajectory positions 4 and 6 (peak ∼25%) while the default "Denver" attractor is fully suppressed; the supporting Michigan and Say(Lansing) supernodes lift the answer without disturbing the prompt-anchor token. vsMax +17.5, target logit rank 342 → 1. Regime A in logit trajectory (target UP, source DOWN), clean flip, on-topic continuation.

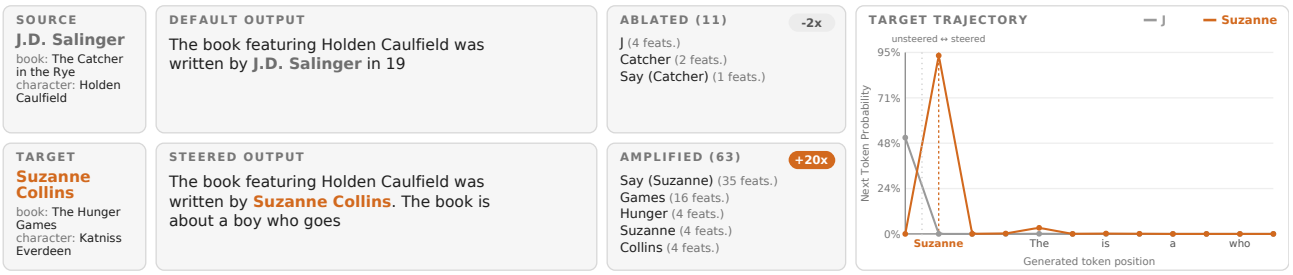

Figure 13. Holden Caulfield → Katniss Everdeen (`book+author` variant). Eleven ablated features remove "J.D."; 63 amplified features push "Suzanne" to $p \approx 0.95$ at position 0, and the steered continuation also recovers the correct book title ("The Hunger Games"). The amplify panel is dominated by Say(Suzanne) and a Hunger-Games scaffold of $\sim$ 16+4 features. vsMax $+21.79$, the largest single-pair vsMax in our dataset; illustrates how a small but correctly-targeted amplify set can dominate a graph $\sim 50\times$ its size.

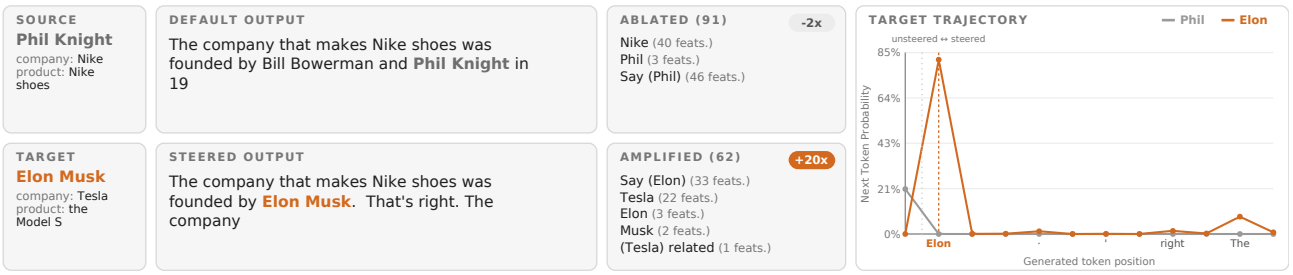

Figure 14. Nike → Tesla (`company+founder` variant). The orange "Elon" curve spikes to $\sim 85\%$ at position 0 and the grey "Phil" source vanishes immediately; the steered continuation also produces the correct company token "Tesla". The 91 ablated features are dominated by a single Nike supernode (40 features). vsMax $+12.25$, target rank $5{,}933 \rightarrow 1$, steered first-token probability p=0.82. Redirection is complete at position 0.

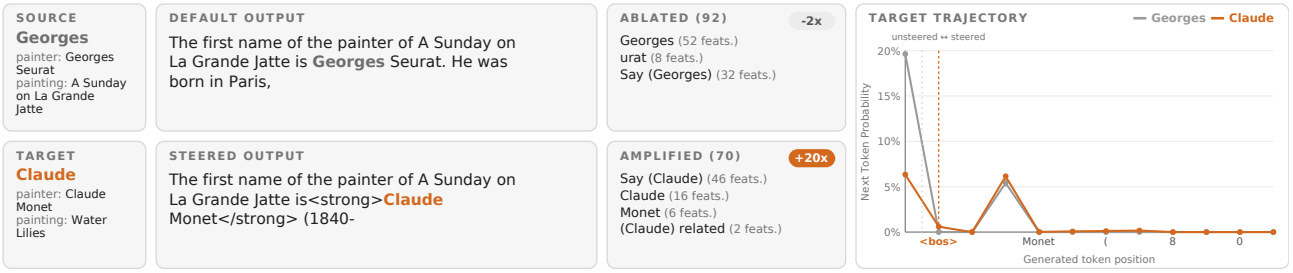

Figure 15. La Grande Jatte → Water Lilies (`painter+first_name` variant). The orange "Claude" curve barely peaks at $\sim 6\%$; "Georges" still dominates the trajectory and the steered continuation actually could re-emit "Georges Seurat". A Hit in name only – the position-0 logit prefers Claude. vsMax $+9.375$; The baseline rank of "Claude" was already 3. Illustrates that Paintings can be redirected only when baseline conditions are already favourable.

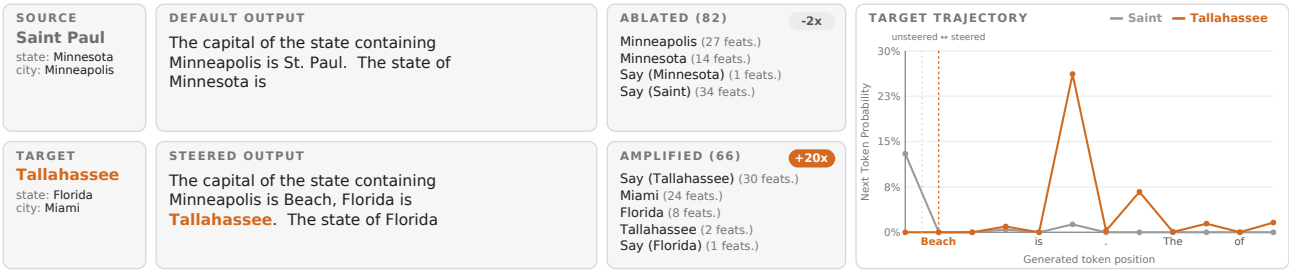

Figure 16. Minnesota → Florida (all-three-fields variant, `state+capital+city`, M=20). Position 0 emits the incoherent token "Beach"; "Tallahassee" only spikes to $\sim$ 30% at position 3, after the autoregressive decoder has already committed to a derailed continuation. The grey source curve never disappears, signalling residual leakage from the un-ablated source scaffold. vsMax +16.06 but flip@0 is `False`, and control-stability reaches 16.89 (very high collateral disruption). Clear illustration of the all-fields cost: position-0 disruption corrupts the first generated token even though the target logit eventually wins later.

E.2. Six failure cases covering four of the five failure modes

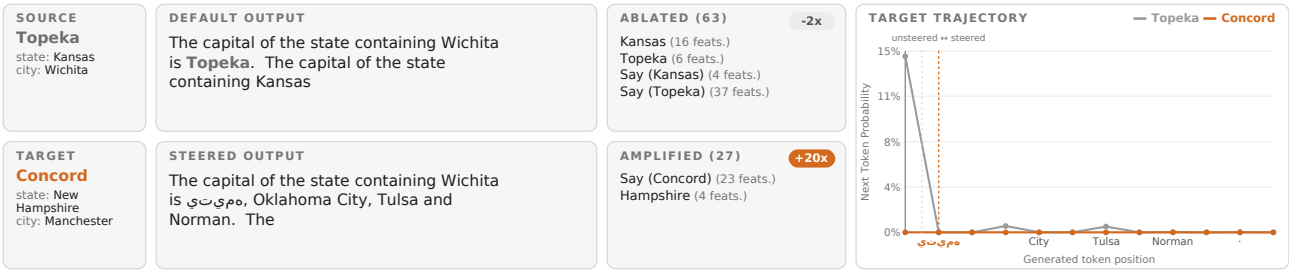

Figure 17. Kansas/Wichita → New Hampshire/Manchester (total failure). The amplified target features fail to lift "Concord" over the dominant "Oklahoma" attractor; the steered continuation collapses into a different neighbouring-state capital ("Oklahoma City. . . Tulsa"). Trajectory shows neither curve ever crosses $\sim$ 10% – a complete failure to retarget. vsMax −6.875, target rank 5,816. Kansas source features are successfully suppressed, but the amplified New Hampshire features do not encode Manchester-capital circuitry strongly enough. Failure mode: feature specificity.

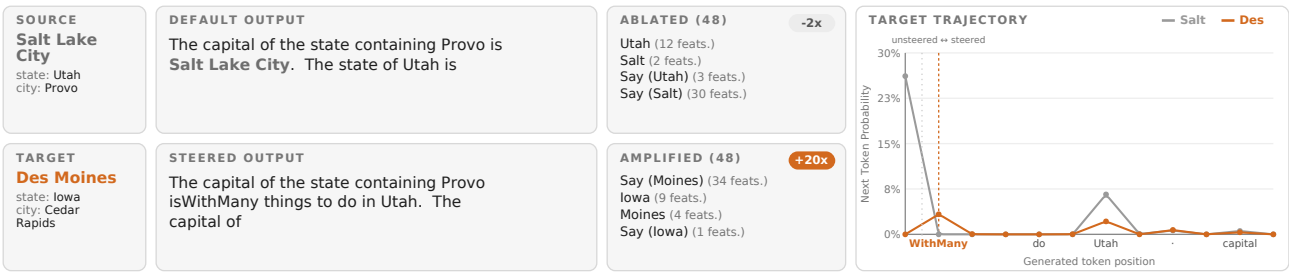

Figure 18. Utah/Provo → Iowa/Cedar Rapids (rank-hit misalignment). Textbook RkGrp-without-Hit case: the trajectory shows the target winning at position 0, but the steered output reverts to a Utah-themed ramble rather than naming Des Moines anywhere. vsMax +14.0, RkGrp 1; the target logit wins at position 0 but subsequent tokens are disrupted enough that the Hit criterion (target token anywhere in continuation) fails. Failure mode: feature-interaction noise.

