Final Report: Exhausting Backup Circuits: Redundancy, Universality, and Necessity in Dense and Weight-Sparse Transformer Circuits

Anwen Hao
ah3973@columbia.edu
Columbia University

Rick Goldstein (Mentor)
rickgoldstein12@gmail.com

Supervised Program for Alignment Research — Spring 2026

Abstract

Mechanistic interpretability seeks to explain neural networks as sparse circuits, but two concerns threaten this program: pruned circuits may be *unfaithful*, and a model may hold many redundant "backup" circuits, so that ablating one removes no capability. We study both by iteratively pruning minimal task circuits from a 2-layer dense transformer (d=128) and a weight-sparse transformer (d=1024, 1.56% nonzero weights), forcibly excluding the most-shared node each round until no circuit reaches target loss. On a pronoun-gender task we find that backup circuits are finite and exhaustible; the discovered core is task-selective (ablating it breaks the task but spares an unrelated tense task) and generalizes to held-out names. A probabilistic "avalanche" model predicts the exclusion dynamics (in-sample correlation 0.98) and shows the terminal core is seed-dependent. Decomposing circuits into atoms with non-negative matrix factorization and peeling them to exhaustion, we find that weight-sparsity yields *deep, cheap, modular* redundancy over a small (∼15-node) necessary core, whereas the dense model has *finite, expensive* redundancy that collapses through a phase transition. Weight-sparse circuits are causally more universal.

Keywords mechanistic interpretability · circuits · weight sparsity · universality · backup circuits · AI safety

1 Introduction

Mechanistic interpretability aims to reverse-engineer neural networks into human-understandable *circuits*—small subgraphs of components that implement a behavior [9, 4, 11]. Recently, Gao et al. [6] showed that *weight-sparse* transformers, trained with almost all weights pinned to zero, admit unusually small and clean task circuits, suggesting a route to scalable interpretability. But Drori [3] demonstrated that such pruned circuits can be *unfaithful*: a circuit that reaches target loss may compute differently from the full model.

We highlight a second concern: *redundancy*. If a capability is implemented by many overlapping backup circuits, then (i) the minimal circuit recovered by pruning is merely one of many, complicating any faithfulness claim, and (ii) deleting "the circuit" does not remove the capability—directly relevant to unlearning, capability removal, and robustness. Backup behavior is known anecdotally (e.g. backup name-mover heads [11]), but its *extent* and *structure* have not been quantified.

We therefore ask: *How many distinct backup circuits implement a task, how is the redundancy structured, and does weight-sparsity make circuits more universal—fewer, cleaner, and more shared—as the interpretability case requires?* Our approach is to *exhaust* the backup circuits: repeatedly

prune a minimal circuit from many random seeds, forcibly exclude the most-universal node, and re-prune until the task can no longer be solved. Success means a faithful, quantitative characterization of redundancy and a causal test of the weight-sparse-interpretability thesis on matched dense and sparse models.

Our contributions are: (1) we show the pronoun-task core is finite, task-selective, and generalizing, and identify its mechanism; (2) we build and validate a probabilistic model of the exclusion dynamics, showing the terminal core is a seed-dependent random variable; and (3) using an NMF decomposition into atoms and an iterative "atom-peel," we obtain a *necessary-core / redundant-periphery* decomposition and show weight-sparsity buys deep cheap redundancy over a small clean core, while dense redundancy is finite and fails through a phase transition.

2 Related Work

Circuits and universality. The circuits program treats features and computations as reusable, often universal motifs [9, 4]; the indirect-object-identification (IOI) circuit [11] is the canonical worked example and first documented "backup" heads that take over when primary heads are ablated. Our pronoun-gender task is a minimal instance of entity–attribute binding [5].

Circuit discovery. Automated methods include ACDC [1] and attribution-based pruning such as EAP-IG [7]. We instead use the mask-based pruning of Gao et al. [6], which optimizes a node mask directly against a contrastive task loss.

Weight-sparse interpretability and faithfulness. Gao et al. [6] argue weight-sparse transformers have interpretable circuits; Drori [3] argues they can be interpretable yet unfaithful, using circuit size and degenerate (uniform) attention as warning signs. We build directly on these models and reframe the debate around redundancy, evaluating faithfulness functionally rather than by circuit size.

