How Much Attribution Mass Lands on Tokens that Cannot Matter? A Faithfulness Benchmark for Transformers

Anonymous authors
Paper under double-blind review

Abstract

Attribution methods for transformers are usually judged by whether their heatmaps look plausible to a human reader. We take a more direct route. We train small decoder-only transformers on four algorithmic tasks (sparse parity, indexed retrieval, associative recall, and modular addition with distractors) in which every input position is, by construction, either causally relevant or provably unable to affect the output. Any attribution mass landing on a provably irrelevant position is an error, with no threshold, reference model, or human judgment involved in that verdict. Under one pre-registered protocol we evaluate fourteen scoring rules, and every number carries a t-based confidence interval over 8 to 11 independently trained seeds. Our most surprising finding concerns training dynamics: through the grokking transition on modular addition, the faithfulness of attention-based methods degrades sharply and becomes far more seed-dependent, while test accuracy rises to 100%. Explanations of the better model are worse. Second, cheap attribution and causal measurement diverge exactly where feature interactions matter: on sparse parity a resample-ablation oracle reaches 0.031 false-positive mass while every attribution method stays above 0.49, a gap that rank-preserving sharpening cannot close, so the failure is misidentification rather than diffuse scoring. Third, protocol choices can outweigh the choice of method: on the three tasks with first-order structure, moving integrated gradients from a marginal-mean baseline to a zero baseline raises false-positive mass by factors of 2 to 22. Fourth, paired tests show the top two methods are statistically indistinguishable on two of four tasks, so single-seed rankings among mid-ranked methods carry little information. We release the task generators, model zoo, attribution harness, and evaluation suite.

1 Introduction

When an attribution heatmap explains a transformer's prediction, the claim behind it is that the highlighted tokens are the ones the model actually used. Checking that claim requires knowing which tokens the model used, which is exactly what the heatmap was supposed to tell us. The literature escapes this circle in two ways. Perturbation-based faithfulness metrics score explanations by deleting inputs, but deletion pushes the input off its training distribution and the scores inherit that confound (Hooker et al., 2019; Hase et al., 2021). Ground-truth benchmarks in vision, text, and tabular data (Li et al., 2023; Agarwal et al., 2022) avoid deletion, but their relevance labels are approximations produced by injecting synthetic features into natural data.

A separate line of work builds transformers whose internal mechanisms are known, by compiling programs into weights (Lindner et al., 2023) or training semi-synthetic circuits (Gupta et al., 2024). Its ground truth lives at the level of attention heads and circuit components, not input tokens. The combination is missing. Algorithmic tasks make input-level ground truth exact rather than approximate, because the task definition fixes which input positions can influence the output.

This paper builds that combination. Each task emits a binary relevance label for every input position of every example, and each contains positions guaranteed by construction to have zero causal influence on the target. Attribution mass on such a position cannot be justified by any account of

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

what the model computes. Our headline metric, false-positive mass (FPM), is the fraction of total normalized attribution mass on provably irrelevant positions. It needs no threshold, is comparable across methods, and is immune to the objection that the faithfulness metric is itself unvalidated.

Two design decisions shape everything downstream. The first is the distinction between task-level and model-level ground truth. Faithfulness is a property of an explanation with respect to the *model*, not the task: a model that learned a shortcut is faithfully explained by a heatmap highlighting the shortcut. We therefore evaluate only checkpoints passing a pre-registered held-out accuracy threshold, measure model-level causal importance directly by resample ablation, and report the agreement between task labels and measured causal importance as a result (Table 4) rather than assuming it. The second is resample ablation itself: wherever we measure a token's causal effect we replace it with a value drawn from that position's marginal distribution, because zero and mask ablation push the input off-distribution and conflate "this token mattered" with "this input is now strange" (Hase et al., 2021).

Small models make seeds cheap, and we spend the savings on statistics. Every number carries a t-based confidence interval over 8 to 11 independently trained seeds, between-seed variance is reported separately from between-example variance, and because all methods score identical examples on identical checkpoints, method comparisons use paired tests.

Contributions.

1. A benchmark of four algorithmic tasks with exact per-example input-relevance labels and provably irrelevant positions, plus a unified harness evaluating fourteen scoring rules under one pre-registered protocol, and a released zoo of 50 trained transformers (§3–§5).
2. Evidence that faithfulness moves during training: attention-family explanations degrade, and their seed variance grows, through the grokking transition while the model itself improves (§7).
3. A measurement of the gap between per-token attribution and causal measurement where feature interactions matter: on sparse parity the oracle reaches 0.031 FPM while every attribution method stays above 0.49, and rank-preserving sharpening does not close it (§6).
4. Protocol and statistical results with direct practical import: the integrated-gradients baseline changes FPM by factors of 2 to 22, and paired tests leave the top two methods indistinguishable on two of four tasks (§6).

2 Related work

Attribution methods for transformers. Practitioners inspect raw attention weights, aggregate them across layers with rollout (Abnar & Zuidema, 2020), or combine them with gradients (Chefer et al., 2021). The gradient family includes plain saliency (Simonyan et al., 2014), gradient$\times$input (Shrikumar et al., 2017), integrated gradients (Sundararajan et al., 2017), and GradCAM-style aggregation (Selvaraju et al., 2017). Perturbation approaches include SHAP variants (Lundberg & Lee, 2017) and leave-one-out deletion. Whether attention weights explain anything has been argued in both directions (Jain & Wallace, 2019; Wiegreffe & Pinter, 2019); our results give that debate a quantitative footing on models where the answer can be checked.