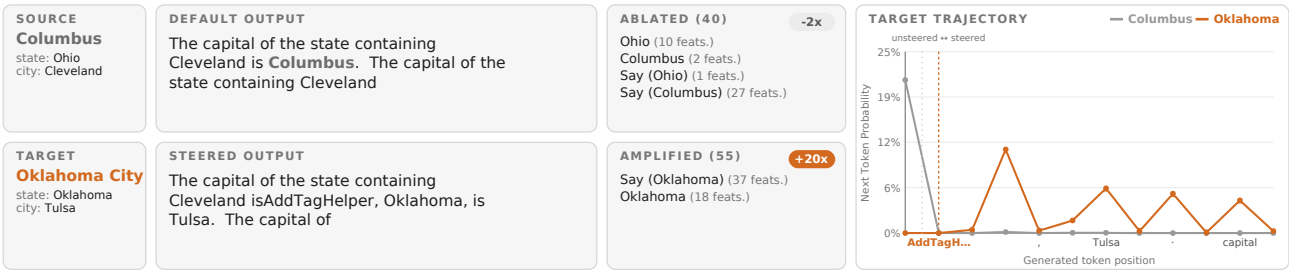

Figure 19. Ohio/Cleveland → Oklahoma/Tulsa (regime D). Position 0 emits "AddTagHelper" (a corrupted code-token), poisoning the autoregressive decoder; the orange "Tulsa" curve does rise later but never overtakes the grey source, so by the Hit criterion the intervention fails despite a positive vsMax. vsMax $+13.31$ but flip@0 is `False`; at $M{=}5$ this same pair becomes a Hit. Failure mode: severe overshoot, rescuable by M-reduction.

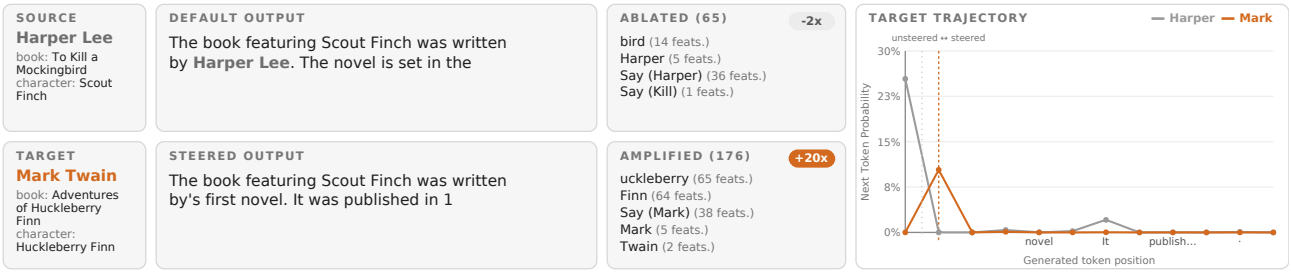

Figure 20. Scout Finch → Huckleberry Finn (`book+author` variant; field interference / substring confound). The amplified "Finn" supernode (64 features) and "uckleberry" (65 features) interfere with each other; the steered output produces neither "Mark Twain" nor a coherent book title, but a possessive fragment ("'s first novel"). The orange "Mark" peak at $\sim 8\%$ is dwarfed by the noise on adjacent positions. 176 amplify features (the highest count in Books), with a substring confound (`Finn` is a common suffix). Rescuable at the `book+author` variant but not at the all-three-fields baseline. Failure mode: field interference.

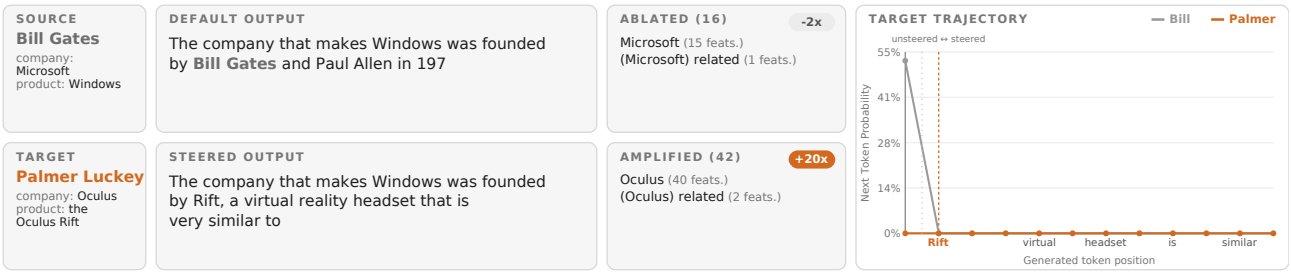

Figure 21. Windows → Oculus (`company`-only variant; product-identity collapse). With only 16 ablate / 42 amplify features, the trajectory shows "Rift" at $\sim 55\%$ at position 0; the orange "Palmer" curve never lifts off zero. The steered continuation reads as a product description rather than a founder attribution. The source (Windows/Microsoft) is suppressed cleanly but the "company" supernodes in the Products domain encode product identity; amplifying them therefore promotes the product name. Failure mode: feature specificity—a perfectly-executed intervention that targets the wrong entity because the label's extension does not match the target concept.

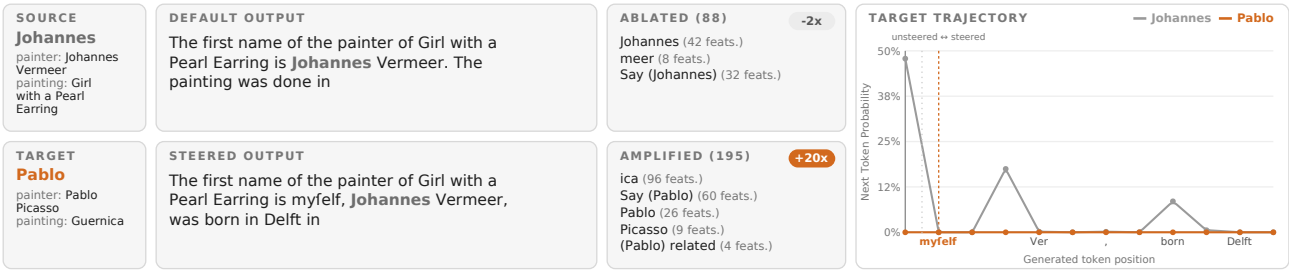

Figure 22. Girl with a Pearl Earring → Guernica (`painter+first_name` variant; cross-concept collapse). Even at 195 amplified features the orange "Pablo" curve never crosses $\sim 5\%$; the grey "Johannes" source still spikes to $\sim 50\%$ at position 2 and to $\sim 13\%$ at position 5, confirming the Vermeer circuit re-emerges in the steered continuation despite source ablation. 195 amplify and 88 ablate features fail to dislodge the source painter; "Pablo" ends up at rank 1,382 at position 0. The source circuit is deeply entrenched (Vermeer features are shared across the painting and its description) and the target is coarse (Picasso has many broader associations). Failure mode: feature specificity combined with low scaffold compatibility.

E.3. Threats to validity for case studies

Case studies select for extremes (the largest vsMax pairs and the most striking failures), and single-example interpretations should not be over-generalized. The distribution-level claims live in the full labeled-vs-random tables (§L), field-additivity tables (§F), and adaptive-M rescue rates (§G). Post-hoc explanations of success and failure (e.g. "the company supernodes encode product identity") are hypotheses supported by the particular pair.

F. Field-Additivity Detail

Each domain has three semantic fields: an *input* field (the entity being named in the prompt), an *intermediate* field (the bridging concept, e.g. the US state), and an *answer* field (the concept whose logit we measure). For each entity pair the field-additivity sweep runs seven intervention variants: three singletons, three pairs, and the all-three baseline. The bolded row is the best variant by Hit%.

USA (state, capital, city; 2,450 pairs/variant).

Subset	Hit%	vsMax	RkGrp	MedRk
city	8.3	−1.26	6.92	190
state	24.1	+2.38	2.72	14
capital	19.8	+0.63	3.69	11
state+capital	**43.6**	**+4.00**	**1.47**	**3**
state+city	15.8	+1.71	2.31	28
capital+city	15.9	+0.58	4.16	30
all 3	31.5	+2.86	1.72	5

Prompts are of the form "The capital of the state containing ⟨city⟩ is". The input field is the city (what the model reads), the intermediate is the state (the reasoning bridge), and the answer is the capital. The state+capital pair gives the best redirection; adding the city field degrades every metric—the mechanism we label *less is more*: city features encode what the model reads.

Books (character, book, author; 90 pairs/variant).

Subset	Hit%	vsMax	RkGrp	MedRk
character	8.9	+1.87	1.77	125
book	63.3	+6.69	1.20	2
author	10.0	+4.71	1.33	12
book+author	**66.7**	**+7.76**	**1.02**	**2**
character+book	15.6	+5.22	1.23	24
character+author	4.4	+3.45	1.30	76
all 3	5.6	+5.97	1.03	17

Prompts are of the form "The author of the book featuring ⟨character⟩ is". The character field is the input; intervening on it alone moves almost nothing. The largest single-field effect comes from the book field (63.3% Hit on its own), and adding the author field only nudges the total up further (66.7%). Adding the character field undoes the effect almost entirely (5.6% Hit for all-three). This is the sharpest single-domain illustration of the less-is-more effect.

Products (product, company, founder; 132 pairs/variant).