Representations. We briefly considered non-linear "onion" representations [2] as an alternative lens before focusing on circuit redundancy. Our decomposition into atoms uses non-negative matrix factorization [8].

The gap we fill: prior work studies individual circuits in isolation. We exhaustively enumerate the *family* of circuits a small model uses for one task and quantify how redundancy differs between a dense and a weight-sparse model.

3 Methods

Models and task. We use two checkpoints from Gao et al. [6]: a fully dense `ss_d128_f1` (d_{model}=128, 2 layers; 2,816 maskable nodes) and a weight-sparse bridge model `ss_bridges_d1024_f0.015625` (d_{model}=1024, 1.56% nonzero weights; 22,528 nodes), both trained on the SimpleStories corpus [10]. The primary task, `dummy_pronoun`, completes "*When [name] went to the park,*" with the gender-correct pronoun; we report two-alternative-forced-choice accuracy (2-AFC: $\text{logit}_{\text{correct}} > \text{logit}_{\text{incorrect}}$) and full-vocabulary cross-entropy (CE). Unrelated comparison tasks are `dummy_tense` and `dummy_article`. Names are the 15 single-token characters from the SimpleStories generation prompt (the "cast"), gender-validated in fp32; we exclude names whose first token is itself a pronoun (e.g. *Henry* → "he") to remove a token-copying confound.

Circuit discovery. Mask-based pruning learns a binary node mask minimizing the contrastive task loss under *zero* ablation (target CE 0.15), with CARBS-tuned hyperparameters, over 100 random seeds per round. A pruned circuit can beat the full model, so full-model loss is not a feasibility gate.

Exhaustion loop. Each round we prune 100 times with 100 random seeds, take the node(s) of maximum frequency across successful circuits, add them to a cumulative excluded set, and re-prune—until 0/100 seeds succeed. Iteration count approximates the greedy minimum hitting set / set cover over the hypergraph of valid circuits.

Generalization. A names×templates split prunes on 80% of names × training templates, discretizes circuits on the same 80% of names × validation templates, and reports 2-AFC/CE only on held-out 20% names × held-out templates (snapshotted; 5 folds).

Selectivity. We ablate the cumulative core and measure logit-diff, 2-AFC, and CE on each task against matched-count random-node ablations (z-scores).

Probabilistic model. We treat the exclusion loop as a stochastic process over rounds t, with cumulative *excluded set* E_t ($E_0{=}\emptyset$, $E_{t+1}{=}E_t{\cup}R_t$). Treating pruning as a map from (E_t, ω) to (circuit, success) on a random seed ω, let N_t be the number of the 100 seeds that return a feasible circuit, so $N_t \sim \text{Binomial}(100, p(E_t))$. Let q_v be node v's marginal inclusion probability (its universality), so its frequency across the N_t survivors is $\text{Binomial}(N_t, q_v)$. The per-round "dump" R_t is the set of nodes tied at the maximum observed frequency M_t. Because the ceiling set (nodes in *every* survivor) is a subset of R_t, linearity of expectation gives $\mathbb{E}|R_t| \geq \sum_v q_v^{N_t}$—a ties-at-max-of-binomials law, tight in the low-N_t avalanche regime and undercounting at high N_t (the q^{N_t} cliff). Given fixed seeds the pipeline is deterministic, so all probabilistic statements concern counterfactual seed ensembles.

Atom decomposition and atom-peel. We assemble a circuit×node matrix X, factor $X \approx WH$ by NMF [8] into *atoms* (co-occurrence motifs), and separate a high-frequency *backbone* (freq ≥ 0.6) from the low-frequency *discriminative* atoms. *Atom-peel* iteratively removes the atom of greatest immediate held-out damage, re-prunes 100 seeds, re-factors, and repeats to exhaustion; a peel *collapses* when re-prune feasibility falls below 0.1. All cross-level readouts are functional—feasibility, recovery cost, held-out 2-AFC, loss—never node-overlap. Appendix A gives full definitions.