Faithfulness evaluation. Deletion-based metrics carry an out-of-distribution confound (Hooker et al., 2019; Hase et al., 2021). Violation tests probe self-consistency (Liu et al., 2022), and studies on vision transformers report mostly negative results (Wu et al., 2024). Ground-truth benchmarks in natural domains (Li et al., 2023; Agarwal et al., 2022; Lee et al., 2025) approximate relevance labels. Sanity checks (Adebayo et al., 2018) showed some methods barely respond to model parameters; passing such a check is necessary but not sufficient, and our causal-agreement metric asks the stronger question of whether scores track measured causal effects.

Interaction-aware attribution. A separate family attributes predictions to *groups* of features: the Shapley-Taylor interaction index (Sundararajan et al., 2020), Integrated Hessians (Janizek et al., 2021), and Archipelago (Tsang et al., 2020). Our sparse-parity result (§6) is exactly the failure mode these were designed for, since a k-way XOR has no first-order structure to recover. We eval-

Table 1: Task suite. Relevant positions are known by construction; every other payload position provably cannot affect the target.

Task	Output	Relevant set	Per-ex.?	Irrel.
Sparse parity (3-of-40)	XOR of hidden subset S	$S, \lvert S\rvert = 3$	no	$37/40$
Indexed retrieval (n=16)	x_i for index token i	$\{i, x_i\}$	yes	$15/17$
Associative recall (8 pairs)	value of queried key	{query, key, value}	yes	$14/17$
Modular addition (p=97, 8 distr.)	$(a + b) \bmod p$	$\{a, b\}$	no	$8/10$

uate the per-token family because that is what practitioners read off heatmaps and what the tooling ecosystem produces, and our parity measurement quantifies the price of that choice. Benchmarking the interaction-aware family here is the natural next use of this suite: the ground truth extends to interactions unchanged, since the relevant subset is known exactly.

Algorithmic transformers with known structure. Tracr compiles programs into transformer weights (Lindner et al., 2023) and InterpBench trains semi-synthetic circuits (Gupta et al., 2024). Grokking on modular arithmetic (Power et al., 2022; Nanda et al., 2023) and the phase transition in sparse-parity learning (Barak et al., 2022) give well-understood training dynamics. All of this certifies internal mechanisms; none provides input-token relevance labels, which is the gap this benchmark fills.

3 Benchmark design

All tasks share one interface. A sequence $[\texttt{BOS}, x_1, \ldots, x_L, \texttt{READOUT}]$ is classified from the final position's logits, and the generator emits a binary relevance mask over payload positions. Structural tokens are excluded from attribution scoring; mass on BOS is measured separately as sink mass (Xiao et al., 2024) rather than counted as error, because softmax attention must place its mass somewhere.

Sparse parity (Barak et al., 2022): 40 binary tokens, target the XOR of a fixed hidden subset of 3 positions. The cleanest ground truth available, with a known feature-learning phase transition. **Indexed retrieval**: an index token followed by 16 value tokens, target the indexed value; the relevant set changes every example, so a method cannot succeed by learning a static positional prior. **Associative recall**: 8 key-value pairs followed by a query key, the classic induction-head task (Olsson et al., 2022), adding a long-range content-based dependency. When measuring causal effects here, resampled keys are redrawn so they never collide with the query, since a collision would change the target. **Modular addition with distractors** (Power et al., 2022; Nanda et al., 2023): operands a, b plus 8 uniformly sampled distractors, which give the classic grokking setup a provable false-positive test.

The tasks split into two regimes, and the split matters for interpretation. On parity and modular addition the relevant set is fixed, so these test whether a method recovers static structure; a positional prior that ignores the model entirely scores FPM $= 0$ on both by construction (Table 2). On retrieval and recall the relevant set changes every example, so only these test whether a method explains a particular computation, and only they can rank a method against a static prior. We keep both regimes because they answer different questions: fixed-set tasks expose absolute failure (a method that cannot find three fixed bits will not find anything harder), while per-example tasks discriminate between methods and priors. Cross-task comparisons in §6 treat the regimes separately.

Causal ground truth. For every evaluated checkpoint we also measure model-level importance directly. Each payload position is replaced by values drawn from its marginal distribution (4 to 8 resamples) and we record the mean absolute change in the target logit. This vector validates the task-level labels (Table 4) and is the reference for the rank-correlation metric below. Leave-one-out attribution built on the same primitive appears among the scored methods as an oracle-style upper reference, not a competitor.

4 METHODS, PROTOCOL, AND METRICS

Weak baselines are mandatory: a recurring finding in this literature is that sophisticated methods fail to beat trivial ones. We include *random* scores, *uniform* scores, and a *positional prior* (per-position relevance frequency over 4,096 training samples). The scored methods are raw last-layer attention (head mean, readout row); attention rollout (Abnar & Zuidema, 2020) with 0.5 residual mixing; attention$\times$gradient (last layer, head mean of $A \odot \partial y/\partial A$); a GradCAM-style variant with per-head weights from mean attention gradients (Selvaraju et al., 2017); plain saliency, the ℓ_2 norm of embedding gradients (Simonyan et al., 2014); gradient$\times$input (Shrikumar et al., 2017); integrated gradients (Sundararajan et al., 2017) via Captum (Kokhlikyan et al., 2020) with 64 steps and two baselines reported separately (the zero embedding, and the mean embedding over 256 marginal samples); Chefer relevance propagation (Chefer et al., 2021); KernelSHAP (Lundberg & Lee, 2017) with 256 coalition samples and marginal-resample baselines; and leave-one-out resample ablation.

The protocol was fixed before any method ran and is identical across methods and tasks: which layer each method reads, head-mean aggregation, the integrated-gradients baselines and step count, and normalization (absolute value, restricted to payload positions, sum to one). Rank-based metrics are primary because raw scores are not comparable across methods; inconsistency in exactly these choices is a known source of attribution results that do not reproduce.