Subset	Hit%	vsMax	RkGrp	MedRk
product	0.8	+1.90	1.61	137
company	9.1	+2.62	1.39	128
founder	16.7	+2.08	1.27	18
company+founder	**34.1**	**+3.06**	**1.27**	**18**
product+company	2.3	+2.78	1.31	93
product+founder	3.8	+2.54	1.23	48
all 3	23.5	+3.47	1.20	26

Paintings (painting, painter, first_name; 90 pairs/variant).

Subset	Hit%	vsMax	RkGrp	MedRk
painting	3.3	+1.50	1.44	71
painter	1.1	+1.68	1.30	66
first_name	6.7	+1.46	1.41	90
painter+first_name	1.1	+1.69	1.30	65
all 3	3.3	+1.55	1.32	66

Paintings is the weakest of the four domains. The answer field is the painter's first name, a coarse token with many non-painting associations (`Claude`, `Pablo`, `Leonardo`); and the `painter` supernode overlaps with `first_name` by construction (the painter supernode name already contains the first name). The less-is-more effect is correspondingly weak: the +3.4 pp main-text gap is the smallest of the four domains.

F.1. Cross-domain single-field aggregates

Averaging over all four domains, the three field types produce very different redirection strengths. Table 4 reports the aggregate numbers: the *intermediate* field carries the strongest single-field signal (28.5% Hit), followed by the *answer* field (24.7%), with the *input* field trailing at 4.8%. Input-field features encode what the model reads; intermediate- and answer-field features encode what the model should produce.

Field type	Hit%	MedRk	vsMax	RkGrp
Input	4.8	131	+1.00	2.94
Intermediate	28.5	53	+3.34	1.65
Answer	24.7	33	+2.22	1.93

Table 4. Per-field aggregates across all four domains. Each row averages the corresponding single-field variant across the four domains, weighted equally.

G. Adaptive M-Search

G.1. Two-phase protocol

When a swap pair misses at the default M_{amplify}=20, we search for a better amplifier in two phases.

Phase 1 (coarse geometric probe). We evaluate $M \in \{0.1, 0.25, 0.6, 1.5, 4.0, 10.0, 20.0\}$ in that order and stop at the first hit. The grid is geometric so that the same number of probes covers both the fine-grained low-M region where some pairs find a hit near $M{\sim}2$ and the $M \approx 20$ neighborhood that the default already explored.

Phase 2 (KL-transition binary refinement). If Phase 1 produces no hit, we compute the KL divergence between the steered and unsteered output distributions at each Phase-1 grid point. The KL curve is close to linear in M for most pairs ($R^2 > 0.93$

for 8 of 10 probed USA pairs), but the hits that do exist tend to cluster near the *onset* of meaningful response. We therefore identify the interval in which KL rises most steeply (the "KL transition"), place a binary-refinement search inside that interval, and run 6 additional probes, accepting a hit at any refinement point. The rationale is that the model's response to multi-feature amplification typically has a sharp onset at a pair-specific M, and hits live near that onset.

G.2. Per-domain rescue totals

Table 5 summarizes the rescue rates split by condition (Labeled, Field-Additivity, and matched-random) and domain.

Cond.	Domain	Eligible	New hits	Hit%	Before	After	Δ
Lab.	USA	1,681	287	17.1	31.4	43.1	+11.7
Lab.	Books	84	8	9.5	6.7	15.6	+8.9
Lab.	Paintings	86	6	7.0	4.4	11.1	+6.7
Lab.	Products	101	3	3.0	23.5	25.8	+2.3
FA	USA	740	74	10.0	69.8	72.8	+3.0
FA	Books	21	1	4.8	76.7	77.8	+1.1
FA	Paintings	78	5	6.4	13.3	18.9	+5.6
FA	Products	77	0	0.0	41.7	41.7	+0.0
Rand.	USA	2,445	12	0.49	0.20	0.69	+0.49
Rand.	Books	90	0	0.00	0.00	0.00	+0.00
Rand.	Paintings	90	1	1.11	0.00	1.11	+1.11
Rand.	Products	130	8	6.15	1.52	7.58	+6.06
Total	—	**5,623**	**405**	—	—	—	—

Table 5. Adaptive M-search rescue rates. "Eligible" is the number of pairs that missed at the default (M_{amplify}=20, all-three-fields). "Before" and "After" are the domain-level Hit% before and after the search. For the labeled (LAB.) and field-additivity (FA) blocks the rescue is per intervention configuration; for the matched-random (RAND.) block we apply the same outer adaptive-M harness to all three replicates per labeled pair and score the per-pair best-of-(replicate $\times$ {default, M-tuned}) under the same lexicographic rule. Labeled sweeps recover between 2.3 and 11.7 pp per domain; field-additivity sweeps recover more modestly; the symmetric matched-random sweep recovers ≤ 6.1 pp in every domain (Products' $+6.1$ pp is the largest, all from a small set of low-M rescues), confirming that the labeled–random gap of Table 1 is not an M-tuning artefact.

The overall pattern: 93% of adaptive hits are found in Phase 1 (coarse geometric), and the winning-M distribution is bimodal at $M \sim 2.4$ (47% of labeled hits) and $M \sim 6.9$ (40%), with a long tail to $M \geq 10$ (13%).

G.3. High-M rescue is null

To rule out the alternative explanation that the M=20 default is *too small*, we ran $M \in \{50, 100, 200\}$ on 80 near-miss pairs across three domains. The result was 0 new hits. Existing hits at M=20 degraded monotonically as M increased: at M=200 the generated continuation is typically dominated by a single high-frequency token from the amplification set, with no remaining structural coherence. KL between steered and unsteered saturates above $M \sim 50$, so the model's output distribution is essentially maximally disrupted long before M reaches 200.

G.4. Top-k rescue is null

A second alternative is that the M=20 signal is distributed over too many features, and that concentrating amplification on the top-k features by graph influence might yield targeted hits at high M. On `products`→`facebook` we ran the full factorial $k \in \{1, 3, 5, 10, 67\}$ with $M \in \{20, 50, 100, 200\}$ (six source entities, so 120 runs per cell). The result was 0 hits for any $k < 67$ at any M. Only the full 67-feature set achieves hits. A worked example: at k=1, M=20 the top-influence feature for `alibaba`→`facebook` generates "the founder of Alibaba in 1999…", at k=3 the output is garbled token sequences, at k=10 the first token is a programming-code token, and only at k=67 does the model correctly emit "Mark Zuckerberg…". Graph influence correlates *negatively* with stored activation in this domain ($\rho = -0.362$, p=0.003): the top-k by graph influence are the loudest but not the most informative features, and concentrating amplification on them produces generic "question answering" output rather than the target answer.

G.5. Five failure modes surfaced by the M-sweep

The pairs that remain unrescued after adaptive search fall into five qualitatively different failure modes. Identifying the five modes was itself a research output: the labels below are pointers to the investigation.

(1) Severe overshoot. A pair whose hit exists at small M but is destroyed at the default M=20. Rescuable by M reduction. Example: `oklahoma_tulsa` high-vsMax pairs, $37/42$ pairs rescued at M=5 (88% rescue rate). At M=20 the first token is typically a garbage token (`AddTagHelper`) followed by "Oklahoma, is Tulsa" (the city, not the capital); at M=5 the first token is a comma and the continuation corrects to "Oklahoma City, and Tulsa is Oklahoma".

(2) Field interference. A pair that hits on the intermediate+answer subset but not on the all-three-fields variant. Rescuable by field-additivity search (the field-composition effect of §**??**).

(3) Feature-interaction noise. The label is correct but at high M distributed noise masks the target. Example: `indiana`→`arkansas` produces `tonode` at M=20 and correctly produces `Little Rock` at M=5. A subset of these are rescued by M reduction; the remainder are irreducible.

(4) Signal collapse. The target logit collapses below a pair-specific M threshold. Example: `vermont_burlington`→`kansas_wichita` has vsMax 8.34 at M=20 but only 0.31 at M=5. For these pairs the useful M range is narrow and lies above the default; no value of M in $\{5, 7, 10, 20\}$ produces a hit.

(5) Feature specificity failure. The intervention correctly activates the answer concept but the wrong specific entity within the concept. Example: `vermont`→`kansas` generates "Kansas is Hutchinson" (wrong Kansas city), not "Topeka". Not fixable by M adjustment: the features that were grouped under the Kansas supernode appear to encode "Kansas cities" rather than "Topeka" specifically.

The coarse taxonomy is important for interpreting the aggregate recovery numbers. Modes (1)–(3) are rescued by adaptive M-search; mode (4) is occasionally rescued; mode (5) is not.

H. KL Divergence as a Steering Diagnostic

Linearity of KL in M. Within the $M \in [5, 20]$ range, the per-pair KL curve between the steered and unsteered distributions is close to linear: $R^2 > 0.93$ for 8 of the 10 USA pairs on which we estimated the linear fit. Table 6 lists the slope and intercept of the linear fit for each probed pair; the slope varies by a factor of $5\times$ across pairs, and is strongly predicted by the pair's ablate count (more source features ablated ⇒ steeper KL rise per unit M).

Pair (source → target)	Slope a	Intercept b	R^2	amp.	total
kansas → oklahoma	0.318	8.28	0.941	73	163
delaware → oklahoma	0.269	7.49	0.995	73	152
texas → oklahoma	0.218	9.34	0.986	73	138
florida → oklahoma	0.263	10.06	1.000	73	139
vermont → kansas	0.079	8.43	0.864	90	130
rhode_island → wisconsin	0.194	8.16	0.982	86	156
iowa → utah	0.245	6.99	0.935	93	253
indiana → arkansas	0.374	5.09	0.967	69	269
indiana → minnesota	0.317	6.16	0.946	82	282
hawaii → oklahoma	0.109	13.69	0.848	73	158

Table 6. Per-pair linear fits of KL(baseline ‖ steered) as a function of M, estimated from $M \in \{5, 10, 20\}$. `amp.` and `total` count the amplify and total features.