4 Results

Backup circuits are finite. On the dense model the pronoun task is solved by ∼63-node circuits; iteratively excluding the most-universal nodes drives the success rate to zero after 9 iterations (10-name task), accumulating an 84-node universal core. The success-rate trajectory always terminates at zero (fig. 1)—monotonically for a single name pool, and via collapse–rebound when the name pool or seed block changes. The model's pronoun ability is concentrated in finitely many key nodes, not freely re-synthesizable.

The core is task-selective (table 1). Ablating the dense core collapses pronoun 2-AFC 100%→65% but leaves the unrelated tense task behaviorally intact (100%; logit-diff 6.54→4.47, only reduced confidence), beyond matched random ablation. The sparse model shows the same pattern (z=8.9 pronoun; tense accuracy survives).[1] Nodes thus have specialized, task-aligned functions.

The core generalizes and is mechanistic. Held-out-name 2-AFC stays ≥ 98% (frequent names) and ≈ 96% (the three rarest names) at every iteration, even when loss-based confidence does not—the routing transfers, the calibration does not. Mechanistically the dense circuit is a single layer-1 attention head that copies the gender already present in the name's token embedding onto the to-be-generated position: ablating it inside a pruned circuit drops logit-diff 5.40→−0.02; patching its value at the name with the opposite-gender mean flips a female name's "she"→"he"

[1] The sparse selectivity figures come from the original weight-sparse exhaustion run; its 119-node core list was not retained in the current output snapshot, so these values are reported from the contemporaneous run record rather than regenerated from disk—and, per our seed-dependence result, a fresh run would recover a different core.

∼97% of the time (the reverse barely moves—"he" may be buttressed by redundant pathways); and its OV (output value) circuit projected onto the she−he direction separates genders perfectly. A layer-1 MLP counterbalances (direct logit attribution: attention +142%, MLP −42% of the she−he gap).

Exclusion dynamics are lawful but seed-dependent. The avalanche model reproduces per-round dump sizes in-sample (log-correlation 0.98 across nine runs; dumps 1–171); large dumps are "avalanches" where 2–3 surviving circuits share a near-identical node set. The terminal core is a *random variable* over the seed set: seeds 0–99 vs. 100–199 give cores of 84 vs. 67 nodes (Jaccard similarity 0.36) over 9 vs. 26 iterations. Post-collapse "rebounds" in the success rate are the discovery of *new* circuit families (within-iteration clustering beats a Bernoulli null; rebound families are genuinely new, not pre-existing minorities).

Weight-sparsity yields deep cheap redundancy over a small core (figs. 2 and 3). NMF (fig. 2) shows the dense circuit is mostly a large, diffuse frequency backbone with only weak, high-rank *discriminative* structure, whereas the sparse circuit carries a strong low-rank discriminative atom over a small backbone—a rank-2 atom basis beats a marginal-preserving (column-shuffled) null on held-out reconstruction by +0.265 (sparse) vs +0.050 (dense). Peeling atoms to exhaustion (fig. 3): the sparse model peels **34** discriminative atoms at near-zero behavioral cost (feasibility never approaches collapse, recovery cost flat ∼15–17 nodes, 2-AFC = 1.0 throughout) and *never collapses*—it runs out of discriminative atoms, exposing an irreducible 15-node core used by 100% of solutions. Forbidding those 15 core nodes—on top of the already-peeled ∼198-node periphery, with ∼22k nodes still free—drops feasibility to 0/100: a genuinely necessary cut, of which leave-one-out identifies only *3* individual articulation points (an attention read→discriminate spine). The dense model instead *collapses at depth 22*: recovery cost is flat (∼63–82 nodes) until a depth-17 phase transition where cheap attention basins exhaust, an MLP fallback basin appears, cost explodes to ∼140 nodes, and the task dies.

Negative results. Attempts to prove closed-form expressions for the iteration count (set cover) and the number of distinct "hypercircuits" (a spectral count of a block-diagonal Jaccard matrix; ≈ 4.25 effective on the dense task) did not survive scrutiny. Node-Jaccard proved backbone-confounded and was retired for functional and loadings-cosine metrics. "One clean motif per iteration" failed its survival check in *both* models—excised motifs re-form from substitute nodes—confirming redundancy is architecture-independent in kind, differing only in depth and cost.