Metrics. Let $s \in \Delta^{L-1}$ be the normalized attribution over payload positions and R the relevant set. **False-positive mass** is $\mathrm{FPM}(s) = \sum_{j \notin R} s_j$. We also report AUROC, average precision, and precision at $k = |R|$ against the binary labels; per-example Spearman correlation between scores and measured resample-ablation effects, which tests model-level faithfulness without using task labels at all; **task–model agreement**, the Spearman correlation between task labels and measured causal effects, which quantifies how safe the "known ground truth" assumption is; and **sink mass**, the fraction of raw attribution magnitude on `BOS`. Every number is a mean over seeds with a 95% interval from between-seed variance using the Student t distribution, since n ranges from 8 to 11 and a normal approximation would be optimistic. Between-example variance is tracked separately, because pooling the two understates seed-level instability. Comparisons between methods are paired, with paired t-tests and Wilcoxon signed-rank checks on per-seed means; the comparisons were fixed in advance (best against random, best against second-best, and the two integrated-gradients baselines against each other), so we report uncorrected p-values, and every conclusion either clears $p < 10^{-3}$ or fails to reach 0.05, so a Bonferroni-style correction over these three families changes none of them.

5 EXPERIMENTAL SETUP

Models and training. Decoder-only transformers (2 to 3 layers, d_{model} 128 to 256, 4 heads, learned positional embeddings) trained with AdamW under fixed hyperparameters per task; within a cell only the seed varies. Checkpoints follow a log-spaced schedule. Modular addition uses a 40% train split of the $97^2 = 9409$ operand pairs with weight decay 1.0 to induce grokking. Parity requires weight decay 0: every tested value of 10^{-2} or above pinned all seeds at chance accuracy for at least 60k steps, which is worth knowing if you train sparse parity yourself. Full configurations are in Appendix A.

Filtering and seed accounting. Attribution is evaluated only on runs whose final held-out accuracy exceeds the pre-registered threshold of 0.95. The pass counts are results, not nuisances: parity 11/20 seeds (8 never left the plateau within 30k steps, one was caught mid-transition at 0.755), retrieval 10/10, recall 8/10, modular addition 9/10. Under identical hyperparameters, whether the mechanism forms at all depends on the seed for three of four tasks. This filtering is a selection effect and we flag it: seeds that generalize may differ systematically from seeds that never left the plateau, so our results describe models conditioned on success. That conditioning matches practice, since deployed models are also selected for working, but it is a restriction on the population all the same.

Evaluation. Final-checkpoint results use 512 held-out examples per task and 8 resamples per causal measurement, with all methods seeing identical examples and checkpoints. Training-

Table 2: False-positive mass (mean ± 95% t-CI over seeds; lower is better). Bold marks the best method per task, excluding baselines and the oracle. Random and uniform sit at each task's irrelevant-fraction floor. The positional prior is exact on the two fixed-relevant-set tasks by construction.

Method	Parity	Retrieval	Recall	Modular
Random	0.925 ± 0.000	0.885 ± 0.000	0.820 ± 0.000	0.796 ± 0.000
Uniform	0.925 ± 0.000	0.882 ± 0.000	0.824 ± 0.000	0.800 ± 0.000
Positional prior	0.000 ± 0.000	0.469 ± 0.000	0.583 ± 0.000	0.000 ± 0.000
Raw attention (last)	0.719 ± 0.162	0.007 ± 0.001	0.552 ± 0.154	0.474 ± 0.272
Attention rollout	0.574 ± 0.068	0.444 ± 0.017	0.624 ± 0.074	0.175 ± 0.086
Attention×grad	0.714 ± 0.152	**0.003 ± 0.001**	0.479 ± 0.206	0.399 ± 0.228
GradCAM	0.762 ± 0.122	0.182 ± 0.030	0.570 ± 0.190	0.455 ± 0.275
Saliency (ℓ_2)	**0.496 ± 0.059**	0.292 ± 0.015	0.347 ± 0.133	0.054 ± 0.028
Grad×input	0.729 ± 0.159	0.068 ± 0.028	0.191 ± 0.120	0.261 ± 0.186
IG (zero base)	0.584 ± 0.051	0.543 ± 0.085	0.535 ± 0.125	0.347 ± 0.107
IG (marginal base)	0.518 ± 0.069	0.075 ± 0.015	0.259 ± 0.079	**0.016 ± 0.010**
Chefer LRP	0.528 ± 0.085	0.165 ± 0.064	**0.109 ± 0.059**	0.081 ± 0.039
KernelSHAP	0.593 ± 0.013	0.390 ± 0.004	0.475 ± 0.003	0.220 ± 0.004
LOO resample (oracle)	0.031 ± 0.017	0.020 ± 0.002	0.034 ± 0.017	0.005 ± 0.001