The slope–count correlations are $r(\text{slope}, \text{ablate_count}) = +0.68$ and $r(\text{intercept}, \text{ablate_count}) = -0.63$: high-ablate-count pairs have steeper KL rises and lower starting KL, so they are the ones most sensitive to M and most likely to benefit from M-reduction.

KL ≥ 12 as a hit veto. Across 30 USA observations, no hit occurs at KL ≥ 12. As a binary classifier on "hit iff KL < 12", recall is $7/7 = 100\%$ and precision is $7/19 = 37\%$; the 12 false positives break down into three feature-specificity failures, three evaluator gaps (correct output miscounted due to tokenization differences such as "St. Paul" vs "Saint Paul"), three signal-collapse pairs, and three near-threshold misses. KL therefore functions as a reliable veto: it reliably rules out

failure modes where the output distribution is too disrupted for any target to survive.

Use in adaptive search. The adaptive harness uses this in four steps: (i) run at M=20 and measure KL at position 0; (ii) if KL ≥ 12, fit the linear KL(M) from one additional probe and compute a pair-specific $M_{\text{crit}} = (12 - b)/a$; (iii) re-run at $M = \lfloor 0.8 \cdot M_{\text{crit}} \rfloor$ with a safety margin; (iv) if the intercept b already exceeds 12 (as for `hawaii` $\rightarrow$ `oklahoma`, where b=13.69), flag the pair as intrinsically disruptive and do not probe further.

A note on target recovery as a signal. An earlier version of the analysis promoted *target recovery rate*—whether the target's logit exceeds its own unsteered baseline at any trajectory position—as a primary label-evidence metric, on the basis of a 92% (labeled) vs 29% (random) gap in USA regime C. A deliberate replication attempt on Books collapsed this gap to 92% vs 89% in the same regime, because the Books model's smaller answer set and distinctive author signatures make random recovery mechanically easy. The lesson, which we took seriously when finalizing the main-paper metrics, is that high-discrimination binary flags may reflect a single domain's structure; *continuous* magnitude metrics (max excess over baseline, vsMax) are more portable. We therefore use Target Recovery only as one of three supporting signals in the within-regime-C analysis of Section 4.1.

I. Logit-Shift Regime Taxonomy

I.1. Definitions (position 0)

We classify each swap by what happens to the target and source logits at position 0, relative to their unsteered baselines. Four regimes matter for interpreting results: **A** = target up, source down, flip (clean redirection); **C** = both down, flip (differential disruption—target recovers more than source); **D** = both down, no flip (generic disruption); **E** = target flat, source down (pure suppression, no target promotion).

I.2. Prevalence at the default and at the best variant

Table 7 compares regime prevalence across three conditions: the per-pair best field-additivity variant, the full labeled (M=20, all three fields), and the matched-random control under the symmetric per-pair best-of-(replicate $\times$ adaptive-M) rule of Table 1. The clean-redirection regime A is the one we ultimately want to maximize; the generic-disruption regime D is the one matched-random concentrates in.

Domain	Regime A (%)			Regime D (%)		
	best FA	full lab	rand.+M-srch	best FA	full lab	rand.+M-srch
USA	34.9	8.9	15.3	9.1	19.4	38.7
Books	62.1	38.8	26.7	3.3	3.3	34.4
Products	62.1	56.8	28.8	2.3	2.3	18.9
Paintings	47.8	17.8	16.7	2.2	6.7	36.7

Table 7. Regime A (clean redirection) and Regime D (generic disruption) prevalence under the per-pair best field-additivity variant, full labeled, and the symmetric per-pair best-of-(replicate $\times$ adaptive-M) matched-random condition. The best variant dramatically amplifies regime A and suppresses regime D compared to full labeled; the matched-random control concentrates in regime D (35–39% in USA, Books, and Paintings) even with the symmetric M-search harness.

Three patterns stand out. The best FA variant pushes many more cases into regime A (clean redirection) than the full labeled variant does (USA: $8.9 \rightarrow 34.9\%$; Books: $38.8 \rightarrow 62.1\%$); removing input-field features eliminates the generic disruption that was pushing cases into regime C/D.

Within regime A, hit rates also improve: USA goes from 44% (full labeled) to 80% (best variant); Books from 4.3% to 54.4%. The best variant is not merely redistributing pairs across regimes—it is also raising the quality of each regime.

Regime D (generic disruption, no flip) nearly vanishes under the labeled condition ($19.4 \rightarrow 9.1\%$ in USA), while the symmetric matched-random+M-srch control concentrates there (35–39% in USA, Books, and Paintings; 19% in Products). This is why vsMax separates labeled from random even when both conditions produce high suppression rates and even when the random side is given the same adaptive-M harness.

I.3. Within-regime-C signals (USA)

Regime C is the largest and most ambiguous regime in the labeled condition: both logits drop, but the target less so. The richest within-regime-C signals are given in Table 8, which is the data behind the within-regime-C result reported in Section 4.1.

Signal	Labeled	Random
Target-recovery rate (%)	92.2	29.3
Sustained dominance (`tgt_win_pct`)	0.673	0.319
Mean vsMax	+2.33	−0.10
Hit% (conditional on regime C)	24.8	1.8

Table 8. Within-regime-C signals, USA. Labeled features cause differential disruption in which the target recovers above its own unsteered baseline while the source stays suppressed; matched-random does not.

The target-recovery signal (whether the target's logit at any point exceeds its own unsteered baseline) is strong in USA regime C but does not generalise: in Books regime C the labeled vs. random gap collapses to near zero. Books has a small answer space and sharply distinct entity signatures, so any large enough perturbation mechanically pushes the target above its baseline. Target-recovery therefore serves as one of three supporting within-regime signals.

J. Natural Cluster Geometry of CPAS Metrics

The decision tree (Table 3) partitions features into four functional roles using six axis-aligned threshold cuts. A natural concern is whether these four labels reflect dense regions of the feature metric space or impose arbitrary cuts. We tested it by re-aggregating per-feature metrics across all four datasets ($\approx 8{,}000$ deduplicated features) and running unsupervised clustering with several methods.

Threshold sensitivity. Each of the four primary cuts was perturbed by $\pm 10\%$ on the manifest. Per-cut flip rates ranged from 0.05% to 1.5%, and within tight neighborhoods of each threshold the local flip rate stayed below 4%. The simplest "many points sit on arbitrary boundaries" reading is therefore inconsistent with the data.

Visual structure. We standardized the six metrics and projected them with t-SNE per dataset (Figure 23). Across all four domains the four rule labels occupy spatially coherent regions: RELATIONSHIP forms a single low-sparsity arc, SAY-X a tight late-layer cluster, and SEM-CONC occupies the layer-shallow midband. The SEM-DICT bucket consistently splits into two visually disconnected regions, an early hint that the rule's semantic-fallback branch sweeps up two genuinely different feature populations.

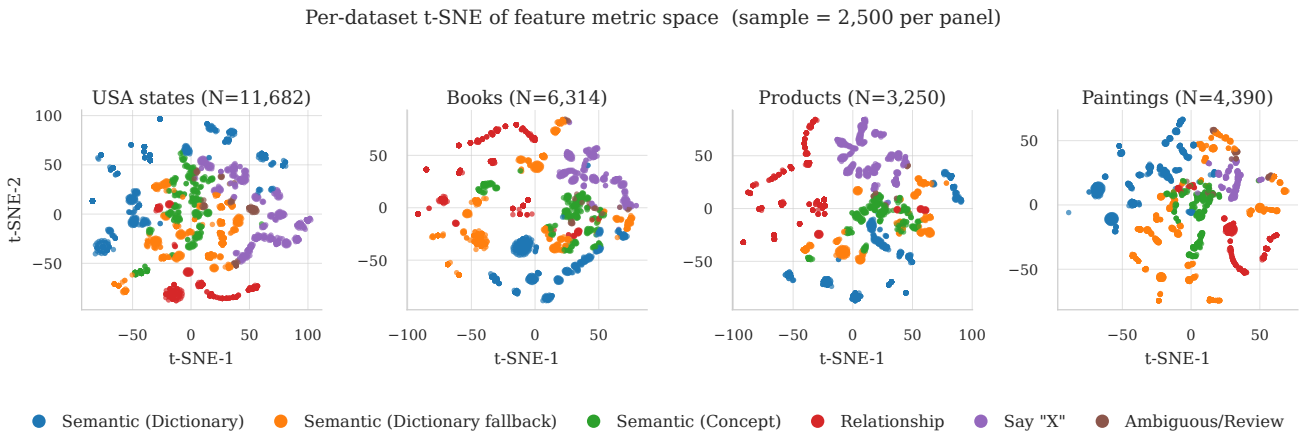

Figure 23. Per-dataset t-SNE of the standardized 6-metric CPAS vector. Points are coloured by rule label. RELATIONSHIP and SAY-X form tight clusters; SEM-DICT is multi-modal, an early hint of the bimodality discussed below.

Natural number of components. Three principled criteria for the natural number of clusters disagree with the rule's k=4. Density-based clustering (HDBSCAN, `min_cluster_size`= 150) settles at 13 components after excluding noise; Ward agglomerative silhouette plateaus at k=8; and Gaussian-mixture BIC monotonically prefers k=10 (full covariance) with a BIC drop of 1.13×10^5 nats from k=4, far beyond any plausible noise level. Figure 24 shows the BIC curve and a per-dataset UMAP overlay of the BIC-preferred clusters.