Table 1: Effect of ablating the pronoun core (dense: 84 nodes; sparse: 119 nodes) vs. matched-count random ablation (parentheses). Pronoun breaks; the unrelated tense task survives behaviorally (accuracy preserved, confidence reduced). z compares the sparse core's ΔCE to random ablation.

Task	Relation	Dense 2-AFC full→core (rand)	Sparse acc. full→core (rand)	Sparse ΔCE (z) core (vs rand)
`dummy_pronoun`	control	100%→65% (99%)	1.00→0.50 (1.00)	+9.29 (8.9)
`dummy_tense`	unrelated	100%→100% (91%)	1.00→1.00 (0.95)	+3.59 (7.8)

5 Discussion

Universality ≠ necessity. Node frequency measures basin-of-attraction size (which solution seeds prefer), not functional load. The two come apart: frequency does not predict leave-one-out articulation, and most "universal" nodes are individually substitutable. The frequency-defined

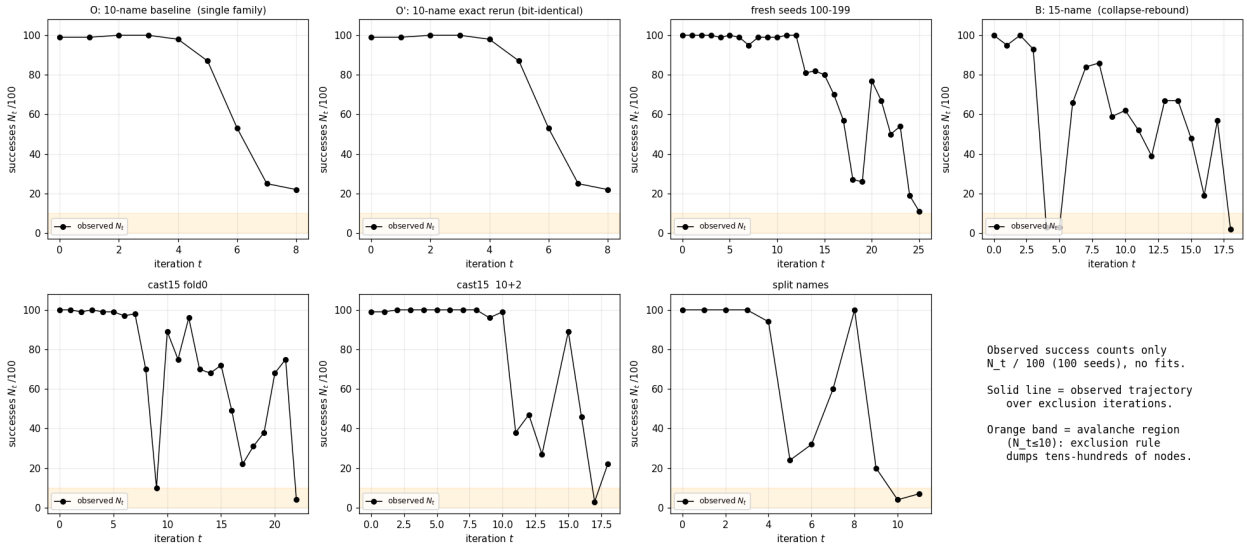


Figure 1: Per-iteration success counts ($N_t/100$) for seven exhaustion runs (observed trajectories, no fits). Backup circuits are finite (success $\to 0$). Single-family runs (O, O′) are monotone; pool/seed changes induce collapse–rebound (e.g. 15-name, cast15). The orange band ($N_t \leq 10$) is the avalanche regime, where one exclusion round dumps tens to hundreds of nodes.

backbone is therefore the wrong cleavage; true necessity concentrates in a few read→discriminate steps—an attention input that reads the name and the value/output that performs the gender discrimination.