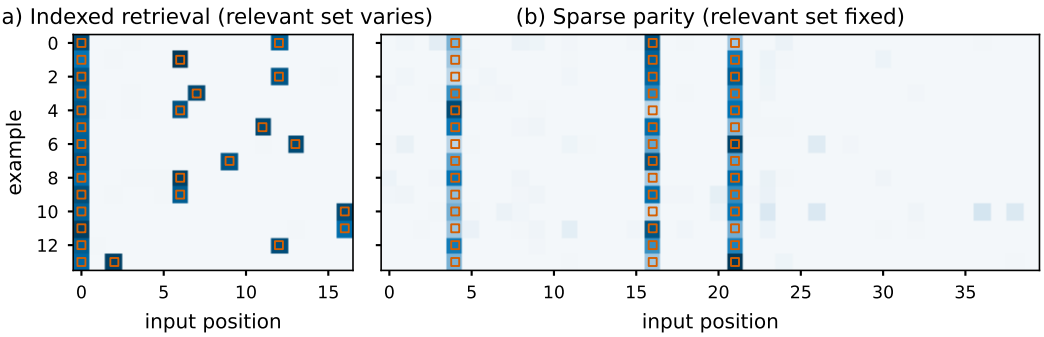

Figure 1: Causal reference (leave-one-out resample ablation) on generalizing models, for 14 held-out examples of two tasks. Shading is normalized causal mass per input position; orange squares mark the positions the task definition makes relevant. (a) On indexed retrieval the relevant set moves with the index token, and the measured dependence follows it. (b) On sparse parity the measured dependence concentrates on exactly the three bits of the hidden subset, with negligible mass on the other 37. This is the evidence behind the claim that parity models are not the problem: the models really do depend on the three relevant bits, so attribution failures there are failures of the explanation.

dynamics results (§7) use 12 log-spaced checkpoints per seed at 128 examples and 4 resamples. Everything runs on a single 8 GB laptop GPU in roughly one GPU-day.

6 Results at the final checkpoint

No method is faithful everywhere. The best non-oracle method differs on every task: saliency on parity (0.496), attention×gradient on retrieval (0.003), Chefer relevance propagation on recall (0.109), and integrated gradients with a marginal baseline on modular addition (0.016). The same attention-family methods that are nearly perfect on retrieval, where a clean induction-style circuit attends directly to the relevant slot, put 40% to 76% of their mass on provably irrelevant tokens on parity and modular addition. Conclusions about attribution faithfulness drawn from one task family do not transfer to another.

Table 3: AUROC of attribution scores against ground-truth relevance (mean ± 95% t-CI over seeds).

Method	Parity	Retrieval	Recall	Modular
Random	0.502 ± 0.000	0.486 ± 0.000	0.515 ± 0.000	0.509 ± 0.000
Uniform	0.500 ± 0.000	0.500 ± 0.000	0.500 ± 0.000	0.500 ± 0.000
Positional prior	1.000 ± 0.000	0.755 ± 0.000	0.659 ± 0.000	1.000 ± 0.000
Raw attention (last)	0.521 ± 0.054	0.709 ± 0.074	0.603 ± 0.083	0.722 ± 0.267
Attention rollout	0.807 ± 0.059	0.988 ± 0.020	0.866 ± 0.058	**1.000 ± 0.000**
Attention×grad	0.522 ± 0.052	0.686 ± 0.069	0.603 ± 0.084	0.839 ± 0.127
GradCAM	0.519 ± 0.040	0.665 ± 0.058	0.584 ± 0.050	0.738 ± 0.224
Saliency (ℓ_2)	0.808 ± 0.061	**0.999 ± 0.001**	**0.961 ± 0.032**	**1.000 ± 0.000**
Grad×input	0.677 ± 0.094	0.740 ± 0.064	0.649 ± 0.064	0.894 ± 0.109
IG (zero base)	0.836 ± 0.054	0.903 ± 0.033	0.771 ± 0.073	0.909 ± 0.064
IG (marginal base)	**0.927 ± 0.020**	0.990 ± 0.002	0.938 ± 0.022	1.000 ± 0.001
Chefer LRP	0.722 ± 0.049	0.975 ± 0.041	0.768 ± 0.056	0.994 ± 0.010
KernelSHAP	0.631 ± 0.010	0.898 ± 0.003	0.889 ± 0.005	0.955 ± 0.007
LOO resample (oracle)	0.999 ± 0.003	1.000 ± 0.000	0.999 ± 0.001	1.000 ± 0.000

Attribution and causal measurement diverge where interactions matter. Sparse parity is the sharpest version of this. The causal oracle, which merely measures each token's resample-ablation effect, reaches 0.031 FPM; every attribution method stays above 0.49, against a random floor of 0.925. The models are not the problem: they reach 100% held-out accuracy, and the oracle confirms they depend almost entirely on the three relevant bits (Figure 1b). The mechanism is no mystery, and we state it plainly: XOR is non-additive, each relevant bit's marginal effect flips sign depending on the other two, and per-token first-order scores have no stable signal to latch onto. In that sense the direction of the result is expected. What the benchmark contributes is the size of the penalty, measured exactly: on a three-way interaction, cheap attribution loses to a causal probe costing a few forward passes per token by a factor of sixteen in FPM, and the gap does not close with more training (§7). Consistent with the sign-instability account, saliency, which discards sign, beats gradient×input, which multiplies in exactly the unstable quantity (0.496 against 0.730).

The integrated-gradients baseline matters more than the method, on three tasks of four. Switching integrated gradients from a marginal-mean baseline to the zero embedding moves FPM from 0.075 to 0.543 on retrieval, 0.016 to 0.347 on modular addition, and 0.259 to 0.535 on recall (paired $p \leq 3 \times 10^{-3}$ on all three). These are factors of 2 to 22, larger than most gaps between distinct methods, and on retrieval the zero baseline is worse than the trivial positional prior. Parity is the exception, and it qualifies the rule rather than confirming it: there the two baselines are statistically indistinguishable (0.584 against 0.518, $p = 0.10$), and under the sharpening analysis of §6.1 the ordering flips (0.306 against 0.361 at γ=8) though not significantly ($p = 0.38$). So the marginal baseline dominates wherever a first-order signal exists and its advantage disappears on the interaction-dominated task, where neither baseline recovers the relevant bits. The reason to prefer an in-distribution baseline is that it helps when attribution works at all, not that it rescues attribution where the method has no signal to integrate. A paper reporting "integrated gradients" without naming the baseline has reported an underdetermined method.