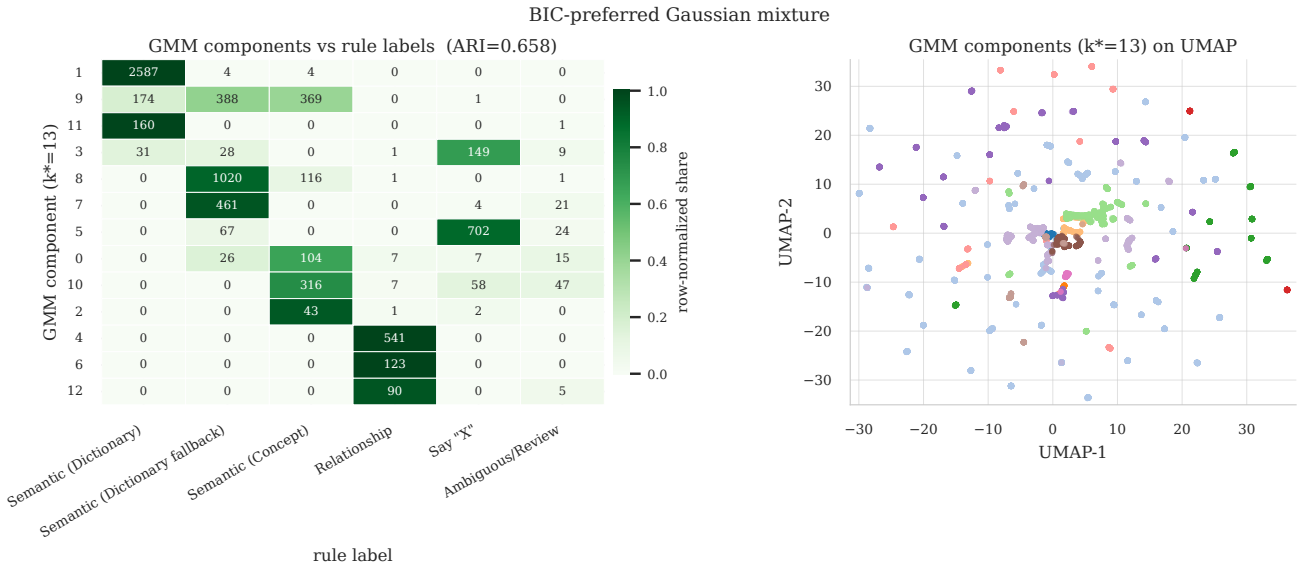

Figure 24. Left: Gaussian-mixture BIC as a function of the number of components for full and diagonal covariance, on the deduplicated manifest. Both curves prefer $k \in \{10, 12\}$. Right: per-dataset UMAP coloured by the BIC-preferred GMM partition; the rule labels (overlaid markers) are consistent with the natural geometry but coarser than it.

Coarse-grained recovery. Despite the k>4 preference, the rule labels are an Adjusted-Rand-Index–coherent coarse-graining of the natural geometry: KMeans-k=4 yields ARI ≈ 0.51 against the rules, and the BIC-preferred $k\approx$10 yields ARI ≈ 0.62. Per-dataset transfer of the rule geometry varies (Table 9): the rule partition transfers cleanly to Products and Books, weakly to Paintings, and only partially to USA, consistent with USA's heavier mix of SEM-DICT-fallback features and Paintings' smaller, lower-quality circuit population (Section Q).

Dataset	KMeans k=4 ARI	GMM diag k=4 ARI
USA	0.35	0.35
Books	0.50	0.51
Products	0.60	0.62
Paintings	0.24	0.25

Table 9. Per-dataset Adjusted Rand Index between rule labels and unsupervised k=4 partitions of the standardized 6-metric vector.

The four-role partition is best understood as a deliberately coarse-grained labeling of a finer natural manifold. The rule labels are spatially coherent, threshold-stable, and ARI-coherent at k=4, but the data has more structure than four classes: the SEM-DICT fallback in particular is bimodal and is a natural target for a refined taxonomy in future work (Section 6).

K. Human-curated Dallas Case Study

The Dallas/Austin attribution graph is the only target circuit in our datasets that has a publicly-released human-curated subgraph (the Neuronpedia graph `gemma-fact-dallas-austin` by user `mh2parker`). We use this single-circuit asymmetry to run a focused case study: hold the target circuit fixed at `texas_dallas`, run every non-Dallas USA state (49 sources) as the source side, and vary the target-side feature *bag* across four conditions: human curation, our auto pipeline, an unlabeled top-K-by-influence control, and a label-shuffled floor control. The headline comparison is ours-vs-human; the top-K family is a control that asks whether pure influence ranking can substitute for either, and the shuffled-labels condition is a sanity floor.

K.1. Setup

One target circuit (`texas_dallas`, the Dallas/Austin prompt with 1,182 features at our cumulative-influence threshold), 49 non-Dallas USA source states (every entity from `full_swap_human_dallas.yml`), Dallas always the target. Each *condition* is a different target-side bag for the Dallas circuit:

- **ours**: full canonical auto Dallas grouping (the eight concept-aligned supernodes of Figure 3, 458 classified features). Per-pair best of the 7-variant field-additivity sweep with adaptive M-search.
- **human**: the 22 features pinned in the Anthropic Neuronpedia public graph `gemma-fact-dallas-austin` (5 named supernodes; manifest at `output/usa_states_fact_batch/_swap_conditions/human_dallas/texas_dallas/manifest.json`). Same field-additivity + M-search protocol.
- **top-K (control, single-bag)**: top-K Dallas features by max `node_influence` from the canonical 1,182-feature universe, with $K \in \{10, 21, 100, 200\}$, run as a flat unlabelled bag — *no* field-additivity, *no* concept-field semantics, *no* `supernode_name` filter, *no* per-variant subsetting. Adaptive M-search enabled. This is the fair influence-only control: an earlier version of the experiment ran the top-K family with `control.mode: additivity` and inherited `auto`'s supernode labels onto the top-K rows, which let the field-additivity matcher subset the bag per-variant; that variant of the protocol is no longer used (see comparison below).
- **shuffled-labels (floor control)**: same 22 features as human, supernode labels permuted; 7 field-additivity variants + M-search.

The source side (the 49 source states' ablations) is the canonical auto Dallas-target grouping in every condition — on average 1,253 features per state — so the only quantity that varies across rows is the *target bag*. The source-side bag is not held to any matching constraint, just held *constant* across conditions (asymmetric design). Hit metric is as defined in §L; for this target "Austin" is a single six-letter word, so rule (iii) detects the same hits as a first-subword check in nearly all cases. Source hit-rate is the fraction of the 49 source states for which at least one cell hits.

K.2. Headline result: ours $\approx$ human $\gg$ top-K, shuffled = 0

Table 10 gives the per-condition source coverage and per-cell efficiency. Both labeled methods (ours, human) clear 78% of source states; every fair single-bag top-K saturates below 13%; shuffled-labels is at 0%. *The labeled methods reach the same source-coverage ceiling, but at very different per-cell costs*: human uses 4 amplified features per call on average and median M_{tuned}=2.0 (collapsed near default), our auto pipeline uses 96.5 features per call and median M_{tuned}=4.1 (consistent across pairs).

Condition	Sources hit / 49	Cell hit-rate	Mean amp. feat / call	M_{tuned} median	cum infl / call
ours	40 / 49 (82%)	30.1%	96.5	4.08	0.067
human	38 / 49 (78%)	26.2%	4.0	2.00	0.014
top-100	6 / 49 (12%)	12.2%	100.0	2.40	0.146
top-21	3 / 49 (6%)	6.1%	21.0	6.93	0.073
top-10	3 / 49 (6%)	6.1%	10.0	20.0†	0.051
top-200	1 / 49 (2%)	2.0%	200.0	4.08	0.197
shuffled-labels	0 / 49 (0%)	0.0%	3.4	—	0.004

Table 10. Per-condition source coverage on the 49-state Dallas-target swap (target = `texas_dallas`; 49 non-Dallas USA states as sources). *Cum infl / call* is the mean per-pair cumulative `node_influence` consumed by the target-side amplification bag — the x-axis of Figure 4. Top-K rows are the fair single-bag re-run; ours/human/shuffled use the standard 7-variant field-additivity sweep. †top-10 hits all occur at the default M=20; no M-search refinement was needed.

Figure 4 plots successful swaps (%) (y) against mean cumulative influence per swap (x). The three green top-K markers (top-21, top-100, top-200; top-10 omitted from the figure as redundant with top-21) sit near the floor; `top-21` sits just above ours (1.09$\times$ cumulative influence), `top-100` at 2.17$\times$, and `top-200` at 2.93$\times$. None clears the $\sim 12\%$ successful-swaps ceiling. The labeled markers (red, blue) sit at $\sim 80\%$. The non-monotone K=100 $\rightarrow$ K=200 drop ($12\% \rightarrow 2\%$) is consistent with high-K disruption (every K=200 hit needed M=4.1 via M-search; the default M=20 never hit).

K.3. Per-source heatmap: methods are non-redundant

The source-coverage ceiling hides genuine method-specific wins. Of the 49 sources:

- $22/49$ are universally easy (every method except shuffled hits).
- $1/49$ is universally hard (`idaho_idaho_falls`; no method hits).
- $26/49$ form a disagreement set in which ours and human partially complement each other, and a small number of sources are accessible only to top-K:
  - *human-only*: `missouri_kansas_city`, `north_dakota_fargo`. The 22-feature human curation captures something neither ours nor any of the four top-K controls reaches.
  - *ours-only*: `colorado_colorado_springs`. The label-driven supernode composition captures something even top-200 misses.
  - *top-200-only*: `oklahoma_tulsa` (in the previous unfair phase3v3 protocol). A pure-volume influence win that disappears in the fair single-bag re-run; the source is hit only when the field-additivity boost is also active.

The two human-only sources are noteworthy: a 22-feature curated bag finds a redirection that an automated pipeline with ~ 100 features and influence ranking miss. Our auto pipeline reciprocally finds at least one source neither curation nor any single-bag top-K reaches.

K.4. Naturalness: M_{tuned} distributions

Across all hits, M_{tuned} medians sit at 2.0 (human), 4.1 (ours), and 2.4 (top-100); only `top-21` required $M{\sim}6.9$ on its three M-search hits. Human is the most *natural*: when it hits, it hits at small M, and many of its hits already exist at the default. Ours is the most *consistent*: tight M_{tuned} IQR, no high-M outliers above $\sim$7, and all 40 hits cluster within a 4-point M window. The high-K controls drift toward smaller M (top-100 at $M{=}2.4$, top-200 at $M{=}4.1$) because the default $M{=}20$ overshoots when K is large — M-search is doing all the work for these conditions.