Redundancy is real, but finite and structured. The weight-sparse model realizes a clean necessary-core / redundant-periphery decomposition: many structurally distinct but functionally identical copies of the name→gender route, bottoming out at a small irreducible core—an overcomplete frame over a low-dimensional functional subspace. Crucially this redundancy is bounded, yet the periphery genuinely re-routes: forbidding the ∼198 peeled discriminative nodes *alone* leaves the task feasible (99/100, substitutes recruited from the ∼22k free nodes), but the peel still exhausts its substitutable atoms after 34 levels and bottoms out at the irreducible 15-node core—forbidding *that* (with ∼22k nodes still free) gives 0/100. Substitution is therefore finite, drawn from a bounded trained-in pool rather than synthesized from arbitrary nodes. The dense model has lower overcompleteness and fails violently through an attention→MLP basin switch.

Implications. This causally supports the claim that weight-sparse models are more interpretable—their circuits are fewer, cleaner, and more universal—while qualifying faithfulness: the minimal pruned circuit is one *preferred* basin among many substitutable ones (re-pruning with it forbidden recovers behavior at ∼zero cost, recruiting different atoms), so reporting a single circuit overstates uniqueness. For AI safety the news is mixed: task selectivity means cores are localizable and ablations are targeted, but deep redundancy means unlearning or capability removal cannot succeed by deleting one circuit—breaking the sparse atom requires removing ≥ 2 of a redundant pair, and only forbidding the entire discriminative pool kills the task.

Limitations. Our models are tiny 2-layer SimpleStories networks; results center on one task family; zero ablation is a harder, more off-distribution intervention than mean ablation; loss-based generalization is confounded by the full model's own calibration (we therefore lead with 2-AFC); and the avalanche model's one-step-ahead dump prediction is biased at high success counts. **Future**