Rank metrics soften the picture but do not reverse it. By AUROC (Table 3), several methods look respectable on tasks where their FPM is poor: integrated gradients with a marginal baseline reaches 0.93 AUROC on parity while placing half its mass on irrelevant tokens. The metrics disagree because AUROC rewards ranking relevant tokens above the *typical* irrelevant token, while FPM charges for the total mass landing in the wrong place, and a practitioner reading a heatmap experiences mass, not rank. On parity the attention-family methods sit at chance AUROC (0.52), so their heatmaps carry no relevance information there at all. The model-level check agrees with FPM: Spearman correlation against measured causal effects ranks methods consistently with it, so the failures are failures against the model rather than artifacts of task labels. Task–model agreement (Table 4) is 0.46 to 0.70, high enough to validate the filtering and low enough to justify measuring both; on generalizing models the causal reference concentrates nearly all mass on the constructed

Table 4: How safe is the "known ground truth" assumption, and where does the attention sink sit? Task–model agreement is the Spearman correlation between binary task-level labels and continuous measured causal effects; ties within the irrelevant set cap its ceiling below one. Sink mass is for raw attention.

	Parity	Retrieval	Recall	Modular
Task–model agreement	0.464 ± 0.019	0.559 ± 0.000	0.660 ± 0.002	0.696 ± 0.000
BOS sink mass (raw attention)	0.045 ± 0.046	0.000 ± 0.000	0.017 ± 0.022	0.050 ± 0.027

Table 5: Sharpening analysis on sparse parity (mean $\pm$ 95% t-CI over 11 seeds). Entropy is the normalized entropy of the score distribution (1 = uniform). FPM_γ applies the rank-preserving transform $s \mapsto s^\gamma$ before measuring FPM; γ=1 is the headline metric. Sharpening cannot repair any method: the best sharpened value (IG zero, 0.306) stays 76$\times$ above the oracle.

Method	Entropy	$\text{FPM}_{\gamma=1}$	$\text{FPM}_{\gamma=2}$	$\text{FPM}_{\gamma=4}$	$\text{FPM}_{\gamma=8}$
Random	0.948 ± 0.000	0.925 ± 0.000	0.924 ± 0.000	0.925 ± 0.000	0.927 ± 0.000
Uniform	1.000 ± 0.000	0.925 ± 0.000	0.925 ± 0.000	0.925 ± 0.000	0.925 ± 0.000
Positional prior	0.298 ± 0.000	0.000 ± 0.000	0.000 ± 0.000	0.000 ± 0.000	0.000 ± 0.000
Raw attention (last)	0.368 ± 0.112	0.719 ± 0.162	0.692 ± 0.168	0.679 ± 0.170	0.675 ± 0.168
Attention rollout	0.547 ± 0.059	0.574 ± 0.068	0.436 ± 0.102	0.351 ± 0.127	0.320 ± 0.140
Attention$\times$grad	0.322 ± 0.090	0.714 ± 0.152	0.691 ± 0.154	0.687 ± 0.152	0.749 ± 0.124
GradCAM	0.367 ± 0.096	0.762 ± 0.122	0.752 ± 0.128	0.773 ± 0.130	0.907 ± 0.038
Saliency (ℓ_2)	0.448 ± 0.044	0.496 ± 0.059	0.398 ± 0.086	0.356 ± 0.105	0.355 ± 0.115
Grad$\times$input	0.301 ± 0.073	0.729 ± 0.159	0.723 ± 0.158	0.737 ± 0.150	0.817 ± 0.108
IG (zero base)	0.625 ± 0.036	0.584 ± 0.051	0.393 ± 0.076	0.324 ± 0.079	0.306 ± 0.077
IG (marginal base)	0.496 ± 0.044	0.518 ± 0.069	0.422 ± 0.097	0.379 ± 0.110	0.361 ± 0.115
Chefer LRP	0.294 ± 0.035	0.528 ± 0.085	0.484 ± 0.101	0.464 ± 0.105	0.516 ± 0.105
KernelSHAP	0.614 ± 0.024	0.593 ± 0.013	0.522 ± 0.008	0.462 ± 0.017	0.430 ± 0.022
LOO resample (oracle)	0.316 ± 0.017	0.031 ± 0.017	0.006 ± 0.008	0.004 ± 0.008	0.004 ± 0.008

relevant set (oracle FPM 0.005 to 0.034), so the residual disagreement is rank noise among near-zero effects rather than hidden shortcuts. Sink mass stays small: at most 0.050 ± 0.027 for raw attention, and 0.148 ± 0.061 across all fourteen rules and four tasks (Chefer on retrieval). We report it rather than counting it as error.

Seed variance limits what rankings can say. Between-seed standard deviations on grokked modular models reach 0.35 for raw attention and 0.36 for GradCAM, larger than most gaps between methods on that task. The paired tests sharpen this into a statement about resolution. Best method against random is decisive everywhere (paired $p < 10^{-7}$). Best against second-best splits: significant on retrieval ($p = 1.2 \times 10^{-5}$) and modular addition ($p = 2.2 \times 10^{-3}$), but not on parity (saliency against integrated gradients, $p = 0.52$) and marginal on recall (Chefer against gradient$\times$input, $p = 0.06$). With 8 to 11 seeds of paired data the top of the ranking is not resolvable on half the tasks, and a single-seed comparison among mid-ranked methods is a point estimate of a quantity whose seed-to-seed swing exceeds the differences being claimed.

6.1 Is high FPM misidentification or just diffuseness?

FPM mixes two failure modes. A method can put mass on the wrong tokens, or it can rank every token correctly and still score badly by spreading well-ranked mass too thinly. A method assigning 0.15 to each of three relevant positions and 0.015 to each of 37 irrelevant ones has identified everything and still records FPM 0.55. Since the headline claim rests on FPM, the two modes must be separated.