K.5. Top-K saturation as a control: precise protocol

The fair top-K family is included to test whether pure influence-only ranking can substitute for either the human curation or our pipeline. Each top-K condition runs the following protocol:

- Target bag: top-K Dallas features by max `node_influence` over the canonical Dallas grouping universe. Source bag: canonical auto source's full grouping ($\sim$ 1,253 features per state on average, no filter, no field-additivity).
- No `concept_fields`, no `supernode_name` filtering, no per-variant subsetting — one intervention per pair (`control.mode: single_bag_grouping`).
- Adaptive M-search: Phase 1 coarse $M \in \{0.1, 0.25, 0.6, 1.5, 4.0, 10.0, 20.0\}$; Phase 2 KL-binary refinement; same parameters as `fullscale_usa_labeled_msearch.yml`.
- Hit metric: as defined in §L ("Austin" as a six-letter word matches via rule (iii)). Source hit-rate: fraction of the 49 source states with at least one hit.
- Configs: `phase4_topk_{10,21,100,200}_dallas_singlebag.yml`; control: `scripts/experiments/batch/pipeline/controls/single_bag_grouping.py`; launcher: `tools/launch_phase4_topk_singlebag.sh`; aggregator: `tools/phase4_topk_singlebag_aggregate.py`.

The top-K saturation was originally measured (Phase B v3, May 5) with `control.mode: additivity`, which transferred `auto`'s supernode labels onto the top-K rows and let the field-additivity matcher subset the bag per-variant. That earlier version reported top-21 / top-100 / top-200 source coverage of $63\%/69\%/82\%$, suggesting top-K saturated near `ours`. The fair re-run with the same target bags but no field-additivity gives $6\%/12\%/2\%$. The 57–80 pp gap was the field-additivity boost masquerading as influence-ranking quality. The result reported in this section is the fair version; the cross-domain influence-matched analogue (§M) reaches the same conclusion on four in-scope domains under a per-pair budget-match.

K.6. Threats to validity

Single target circuit (Dallas) and single domain (USA states); the Dallas top-K also happens to be unusually concept-pure ($\sim 90\%$ of top-21 features fall in the eight `auto`-labeled supernodes), so the field-additivity boost reported above may be larger here than on a domain with messier labels. The cross-domain influence-matched experiment (§M) runs the analogous test on four domains under a stricter per-pair budget-match.

The source-side ablation uses canonical auto's labels for every condition, so the auto pipeline is "running" on every row as ablation, even when the target side is human/top-K/shuffled. This is the standard convention for this experiment (the only Dallas-target curated graph is the human one, so we cannot run a symmetric human-on-both-sides bag), but it means we cannot disentangle "labels matter for ablation" from "labels matter for amplification" in this specific case study. §M runs both sides influence-matched and reaches the same direction of result.

L. Per-Domain Full-Scale Labeled vs Random

Symmetry of the per-pair best-of construction. The labeled FA+M-srch column of Table 1 reports the per-pair best across $\{$FIELD-ADDITIVITY VARIANTS$\} \times \{$default, M-tuned$\}$. To make the matched-random control comparable we apply the *same* per-pair best-of rule to its three replicates: for every labeled pair we take the best across 3 replicates $\times$ $\{$default, M-tuned$\}$ random candidates under the same lexicographic score (hit, then $-$rank, then vsMax). This means both columns get one row per labeled pair, both columns see the same outer adaptive-M harness, and both columns are scored under the unified hit rule below. The per-replicate random row at the default M is kept in Table 11 as a baseline for the rescue accounting in Table 5, but it is not the right object to compare against the labeled FA+M-srch number; that comparison happens between Table 1's *Rand. +M-srch* column and the matching per-pair-best random+M-srch row of Table 11.

What counts as a hit. Throughout the paper a swap is a *hit* when the target answer is detected in the model's steered output by any of three simple text checks, applied in order: (i) the full target string appears in the output (with punctuation and hyphens normalised, so "St. Paul" and "Saint Paul" match); (ii) the first emitted subword is a substring of the target answer (so emitting "Mark" counts toward "Mark Zuckerberg"); (iii) any content word of the target of length ≥ 3 characters appears as a whole word in the output (so "Harper" anywhere in the continuation counts toward "Nelle Harper Lee"). A small per-domain blacklist removes generic words that would match too easily (e.g. "city" on the USA panel). We arrived at this rule by hand-checking outputs that a stricter first-subword equality rule scored as misses: most were correct redirections that emitted the answer one token late, with a different tokenisation, or surrounded by punctuation; the three checks above recover those cases while the word-boundary requirement and the length-3 minimum keep stop-words and one-letter tokens out. Every comparison condition (matched-random, top-K influence-matched, shuffled labels) is scored under the same rule, so the labeled–control gap is what carries the claims. A future revision can swap the three text checks for an LLM-as-judge that grades the steered output against the target answer with full lexical and semantic flexibility; the three checks here are a deterministic placeholder for that judge and the rest of the paper does not otherwise depend on the choice.

Table 11 gives the full per-domain comparison at the standard defaults (all-three-fields, $M_{\text{ablate}} = -2$, $M_{\text{amplify}} = 20$, attention not frozen). Each domain is reported with the labeled intervention, the matched-random-control replicate set at the default M (three replicates per labeled pair, so per-replicate random N is $3\times$ the labeled N), and the per-pair best of the matched-random set under the symmetric adaptive M-search (one row per labeled pair, taking the best across the three replicates and their M-tuned outputs).

The rightmost four columns tell the same qualitative story from different angles. The labeled intervention promotes the target to a median rank inside the top 70 in every domain, while the matched-random controls leave it near rank 200–550. `Flip%` (how often the target's logit overtakes the source at any trajectory position) is above 96% in every labeled condition, which is why the operational test relies on vsMax rather than on the flip indicator alone.

Labeled–random vsMax gap by domain. Against the per-replicate matched-random control at the default M, Books and USA show the largest labeled–random vsMax (logit margin over the next-best answer) gap ($+7.4$ and $+5.2$ respectively); Products is moderate ($+3.3$) and Paintings weak ($+1.4$). The same ordering holds when the random side is replaced by the symmetric per-pair best-of-(replicate $\times$ adaptive-M) random+M-srch row of Table 11: Books $+6.5$, USA $+4.1$, Products $+2.3$, Paintings $+0.2$. This ordering matches the operational-usefulness verdict, which is one reason vsMax is a primary metric.

Domain	Cond.	N	Hit%	Supp%	vsMax	RkGrp	MedRk	Flip%
USA	labeled	2,450	31.4	92.8	+2.86	1.72	5	98.2
	random	7,350	4.6	83.4	−2.31	9.00	566	69.2
	random+M-srch	2,450	0.7	84.1	−1.23	6.32	148	75.5
Books	labeled	90	6.7	64.4	+6.70	1.03	17	96.7
	random	270	0.0	88.1	−0.73	2.43	283	79.6
	random+M-srch	90	0.0	85.6	+0.19	2.18	266	83.3
Products	labeled	132	23.5	64.4	+3.46	1.20	26	97.0
	random	396	0.5	87.4	+0.14	2.25	354	75.3
	random+M-srch	132	7.6	84.8	+1.17	1.79	262	71.2
Paintings	labeled	90	4.4	37.8	+1.50	1.31	70	97.8
	random	270	0.0	74.4	+0.12	1.96	196	88.9
	random+M-srch	90	1.1	68.9	+1.27	1.42	187	90.0

Table 11. Full-scale labeled vs. matched-random comparison. The first two sub-rows per domain use the all-three-fields default at M_{amplify}=20 (labeled and matched-random at three replicates per pair). The third sub-row reports the matched-random control under the same per-pair best-of-(replicate × adaptive-M) construction as the labeled FA+M-srch column of Table 1, so the comparison is symmetric: every labeled pair is matched to exactly one random-control row, taken as the best of 3 × {default, M-tuned} candidates by the same lexicographic score (hit, then -rank, then vsMax). The labeled–random vsMax gap is the central signal of operational usefulness (§4.1); applying M-search to the random side narrows the per-replicate Hit% gap by at most 7 pp (Products) and leaves the labeled–random ordering unchanged in every domain.

M. Per-Pair Influence-Matched Top-K Baseline

The matched-random control fixes the feature count and per-layer histogram but not the total graph influence the intervention uses. A labeled pair whose features happen to land on high-influence nodes has a structural advantage unrelated to concept alignment. This section adds a second control—TOP-K INFLUENCE-MATCHED—that instead holds the per-side cumulative `node_influence` budget fixed at the labeled best-of value.

M.1. Construction

For every pair (e_A, e_B) in the four in-scope domains, let S^{lab} be the source-side ablated feature set and T^{lab} the target-side amplified feature set selected by the per-pair best-of-(field-additivity × {default, M-search}) labeled run. Define

$$B^{\text{src}}(e_A, e_B) = \sum_{f \in S^{\text{lab}}} \text{influence}_{e_A}(f), \qquad B^{\text{tgt}}(e_A, e_B) = \sum_{f \in T^{\text{lab}}} \text{influence}_{e_B}(f),$$

where $\text{influence}_e(\cdot)$ is the per-feature `node_influence` from e's attribution graph, deduped by max over (layer, id). The matched control's bag is the smallest top-K prefix of each entity's own influence-ranked grouping universe whose cumulative influence reaches the corresponding budget:

$$K^{\text{src}} = \min\left\{ K : \sum_{f \in \text{top-}K(e_A)} \text{influence}_{e_A}(f) \geq B^{\text{src}} \right\},$$

and analogously K^{tgt} on the target side. The intervention is a single bag, with no field-additivity sub-selection, with $M_{\text{ablate}} = -2$, M_{amplify}=20, and the same outer adaptive-M sweep enabled. The grouping universe used for the top-K is the same set of steerable features the labeled pipeline classifies; scaffold/error rows ($f = -1$) and embedding/logit-only rows (layer<0) are excluded by construction.