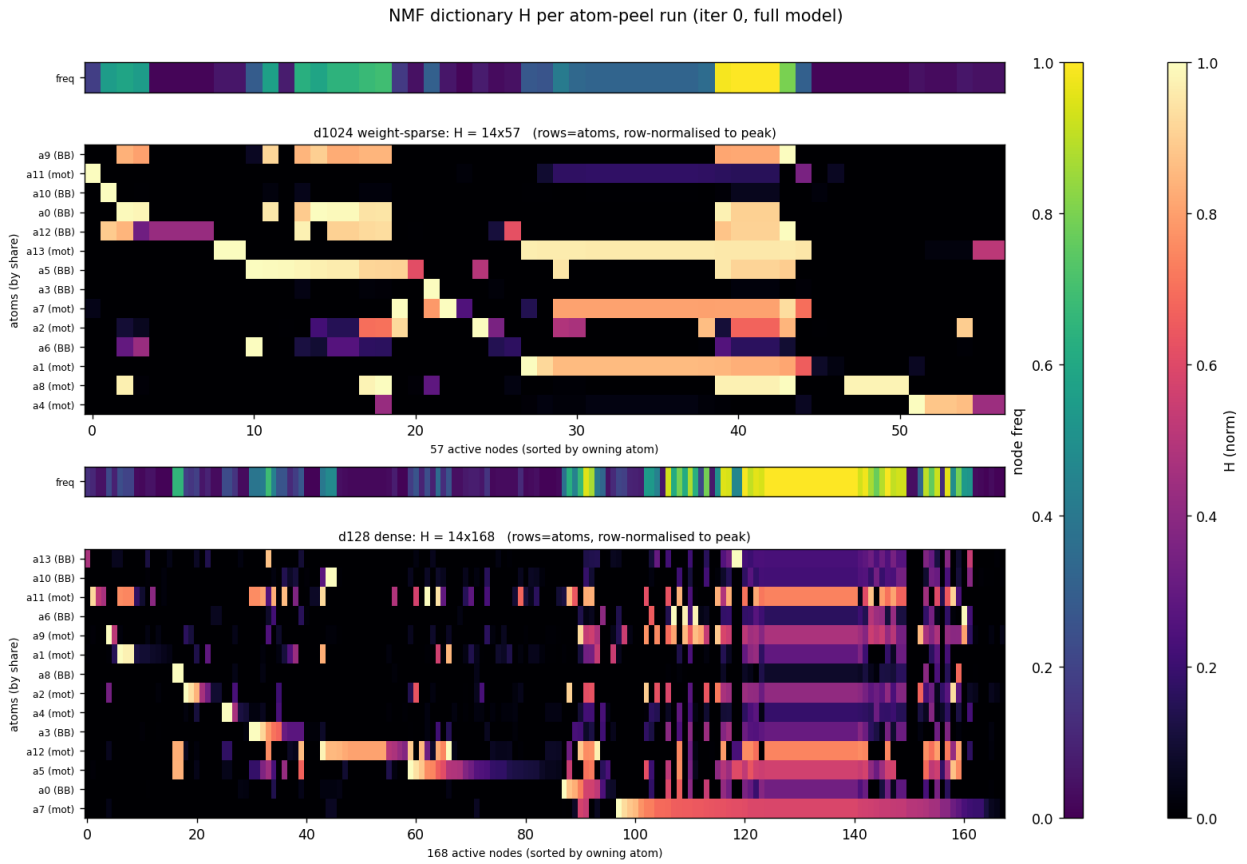
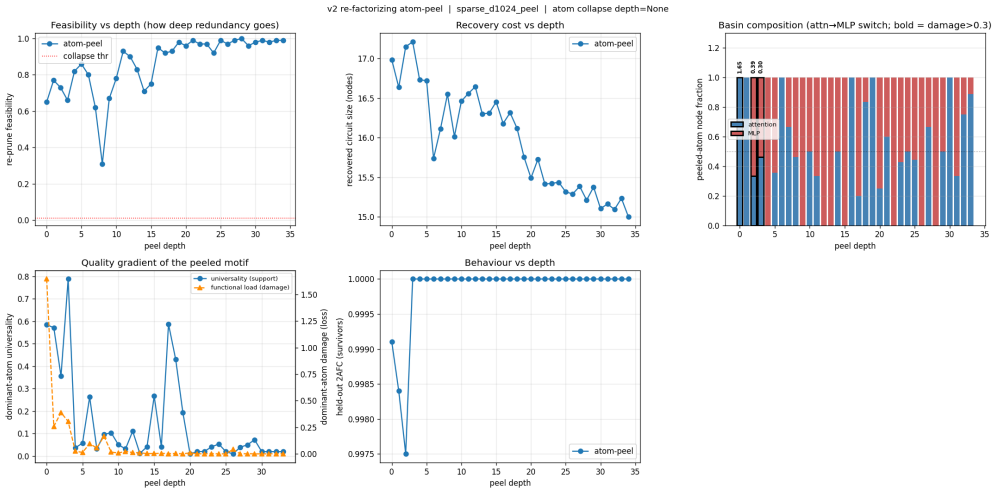


Figure 2: NMF dictionaries (H, rows = atoms, row-normalized) at iteration 0; the strip above each heatmap is per-node marginal frequency. The weight-sparse $d1024$ circuit (top, 57 active nodes) has a small backbone (BB) and compact discriminative atoms over low-frequency nodes; the dense $d128$ circuit (bottom, 168 active nodes) is dominated by a large, diffuse frequency backbone, its discriminative atoms weak and high-rank. Held-out reconstruction against a marginal-preserving (column-shuffled) null confirms the gap: at rank 2 the atom basis beats the null by $+0.265 \pm 0.017$ (sparse) vs $+0.050 \pm 0.006$ (dense)—gaps of $\approx$15 and $\approx$8 standard errors above zero.