We separate them with a rank-preserving sharpening transform: raise each example's normalized scores to a power γ and renormalize. This leaves the ranking of every token untouched and concentrates mass on the top-ranked ones, so as γ grows FPM converges to the error of the tokens the method ranks highest. If high FPM were only diffuseness, sharpening would drive it toward zero. We also report the normalized entropy of each score distribution as a direct diffuseness measure.

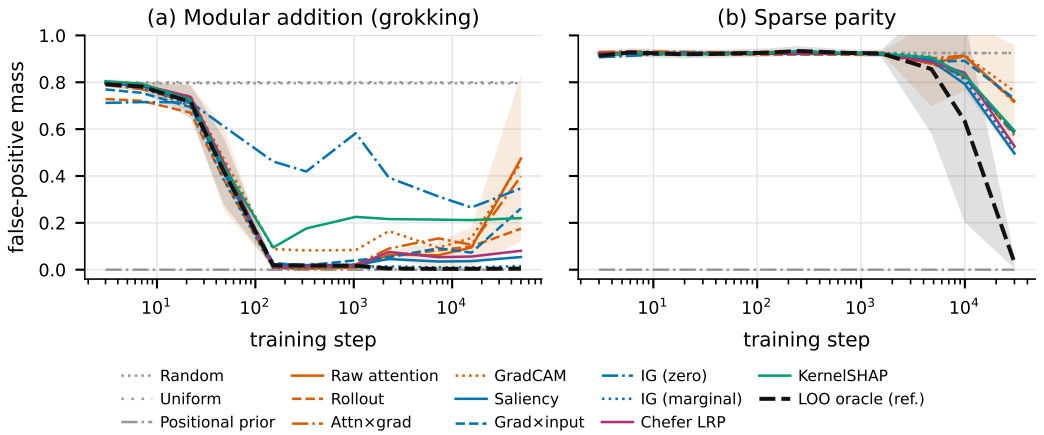

Figure 2: FPM across training (mean over seeds, log-scale steps; shaded bands show ± 1 between-seed SD for raw attention and for the oracle). Colour encodes method family and line style distinguishes members within a family. (a) On modular addition, attention-family FPM collapses to about 0.01 during memorization, then rises to between 0.2 and 0.5 through the grokking transition while held-out accuracy climbs to 100%. (b) On parity every method sits at the random floor until the feature-learning transition, then improves only partially; the oracle drops to about 0.03.

The parity verdict is misidentification. No method drops below 0.306 at $\gamma{=}8$ while the oracle falls from 0.031 to 0.004, a $76\times$ gap. Three methods get *worse* under sharpening (GradCAM $0.762 \to 0.907$, gradient$\times$input $0.729 \to 0.817$, attention$\times$gradient at $\gamma{=}8$), meaning the tokens they rank first are wrong more often than their average mass placement. The tokens these methods trust most are the least reliable part of their output, and no monotone recalibration can fix that. On the other tasks diffuseness does explain part of the moderate FPM values: attention rollout on modular addition falls from 0.175 to 0.000 under sharpening, saliency on retrieval from 0.292 to 0.000, and the positional prior on recall from 0.583 to 0.000, so those methods identify the right tokens and spread mass over them loosely. Both numbers matter and they answer different questions. Plain FPM is what a reader of the heatmap experiences; sharpened FPM isolates identification. The headline parity claim survives under both.

7 Faithfulness across training

This is the result we consider most consequential, because it is the one nobody could call expected. Figure 2 runs the full suite at 12 checkpoints per seed through two phase transitions.

On modular addition, most methods are nearly perfect by FPM *during the memorization phase*: the pre-grokking model attends narrowly to the operands and the explanations mirror that. Then, while held-out accuracy climbs from chance to 100%, the FPM of attention-family methods and gradient$\times$input climbs too, from about 0.01 to between 0.2 and 0.5, with the between-seed spread growing alongside. The oracle and integrated gradients with a marginal baseline stay low throughout, so the model never stops depending almost exclusively on the operands. What changes is how that dependence is represented internally. Nanda et al. (2023) showed that the grokked solution to modular addition implements a distributed Fourier-space algorithm: the network embeds operands as rotations at several frequencies, combines them through trigonometric identities across many neurons and heads, and reads the answer off the summed signal. Attribution then faces a representation whose relevant computation is spread across frequencies and components rather than concentrated on the operand positions in any single attention pattern. Attention maps and input-gradient products track the memorizing solution, which does look up operands directly, and lose the generalizing one.

Two practical consequences follow. First, faithfulness evaluated at a single checkpoint is not a property of a training run: on these tasks it can swing by a factor of 30 depending on when you look, and the swing itself varies by seed. Second, the intuition that cleaner, generalizing models should

be easier to explain is backwards here, at least for attention-based methods. If this pattern holds beyond algorithmic tasks, attribution audits of heavily trained models are operating at the hard end of the difficulty range, not the easy one. On parity the story is simpler: no method conveys anything before the phase transition, all sitting at the 0.925 floor while the model itself is at chance, and after the transition the gap between the oracle (0.03) and the best method (0.50) appears as a persistent plateau rather than a slow decay. Waiting longer does not close it.

8 Discussion and limitations

Methods do beat random everywhere by more than seed variance, so the field clears its lowest bar. Above that bar the three headline findings are the training-dynamics reversal (explanations degrade as the model improves), the interaction penalty (per-token attribution loses to a cheap causal probe by a factor of sixteen where interactions dominate, and sharpening does not close the gap), and the resolution limit (top-of-ranking comparisons are not statistically resolvable on half the tasks even with paired data over 8 to 11 seeds).