M.2. Result

Table 12 reports the head-to-head Hit% comparison and a paired McNemar test (a paired sign test on the hit/miss contingency) on the contingency (b=labeled-only wins, c=top-K-only wins). Labeled features beat the influence-matched top-K baseline in every domain where the test has power: paired p-values are $\ll 10^{-30}$ (USA), 2.7×10^{-20} (Books), 1.1×10^{-13} (Products), and 0.23 (Paintings, N=56, underpowered). The companion 4-panel figure (**??**) plots each domain's Hit% gap against the per-condition mean number of amplified features, confirming that the labeled bag uses substantially *more* features than top-K at the same per-pair influence budget, while still scoring higher on the redirection metric.

Table 12. Per-pair influence-matched top-K baseline vs. labeled best-of, four domains. For every swap pair the top-K baseline selects the smallest prefix of features ranked by per-entity graph influence whose cumulative influence matches the labeled best-of budget; both conditions share the same outer M-search sweep. Hit% is scored under the unified rule (§L) on the demo cross-run intersection. b = pairs where only labeled hits; c = pairs where only top-K hits (contingency cells for the paired McNemar test). Paintings N=56 because the influence-matched run covers only 56 of the 90 demo-intersection pairs; that subset is underpowered. Full per-pair join in `output/research/topk_im_pairs_<domain>.csv`.

		Hit%			McNemar		
Domain	N	top-K	Lab.	Δ pp	b	c	p
USA	2,450	4.2	72.8	+68.6	1,686	5	$\ll 10^{-30}$
Books	90	4.4	77.8	+73.3	66	0	2.7×10^{-20}
Products	132	1.1	50.0	+48.9	44	0	1.1×10^{-13}
Paintings	56	7.1	16.1	+8.9	8	3	0.23

The labeled bag uses substantially more features than top-K at the same per-pair influence budget. Two readings are consistent with this: (i) a substantial fraction of the labeled features are low-influence "scaffold" nodes that nevertheless gate the answer circuit; (ii) the highest-influence nodes are the *loudest* features at the first token, but not the most informative for redirecting the model to a specific target, consistent with §G's top-k-rescue null on Products. We do not separate (i) and (ii) within this experiment.

N. Suppression Is Easy; Steering Is Hard

The most common outcome of a labeled intervention is not a successful redirect but a *suppression without retargeting*: the source answer disappears from the output, but the target does not appear either. This section documents that asymmetry and explains why it is structurally expected.

N.1. The gap between suppressing and steering

Suppression and targeting are not symmetric operations. Ablating source features at $M_{\text{ablate}}=-2$ multiplies each ablated feature's decoder contribution by a negative scalar, driving it to near zero regardless of what that feature encodes semantically; any collection of features that covers the source-entity circuit is sufficient to kill the source token. Amplifying target features only promotes the target if those features actually encode the answer—they must carry the correct answer's identity in their activation. The result is that suppression requires coverage; targeting requires specificity.

N.2. Suppression rates vs. targeting rates

Table 13 compares source-suppression rates with target-detection rates for the labeled intervention and the matched-random control across all four domains.

	Labeled		Matched-random		Rand. +M-srch	
Domain	Supp%	Hit%	Supp%	Hit%	Supp%	Hit%
USA	92.8	31.4	83.4	4.6	84.1	0.7
Books	64.4	6.7	88.1	0.0	85.6	0.0
Products	64.4	23.5	87.4	0.5	84.8	7.6
Paintings	37.8	4.4	74.4	0.0	68.9	1.1

Table 13. Source-suppression rate (Supp%) and target-detection rate (Hit%) for labeled, matched-random (per replicate at the default M), and the symmetric per-pair best-of-(replicate $\times$ adaptive-M) matched-random condition. Random interventions suppress the source at rates comparable to labeled but land almost no hits, and giving the random side the same outer adaptive-M harness as the labeled FA+M-srch column does not close that gap (only Products picks up a few hits, $0.5 \rightarrow 7.6\%$, against 41.7% for FA+M-srch on the same pair set).

The labeled and matched-random conditions suppress the source at broadly similar rates—within 25 pp in three of four domains—while their Hit% differs by a factor of 5–20$\times$. A random bag of features matched in count and layer distribution achieves 74–88% suppression in most domains while hitting essentially zero targets. This confirms the asymmetry: the source disappears because the circuit is disrupted; the target appears only when features that encode the specific answer are

present.

N.3. The suppression-only zone

Most suppressed pairs land in what we call the *suppression-only zone*: the source answer is absent but the target is not detected, so the steered output fills the slot with generic or incoherent text. In the labeled condition at the all-fields default, the suppression-only zone accounts for 63.3% (USA), 66.2% (Books), 44.3% (Products), and 36.3% (Paintings) of all pairs—the single largest outcome bucket in every domain. These correspond to regime E (source logit drops, target stays flat) and the subset of regime C/D pairs that never produce a detectable target token.

The suppression-only zone shrinks substantially when the best field-additivity variant replaces the all-fields default (§F): removing input-field features that compete with the answer circuit moves many pairs from suppressed-only to a clean hit. This is the mechanism behind the less-is-more effect reported in the main text.

N.4. Why suppression is structurally cheaper

Three reasons explain the asymmetry:

(a) Ablation is semantics-blind. The $-2\times$ multiplier drives a feature's contribution toward zero independently of what that feature encodes. Any bag large enough to cover the source-entity circuit suppresses the source.

(b) Random bags are sufficient for suppression. A matched-random bag—same feature count and per-layer histogram as the labeled bag, but drawn from outside concept-aligned supernodes—suppresses the source at nearly the same rate as labeled features (Supp% gap ≤ 25 pp in three of four domains). This is why Supp% is a poor discriminator between conditions; it is not what separates labeled from random.

(c) Targeting distributes probability mass over hundreds of answers. Suppressing the source spreads probability mass across the model's full vocabulary. To install the target answer, the amplified features must collectively shift the target logit above all competitors—a much harder coordination problem than eliminating the incumbent.

N.5. Implication for evaluation

Because suppression is almost universal under both labeled and random conditions, Supp% does not distinguish between a conceptually-driven intervention and a generic disruption. Hit% and vsMax (the logit margin over the next-best answer) are the metrics that track this distinction, and are therefore the primary evaluation signals throughout the paper.

O. Feature Stability Across Prompt Phrasings

The features assigned to an entity should reflect the concept itself across different prompts. We test this directly: for each entity we generate five differently-worded prompts that all call for the same answer, extract the attribution graph for each, and ask whether the same features appear consistently across all five. We ran this analysis on $1{,}592$ entity pairs within the same domain across all four domains (97 entities), using $5{,}000$ bootstrap resamples for confidence intervals and $2{,}000$ permutations per number of prompts per entity for significance. The short answer is yes: feature sets overlap well above chance in every domain, and the overlap is especially strong in early layers.

How much do feature sets overlap across phrasings? Table 14 reports the mean Jaccard overlap (fraction of features shared out of all features seen in either graph) between per-entity feature sets across prompt variants. All four domains have overlap significantly above chance ($p < 0.001$) at every number of prompts per entity tested.

Domain	Jaccard	95% CI	N pairs
USA	0.465	$[0.462, 0.468]$	1,225
Books	0.308	$[0.302, 0.315]$	210
Products	0.364	$[0.356, 0.374]$	91
Paintings	0.286	$[0.279, 0.292]$	66

Table 14. Mean within-domain Jaccard overlap between per-entity feature sets extracted from different prompt phrasings, with bootstrap 95% CIs.

Are individual feature activations consistent across phrasings? Each feature's activation pattern is consistent across prompt variants in every domain (stability > 0.90); peak-token agreement is 85–98% and peak-type (functional vs. semantic) agreement is 93–99%.

Early layers share a backbone; late layers are entity-specific. Table 15 reports Jaccard overlap in early, middle, and late layer bins. The early-to-late ratio is $1.4\times$–$3.0\times$ across domains, confirming that early layers host structural primitives reused across all entities, while late layers host entity-specific features.

Domain	Early	Middle	Late	Early/Late ratio
USA	0.543	0.440	0.293	$1.85\times$
Books	0.347	0.340	0.184	$1.89\times$
Products	0.496	0.308	0.164	$3.02\times$
Paintings	0.302	0.311	0.212	$1.43\times$

Table 15. Layer-binned within-domain Jaccard overlap and early/late ratio.

Do shared features play the same role in both entity graphs? A complementary view: for each feature that appears in two entity graphs, is it assigned to the *same* supernode (*scaffold*), *regrouped* into a different supernode (e.g. `Say(Austin)` in the Dallas graph vs `Say(Sacramento)` in the Oakland graph), or is the assignment *inconsistent* (same feature, semantically unrelated supernodes)? Table 16 gives the breakdown.

Domain	Same (%)	Regrouped (%)	Inconsistent (%)
USA	76.6	16.2	7.3
Books	47.5	12.8	39.7
Products	65.2	16.5	18.4
Paintings	71.0	9.9	19.1

Table 16. Supernode consistency across entity graphs within each domain. "Same" is the scaffold; "Regrouped" captures entity-appropriate re-assignment (typically Say-X features pointing at the new answer); "Inconsistent" flags same-feature-different-role mismatches.

Caveats. We note three non-trivial caveats for this table. First, the original Dallas/Oakland pair that motivated the early analyses is a ∼93rd-percentile outlier within USA; population means are 10–15 pp lower than the single-pair numbers originally reported. Second, the Books "Inconsistent" rate (39.7%) likely reflects keyword-detection limits on literary names (the matcher's treatment of "Anna Karenina" and similar). Third, Paintings' weak/inverted layer gradient is unexplained and may be driven by the small-N and high-error-node-rate structural issues discussed in §Q.