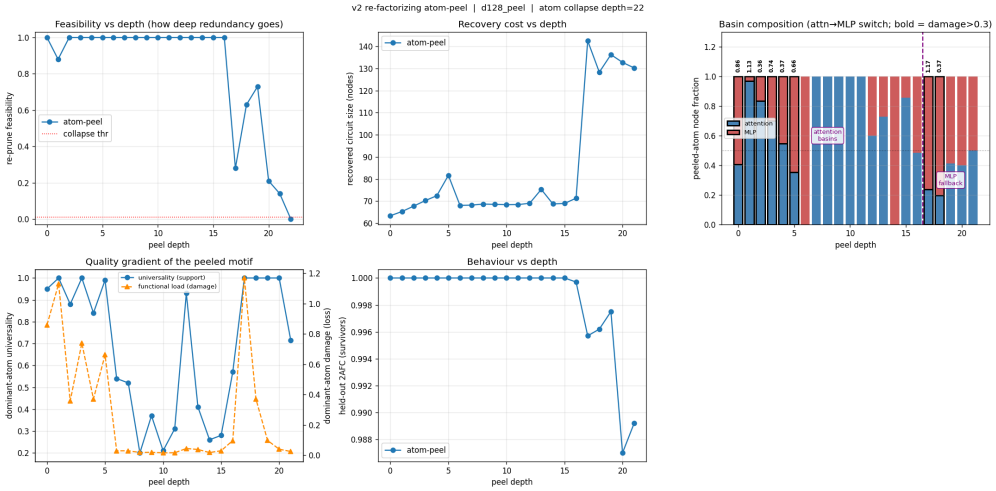
work: (1) instead of using NMF to obtain atoms, train sparse autoencoders on circuit masks to obtain atoms; (2) extend the protocol to more tasks and deeper models.

6 Conclusion

We exhausted the backup circuits of a pronoun-gender task in a dense and a weight-sparse transformer. Backup circuits are finite; the discovered core is task-selective, generalizes to unseen names, and has an identified copy-from-embedding mechanism. Its exclusion dynamics obey a probabilistic "avalanche" law that we validate across nine runs and whose endpoint is seed-dependent. Decomposing circuits into NMF atoms and peeling them to exhaustion separates a small necessary core from a redundant periphery and shows that weight-sparsity buys deep, cheap, modular redundancy over a clean irreducible core, whereas dense redundancy is finite and fails through a cost-exploding phase transition. Universality and necessity come apart: the most universal nodes are mostly substitutable, and necessity concentrates in a few read→discriminate steps. Together these give a quantitative,



(a) Weight-sparse $d1024$



(b) Dense $d128$

Figure 3: Iterative atom-peel. (a) The weight-sparse model peels 34 discriminative atoms at near-zero cost: feasibility never reaches the collapse threshold, recovery cost stays flat (∼15–17 nodes), held-out 2-AFC stays 1.0; it exhausts by running out of discriminative atoms, leaving an irreducible ∼15-node core. (b) The dense model holds flat until a depth-17 phase transition (attention→MLP basin switch) where recovery cost explodes to ∼140 nodes and feasibility collapses to 0 at depth 22: finite, expensive redundancy. The top-right "basin composition" panels show each peeled atom's node split between attention (blue) and MLP (red), with the basin-defining atoms (damage > 0.3) outlined and labelled by their damage: for the dense model (b) the cheap attention-centred basins exhaust and an MLP-dominated fallback basin (damage 1.17) appears at the depth-17 transition (purple line), whereas the weight-sparse model (a) stays mixed throughout with no switch. In the bottom-left "quality gradient" panels, *functional load (damage)* (orange, right axis) is the mean held-out cross-entropy increase (ΔCE) from zero-ablating the peeled atom's nodes from each surviving pruned circuit, averaged over circuits—how much task loss the atom carries—whereas *universality (support)* (blue, left axis) is the fraction of those circuits that contain it.

causal account of circuit redundancy that both supports and tempers the case for weight-sparse interpretability.

References

[1] Arthur Conmy, Augustine N. Mavor-Parker, Aengus Lynch, Stefan Heimersheim, and Adrià Garriga-Alonso. Towards automated circuit discovery for mechanistic interpretability, 2023.

[2] Róbert Csordás, Christopher Potts, Christopher D. Manning, and Atticus Geiger. Recurrent neural networks learn to store and generate sequences using non-linear representations, 2024.

[3] Jacob Drori. Weight-sparse circuits may be interpretable yet unfaithful. LessWrong, 2026. https://www.lesswrong.com/posts/sHpZZnRDLg7ccX9aF/weight-sparse-circuits-may-be-interpretable-yet-unfaithful.

[4] Nelson Elhage, Neel Nanda, Catherine Olsson, et al. A mathematical framework for transformer circuits. *Transformer Circuits Thread*, 2021.