Practical guidance. If a static relevant set is plausible for your task, a positional prior costs nothing and is hard to beat. If you use integrated gradients on token models, the baseline must be in-distribution and must be reported. Raw-attention readings can be trusted on retrieval-style circuits and are misleading on superposed arithmetic ones, and there is no way to know in advance which regime applies without a causal check, which costs a few forward passes per token and outperformed every attribution method in this study.

Limitations. The models are 2 to 3 layers on synthetic tasks. The benchmark certifies necessary conditions only: failure here is evidence against reliability at scale though not proof of it, and success here does not transfer automatically. The accuracy-threshold filter is a selection effect (§5). FPM depends on the normalization convention (absolute value, sum to one), which penalizes the negative evidence in signed attributions, and rank metrics are reported alongside for that reason. The dynamics analysis uses 128 examples per checkpoint, trading precision for coverage. Associative recall loses 2 of 10 seeds to the accuracy threshold, so its intervals are wider.

Reproducibility statement

The benchmark is fully synthetic and seeded, so no data download is required. The task definitions and their provably-irrelevant positions are specified in §3 and Table 1; the fixed attribution protocol, including layer choice, head aggregation, integrated-gradients baselines and step count, and score normalization, is stated in §4; the pre-registered accuracy threshold and seed accounting are in §5; and complete per-task hyperparameters are in Appendix A. The supplementary material contains the task generators, model, attribution harness, evaluation and aggregation code, the statistical tests, and the per-seed result files from which every table and figure in this paper is regenerated, together with a pinned environment specification. The test suite included there checks the property the benchmark depends on, namely that mutating a provably-irrelevant position never changes the target.

Ethics statement

This work uses only synthetic data generated from seeded random number generators, involves no human subjects, and raises no privacy or consent concerns. Its intended impact is on how practitioners interpret attribution methods used to audit deployed models. Our evidence indicates that widely used methods misattribute a large fraction of their mass under the most favourable conditions available, so we expect the main effect of this work to be increased and appropriate scepticism toward explanation-based assurances of model behaviour.

AI use statement

We used generative AI tools for writing assistance (drafting and editing prose in this paper) and for code assistance (implementing the task generators, attribution harness, and evaluation pipeline). We did not use generative AI tools to generate research ideas, experimental designs, or results, and we did not use them to produce or alter any reported numbers. Every number in this paper is the output

of experiments executed by the authors on the hardware described in §5. We have reviewed all AI-assisted work: AI-assisted code was verified against the unit-test suite included in the supplementary material, in particular the tests that check the provably-irrelevant-position property each task depends on, and its outputs were checked against the released per-seed result files. Bibliographic entries were verified against Crossref, the arXiv API, and publisher records. We take responsibility for the final content of this work, including all text, claims, and artifacts produced with the aid of generative AI.

REFERENCES

Samira Abnar and Willem Zuidema. Quantifying attention flow in transformers. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 4190–4197, 2020.

Julius Adebayo, Justin Gilmer, Michael Muelly, Ian Goodfellow, Moritz Hardt, and Been Kim. Sanity checks for saliency maps. In *Advances in Neural Information Processing Systems*, 2018.

Chirag Agarwal, Satyapriya Krishna, Eshika Saxena, Martin Pawelczyk, Nari Johnson, Isha Puri, Marinka Zitnik, and Himabindu Lakkaraju. Openxai: Towards a transparent evaluation of model explanations. In *Advances in Neural Information Processing Systems, Datasets and Benchmarks Track*, 2022.

Boaz Barak, Benjamin L. Edelman, Surbhi Goel, Sham Kakade, Eran Malach, and Cyril Zhang. Hidden progress in deep learning: Sgd learns parities near the computational limit. In *Advances in Neural Information Processing Systems*, 2022.

Hila Chefer, Shir Gur, and Lior Wolf. Transformer interpretability beyond attention visualization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 782–791, 2021.

Rohan Gupta, Iván Arcuschin, Thomas Kwa, and Adrià Garriga-Alonso. Interpbench: Semi-synthetic transformers for evaluating mechanistic interpretability techniques. In *Advances in Neural Information Processing Systems, Datasets and Benchmarks Track*, 2024. arXiv:2407.14494.

Peter Hase, Harry Xie, and Mohit Bansal. The out-of-distribution problem in explainability and search methods for feature importance explanations. In *Advances in Neural Information Processing Systems*, 2021.

Sara Hooker, Dumitru Erhan, Pieter-Jan Kindermans, and Been Kim. A benchmark for interpretability methods in deep neural networks. In *Advances in Neural Information Processing Systems*, 2019.

Sarthak Jain and Byron C. Wallace. Attention is not explanation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics*, pp. 3543–3556, 2019.

Joseph D. Janizek, Pascal Sturmfels, and Su-In Lee. Explaining explanations: Axiomatic feature interactions for deep networks. *Journal of Machine Learning Research*, 22(104):1–54, 2021.

Narine Kokhlikyan, Vivek Miglani, Miguel Martin, Edward Wang, Bilal Alsallakh, Jonathan Reynolds, Alexander Melnikov, Natalia Kliushkina, Carlos Araya, Siqi Yan, and Orion Reblitz-Richardson. Captum: A unified and generic model interpretability library for pytorch. *arXiv preprint arXiv:2009.07896*, 2020.

Jun Rui Lee, Sadegh Emami, Michael David Hollins, Timothy C. H. Wong, Carlos Ignacio Villalobos Sánchez, Francesca Toni, Dekai Zhang, and Adam Dejl. Xai-units: Benchmarking explainability methods with unit tests. In *Proceedings of the 2025 ACM Conference on Fairness, Accountability, and Transparency*, 2025. arXiv:2506.01059.