P. Graph Scaffold Analysis

P.1. Definitions

We partition each cross-entity feature population into three groups. *Scaffold features* appear in both graphs with the *same* supernode assignment (usually structural primitives like copulas, prepositions, and task operators). *Regrouped features* appear in both graphs but with different supernode names (typically Say-X features that point at the new answer, e.g. `Say(Austin)` vs `Say(Sacramento)`). *Entity-only features* appear in one graph only. *Scaffold influence* is the fraction of the graph's total influence carried by scaffold features.

P.2. A worked example: Dallas vs Oakland

The reference pair that originally motivated the scaffold metric is Dallas (Texas capital problem) vs Oakland (California capital problem). Table 17 decomposes it.

Of the 29 regrouped features, 27 are entity-appropriate re-assignments: 15 `Say(Austin)` → `Say(Sacramento)`, 4 `Texas` → `California`, 3 `Dallas` → `Oakland`, 2 `Say(Texas)` → `Say(California)`, and 1 `Austin` → `Sacramento`. Only 4 of the 29 (roughly 2.7% of the shared-feature population) are genuinely inconsistent regroupings. This is the pattern we formalize at the population level in Table 16.

Population	N features	Dallas infl.	Oakland infl.
Scaffold (shared + same supernode)	119	50.6%	49.2%
Regrouped (shared + different supernode)	29	12.3%	13.7%
Entity-only	55/62	21.7%	19.5%

Table 17. Dallas/Oakland scaffold decomposition. Scaffold features account for roughly half the total graph influence in both graphs, and their influence is concentrated in early layers (0–5, where they dominate 75–100% of total influence).

P.3. Cross-domain scaffold gradient

At the domain level, scaffold influence orders the four domains by Hit% at the all-three-fields labeled baseline up to a single inversion (Paintings and Books swap, Spearman ρ=0.8, N=4).

Domain	Scaffold infl.	Shared%	Early scaffold	Late scaffold	Hit%
USA	0.530	63.7%	74.6%	16.3%	31.4
Products	0.422	52.1%	62.1%	10.3%	23.5
Paintings	0.359	44.8%	50.9%	9.2%	4.4
Books	0.253	48.9%	39.7%	4.3%	6.7

Table 18. Per-domain scaffold metrics. "Scaffold influence" is the fraction of total graph influence carried by scaffold features; "Shared%" is the fraction of features (by count) that are scaffold; "Early" and "Late" are the same quantity restricted to layers $\leq L_5$ and $\geq L_{16}$ respectively.

The late-layer gradient is the most striking element of this table: USA shares 16.3% of its late-layer influence as scaffold, while Books shares only 4.3%. The output-generation layers are where entity-specific features concentrate, and where structural compatibility matters most. This connects mechanistically to the less-is-more effect: interventions that include input-field features disrupt the fragile late-layer scaffold.

Q. Paintings as the Weakest of the Four Domains

Paintings sits at the bottom of the four domains with a full-labeled Hit% of 4.4 and a vsMax gap of $+1.38$ ("Weak"). Three structural factors contribute.

(a) The answer field is coarse. The answer is the painter's first name—`Claude`, `Pablo`, `Leonardo`—tokens with many non-painting associations in the general language model distribution. Amplifying features that peak on "Claude" in a Monet circuit therefore moves the logit toward a cluster of non-painting "Claude" meanings.

(b) The painter supernode subsumes the first-name supernode. "Monet" as a string contains much of "Claude" as a concept (the painter is the agent that the first name names), so the `painter` and `first_name` supernodes overlap by construction. This collapses the field-additivity structure: `painter+first_name` is not a very different intervention from `painter` alone.

(c) High error-node rates. Paintings has the highest error-node rate of any domain (17%; an error node is a node whose attribution could not be resolved during graph generation), so a substantial portion of the circuit is simply not visible to the grouping or the intervention. The maximum number of features a paintings intervention can operate on is therefore smaller than in the other domains, and the signal-to-noise ratio is correspondingly worse.

Concretely, $89/90$ pairs fail at the `painter+first_name` variant. The only hit (La Grande Jatte $\rightarrow$ Water Lilies) already had a baseline rank of 3 for "Claude", so the intervention only had to move the logit by two positions. We therefore treat Paintings as a Weak-strength supporting domain. The rank-perfect scaffold prediction (Table 18) is *consistent* with the Paintings Weak verdict.

R. Reproducibility Manifest

R.1. Code, data, demo (anonymized for review)

All artifacts are released anonymously for the review period. Repository: https://anonymous.4open.science/r/attribution-graph-probing-anon (commit hash and license in `README.md`). Interactive demo: https:

//anonymous.4open.science/r/attribution-graph-probing-demo-anon. License: GNU GPL v3. All URLs will be de-anonymized at camera ready.

R.2. Per-domain CLI (USA example)

```
# 1. Generate attribution graph (Neuronpedia API)
python scripts/00_neuronpedia_graph_generation.py \
  --model gemma-2-2b-it \
  --prompt "The capital of the state containing Dallas is" \
  --target " Austin" \
  --node-threshold 0.8 --edge-threshold 0.85 --max-nodes 5000 \
  --output_dir output/usa/dallas_austin/

# 2. Select features (interactive UI)
streamlit run eda/threshold_selection.py --tau 0.95

# 3. Probes + activations + grouping
python scripts/01_probe_prompts.py \
  --graph output/usa/dallas_austin/graph.json \
  --output_csv output/usa/dallas_austin/activations.csv
python scripts/02_node_grouping.py \
  --input output/usa/dallas_austin/activations.csv \
  --graph output/usa/dallas_austin/graph.json \
  --output output/usa/dallas_austin/grouped.csv

# 4. Run swap protocol with matched random control + adaptive M-search
python scripts/03_swap_full.py --domain usa \
  --m_ablate -2 --m_amplify 20 --freeze-attention false \
  --random-replicates 3 --random-seed 42 \
  --field-additivity true --m-search adaptive \
  --output_dir output/usa_states_batch/
```

R.3. Runtime profile

Per single-circuit pipeline (graph + activation + grouping + subgraph): 14–24 minutes on a single L4 GPU. Full per-domain swap sweeps: USA 17h, Books 1.6h, Products 0.9h, Paintings 0.6h on 8 H100 GPUs (matched-control replicates parallelized).

R.4. Artifact release manifest

For each of the four domains, the release contains: (a) attribution graphs in Neuronpedia JSON format; (b) the probe-prompt files; (c) activation matrices; (d) supernode groupings (pre- and post-stability filter); (e) swap dumps with per-pair Hit, vsMax, RkGrp, Sup, Flip, CtrlS, and KL trajectories; (f) field-additivity per-variant tables; (g) adaptive M-search logs (Phase 1 and Phase 2); (h) random-control sha256 seeds.

R.5. Determinism and resume

The pipeline is fully deterministic. Matched-random controls are sha256-seeded from `(run_seed, pair_id, replicate, mode)`; generation uses T=0.3, n=10, frequency penalty 2.0, seed 42, with no random initialization elsewhere in the harness. Per-feature checkpoints permit resumption from partial runs, which is how we recover from transient Neuronpedia API failures.

S. Glossary and Notation

CLT Cross-Layer Transcoder (Ameisen et al., 2025).

Influence The (signed) contribution of a feature node to the target output logit via linearized paths through the replacement model, as computed by the attribution-graph procedure of (Ameisen et al., 2025). Used both per-node (a feature's importance for the prediction) and aggregated over node sets (e.g. the share of total graph influence carried by a population of features).

Cumulative influence The running sum of per-node |influence| taken over features ranked by descending influence and normalized by the total. Step 1 of the pipeline retains the smallest feature set V_τ whose cumulative influence reaches τ (§B); typical reduction is from 600–5,000 raw features to 200–700.

Supernode A group of same-role same-name features merged into a single node of the concept-aligned subgraph (§3.7). The unit of analysis throughout the paper.

Subgraph (concept-aligned) The compressed circuit produced by the pipeline: the attribution graph restricted to the cumulative-influence selection V_τ and re-expressed with each grouped feature replaced by its supernode (Figure 3). Typically 30–50 named supernodes per circuit.

Completeness (Neuronpedia) Fraction of incoming edges to all nodes of the subgraph that originate from *grouped* features, weighted by influence on the output. Pipeline mean 0.89 over 82 entities across 4 domains (§**??**).

Replacement (Neuronpedia) Fraction of end-to-end influence from input tokens to output logits that flows through grouped features. Pipeline mean 0.69 (§**??**).

CPAS Cross-Prompt Activation Signature (Table 2).

τ Cumulative-influence threshold for feature selection (default 0.95).

$M_{\text{ablate}}, M_{\text{amplify}}$ Multipliers for source and target feature decoders (defaults -2, 20).

vsMax $\max_{t \le 10} \left(\ell_{e_B}(t) - \max_{e' \ne e_B} \ell_{e'}(t)\right)$, where $\ell_e(t)$ is the logit of entity e's first-subword token at trajectory position t.

Hit The target answer is detected in the model's steered output by any of three text checks (full-string match with punctuation normalised, first-subword substring, or any content word of the target of length ≥ 3 appearing as a whole word in the output). Defined and motivated in §L.

RkGrp Best target rank within the domain answer set, minimized over trajectory positions.

Target Recovery Binary indicator: the target's logit exceeds its own unsteered baseline at some trajectory position.

Regime A/C/D/E Position-0 logit-shift regimes (§I).

Scaffold Features that appear in two entity graphs with identical supernode assignment.

Scaffold influence Fraction of total graph influence carried by scaffold features.

Field-additivity (FA) The 7-variant ablation of single-field, two-field, and three-field intervention subsets.

Matched-random control Random feature set with the same feature count and per-layer histogram as the labeled intervention, drawn from *outside* every concept-aligned supernode in the domain so that the only property that varies between labeled and control is concept alignment (§3.8).

KL $\mathrm{KL}(\text{baseline} \parallel \text{steered})$ at position 0, unless otherwise stated.