[5] Jiahai Feng and Jacob Steinhardt. How do language models bind entities in context?, 2024.

[6] Leo Gao, Achyuta Rajaram, Jacob Coxon, Soham V. Govande, Bowen Baker, and Dan Mossing. Weight-sparse transformers have interpretable circuits, 2025.

[7] Michael Hanna, Sandro Pezzelle, and Yonatan Belinkov. Have faith in faithfulness: Going beyond circuit overlap when finding model mechanisms, 2024.

[8] Daniel D. Lee and H. Sebastian Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755):788–791, 1999.

[9] Chris Olah, Nick Cammarata, Ludwig Schubert, Gabriel Goh, Michael Petrov, and Shan Carter. Zoom in: An introduction to circuits. *Distill*, 2020.

[10] SimpleStories Project. SimpleStories. Hugging Face Datasets, 2024. https://huggingface.co/datasets/SimpleStories/SimpleStories.

[11] Kevin Wang, Alexandre Variengien, Arthur Conmy, Buck Shlegeris, and Jacob Steinhardt. Interpretability in the wild: A circuit for indirect object identification in GPT-2 small, 2022.

A Expanded methods: atom decomposition and atom-peel

Circuit decomposition into atoms. For an ensemble of successful pruned circuits we build a binary circuit×node incidence matrix X (row = one circuit; X_{cv}=1 iff node v is used) and factor it by NMF, $X \approx WH$ [8]. Each row of H is an *atom*—a soft, non-negative distribution over nodes capturing a set that co-occurs *beyond* its members' independent marginal rates (a reusable circuit motif); each row of W holds a circuit's loadings, so a circuit is a sparse *mixture* of atoms rather than a member of one hard cluster. We set the rank K by held-out reconstruction: for each candidate K we fit NMF on a train split and score relative reconstruction error on held-out circuits, comparing the real error against a column-shuffled null (each node permuted independently across circuits—marginals preserved, co-occurrence destroyed), and take the smallest K within one standard error of the error minimum (*parsimonious*). Nodes split by marginal frequency into a *backbone* (freq ≥ 0.6; the substrate present in nearly every circuit and already explained by marginals

alone) and the low-frequency *discriminative* nodes that distinguish circuits; each discriminative node is assigned to the atom that loads it most (argmax of H, kept when that atom owns $\geq 25\%$ of its total loading), so a *discriminative atom* is the disjoint set of non-backbone nodes that atom owns. Backbone-dominated atoms own no discriminative nodes and drop out.

Atom-peel. Atom-peel strips this redundant substrate one atom at a time. At each level we score every discriminative atom by its *immediate held-out damage*: we zero-ablate the atom's owned nodes from each current surviving circuit and average the resulting increase in held-out cross-entropy over those circuits—a single-knockout estimate of how much task loss the atom carries, measured *before* the model is allowed to re-route. We *peel* the maximum-damage atom, adding its nodes to a cumulative forbidden set F; then *re-prune* 100 fresh seeds with F pinned off to obtain the next level's circuits; then *re-factor* those circuits with a freshly re-selected K. Re-factoring is necessary because fallback solutions live in a different basis than the original atoms (re-pruned circuits load only ~ 0.4 onto them), so each level is described in its own dictionary. We repeat to exhaustion—until the re-prune success rate collapses (< 0.1) or no discriminative atoms remain.

Functional cross-level readouts. Because each level is re-factored, atom identity is *not* comparable across levels, and node-overlap metrics such as Jaccard are backbone-dominated and excision-confounded. We therefore never compare levels by node identity; all cross-level quantities are *functional*, i.e. behavioral or causal. *Feasibility* is the re-prune success rate—the fraction of the 100 seeds that, with F forbidden, recover a circuit reaching the target CE (0.15)—answering whether the task can still be solved at all. *Recovery cost* is the mean node count of those recovered circuits, measuring how expensive the rebuild is (cheap vs. deep redundancy). We also track the held-out 2-AFC of the survivors (is behavior preserved?) and the ΔCE damage above. A peel *collapses* when feasibility falls below 0.1.