Xuhong Li, Mengnan Du, Jiamin Chen, Yekun Chai, Himabindu Lakkaraju, and Haoyi Xiong. M4: A unified xai benchmark for faithfulness evaluation of feature attribution methods across metrics, modalities and models. In *Advances in Neural Information Processing Systems, Datasets and Benchmarks Track*, 2023.

David Lindner, János Kramár, Sebastian Farquhar, Matthew Rahtz, Thomas McGrath, and Vladimir Mikulik. Tracr: Compiled transformers as a laboratory for interpretability. In *Advances in Neural Information Processing Systems*, 2023.

Yibing Liu, Haoliang Li, Yangyang Guo, Chenqi Kong, Jing Li, and Shiqi Wang. Rethinking attention-model explainability through faithfulness violation test. In *International Conference on Machine Learning*, pp. 13807–13824, 2022.

Scott M. Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems*, 2017.

Neel Nanda, Lawrence Chan, Tom Lieberum, Jess Smith, and Jacob Steinhardt. Progress measures for grokking via mechanistic interpretability. In *International Conference on Learning Representations*, 2023.

Catherine Olsson, Nelson Elhage, Neel Nanda, Nicholas Joseph, Nova DasSarma, Tom Henighan, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, et al. In-context learning and induction heads. Transformer Circuits Thread, 2022. `https://transformer-circuits.pub/2022/in-context-learning-and-induction-heads/index.html`.

Alethea Power, Yuri Burda, Harri Edwards, Igor Babuschkin, and Vedant Misra. Grokking: Generalization beyond overfitting on small algorithmic datasets. *arXiv preprint arXiv:2201.02177*, 2022.

Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 618–626, 2017.

Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. Learning important features through propagating activation differences. In *International Conference on Machine Learning*, pp. 3145–3153, 2017.

Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*, 2014.

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *International Conference on Machine Learning*, pp. 3319–3328, 2017.

Mukund Sundararajan, Kedar Dhamdhere, and Ashish Agarwal. The shapley taylor interaction index. In *International Conference on Machine Learning*, pp. 9259–9268, 2020.

Michael Tsang, Sirisha Rambhatla, and Yan Liu. How does this interaction affect me? interpretable attribution for feature interactions. In *Advances in Neural Information Processing Systems*, 2020.

Sarah Wiegreffe and Yuval Pinter. Attention is not not explanation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*, pp. 11–20, 2019.

Junyi Wu, Weitai Kang, Hao Tang, Yuan Hong, and Yan Yan. On the faithfulness of vision transformer explanations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10936–10945, 2024.

Guangxuan Xiao, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis. Efficient streaming language models with attention sinks. In *International Conference on Learning Representations*, 2024. arXiv:2309.17453.

A PER-TASK TRAINING DETAILS

Parity: 2 layers, d_{model}=128, learning rate 10^{-3}, weight decay 0, batch 1024, 30k steps, 20 seeds. Retrieval: same architecture, weight decay 10^{-2}, batch 256, 20k steps, 10 seeds. Recall: 3 layers, d_{model}=256, learning rate 3×10^{-4}, weight decay 10^{-2}, batch 256, 60k steps, 10 seeds. Modular addition: 2 layers, d_{model}=128, learning rate 10^{-3}, weight decay 1.0, batch 256, 40% train split, 50k steps, 10 seeds. All runs use AdamW and a log-spaced checkpoint schedule. Failed-seed counts appear in §5, and failed seeds are released with the zoo.

B ADDITIONAL METRIC TABLES AND CAUSAL VALIDATION

Table 6: Model-level validation: Spearman correlation between attribution scores and measured resample-ablation effects (mean $\pm$ 95% t-CI over seeds).

Method	Parity	Retrieval	Recall	Modular
Random	0.006 ± 0.006	-0.007 ± 0.007	0.007 ± 0.004	0.012 ± 0.006
Uniform	0.000 ± 0.000	0.000 ± 0.000	0.000 ± 0.000	0.000 ± 0.000
Positional prior	0.464 ± 0.019	0.202 ± 0.023	0.139 ± 0.041	0.696 ± 0.000
Raw attention (last)	0.135 ± 0.038	0.111 ± 0.048	0.191 ± 0.147	0.235 ± 0.267
Attention rollout	0.534 ± 0.053	0.538 ± 0.020	0.601 ± 0.080	0.518 ± 0.024
Attention$\times$grad	0.137 ± 0.038	0.116 ± 0.044	0.186 ± 0.147	0.335 ± 0.132
GradCAM	0.088 ± 0.027	0.093 ± 0.038	0.120 ± 0.086	0.241 ± 0.223
Saliency (ℓ_2)	$\mathbf{0.544 \pm 0.064}$	$\mathbf{0.586 \pm 0.016}$	$\mathbf{0.721 \pm 0.067}$	$\mathbf{0.563 \pm 0.026}$
Grad$\times$input	0.388 ± 0.090	0.105 ± 0.045	0.295 ± 0.088	0.400 ± 0.116
IG (zero base)	0.287 ± 0.041	0.311 ± 0.053	0.359 ± 0.064	0.404 ± 0.068
IG (marginal base)	0.442 ± 0.042	0.440 ± 0.014	0.538 ± 0.054	0.546 ± 0.026
Chefer LRP	0.322 ± 0.060	0.522 ± 0.021	0.385 ± 0.088	0.509 ± 0.018
KernelSHAP	0.115 ± 0.007	0.274 ± 0.007	0.366 ± 0.008	0.440 ± 0.008
LOO resample (oracle)	0.960 ± 0.018	0.796 ± 0.007	0.893 ± 0.015	0.729 ± 0.006