A Training-Time Sign Flip During the Formation of an Indirect Object Identification Circuit

Tejas Dahiya [1] **Cole Blondin** [2]

Abstract

Mechanistic analyses often describe how a trained model produces a behavior at a single checkpoint. Yet the causal effect of intervening at the same position can change, and even reverse, as the model learns. We show this in indirect object identification, where a model should choose the name mentioned once rather than the name repeated twice. During training, models briefly prefer the repeated name and fall below 50 percent accuracy on this two-name choice, even as language modeling loss decreases at every sampled checkpoint. At each checkpoint, we replace the residual-stream activations at the repeated name's second mention with activations from an otherwise identical prompt with a new name. In Pythia-160M, this intervention raises the correct name's logit relative to the repeated name by 0.95 at the lowest-accuracy checkpoint, but lowers it by 4.12 at maturity. The same positive-to-negative reversal appears across six Pythia scales from 160 million to 12 billion, nine runs varying seed, data order, or initialization, and Stanford GPT-2 Small. A held-out linear probe decodes whether the name is repeated with about 61 percent accuracy in the window and 99 percent at maturity. Removing the probe direction, however, does not reduce this preference. Meanwhile, the effect of mean-ablating a head that later suppresses the repeated name is near zero during the window and grows during recovery. Readable information, the effect of changing S2, and the contribution of a specific attention head therefore develop on different timelines. In this case, analyzing only the final checkpoint does not merely miss how the behavior developed. It gives the wrong sign for how the same intervention affects behavior earlier in training.

[1]University of Wisconsin–Madison, Madison, WI, USA [2]Independent Researcher. Correspondence to: Tejas Dahiya <tejasdahiya0007@gmail.com>.

Mechanistic Interpretability Workshop at the 43rd International Conference on Machine Learning, Seoul, South Korea, 2026.
Copyright 2026 by the author(s).

1. Introduction

Mechanistic interpretability often explains a model by analyzing how it produces a behavior at a single trained checkpoint. Yet the circuits supporting that behavior develop throughout training, and their causal effects can change as learning proceeds. We study a case in which the effect of the same intervention procedure reverses sign. It improves behavior at an early checkpoint and harms behavior after recovery. A circuit analysis based only on the final model therefore predicts the wrong direction of that intervention earlier in training.

We examine this reversal in indirect object identification, or IOI, a controlled name-tracking task. Given the sentence "When Mary and John went to the store, John gave a drink to", the model should continue with Mary, the name mentioned once, instead of John, the name repeated twice. We call the second occurrence of the repeated name S2. We evaluate the model using the IO-minus-S logit difference, defined as the logit of the correct name minus the logit of the repeated name. Positive values favor the correct name, while negative values favor the repeated name. Accuracy is the fraction of prompts with a positive logit difference, so chance performance on this two-name comparison is 50 percent. This score is separate from the model's highest-ranked prediction over the full vocabulary.

IOI performance changes non-monotonically during training. Pythia-160M briefly falls to 31.7 percent accuracy at step 2000 before recovering to near-perfect performance, while its language modeling loss decreases at every sampled checkpoint over the same period. We test how changing the activations at S2 affects the model's output using a matched intervention. For each prompt, we construct an otherwise identical control in which the second occurrence of the repeated name is replaced by a new name. We then replace the residual-stream activations at S2 with the corresponding activations from the control prompt. The residual stream is the model's running internal representation at each token position. At the lowest-accuracy checkpoint, this intervention raises the correct name's logit relative to the repeated name by 0.95. At the final checkpoint, the same procedure lowers it by 4.12. Because each checkpoint supplies its own matched control activations, the quantity compared across

training is the effect of a fixed intervention procedure. That effect changes from positive to negative.

The reversal is robust across models and training runs. It appears at all six Pythia scales from 160 million to 12 billion parameters, in all nine PolyPythias variants that change random seed, data order, or initialization, and in Stanford GPT-2 Small. The below-chance window appears across the same model scales and training variants. These replications show that the phenomenon is not tied to one model size, checkpoint series, or training configuration.

We use three measurements to understand how the developing mechanism relates to this reversal. A linear probe, a simple classifier trained on hidden states, tests whether repetition can be decoded from the state at S2. Activation replacement tests whether the model's prediction depends causally on that position. Mean ablation replaces an attention head's output with its average over the evaluation set and measures whether that component supports the behavior. These measurements follow different trajectories across training. The linear probe reaches about 61 percent accuracy during the window and 99 percent at maturity, yet removing the probe direction does not reduce the model's preference for the repeated name. Meanwhile, mean-ablating a head that later suppresses the repeated name has almost no effect during the window, and its effect grows during recovery, which places its contribution in the recovery phase rather than the below-chance window.

Together, these results show that the effect of a causal intervention can reverse sign during training. In the case studied here, analyzing only the final checkpoint gives the wrong sign for how the same intervention affects behavior earlier in training.

2. Related Work

Wang et al. (2023) reverse engineered the indirect object identification circuit in GPT-2 Small and identified functional components including name mover heads, which promote the correct name, and suppressor heads, which act against the repeated name. Subsequent work has developed methods for discovering circuits automatically (Conmy et al., 2023) and formalized how attention heads compose through the residual stream (Elhage et al., 2021). These studies primarily characterize mechanisms at trained checkpoints.

A growing body of work instead examines how internal representations and mechanisms emerge during training. Induction heads form at identifiable stages of learning (Olsson et al., 2022), and changes in internal representations can precede changes in behavior (Nanda et al., 2023). Most directly, Tigges et al. (2024) track IOI circuits across Pythia checkpoints and model scales. They find that task performance is generally non-decreasing after acquisition, while also observing that task scores can become negative shortly before the behavior is learned. We focus on this pre-recovery regime. We show that the repeated name receives a higher mean probability and better average vocabulary rank than the correct name, and we identify an intervention at the repeated name's second occurrence whose effect changes from positive to negative across recovery. Where their negative task scores are a behavioral observation, this sign change is a causal one.

This phenomenon differs from two related patterns in learning curves. Grokking describes delayed generalization after training performance has already saturated (Power et al., 2022; Nanda et al., 2023). Apparent jumps in capability across model scales can also arise from thresholded evaluation metrics rather than discontinuous changes in the model (Wei et al., 2022; Schaeffer et al., 2023). Here, language modeling loss decreases at every sampled checkpoint through the low-accuracy window, while the reversal appears directly in the continuous signed logit difference. Methodologically, we combine linear probes (Alain & Bengio, 2017; Belinkov, 2022), activation replacement (Vig et al., 2020; Meng et al., 2022; Goldowsky-Dill et al., 2023; Zhang & Nanda, 2024), and head ablation (Wang et al., 2023) across checkpoints to distinguish linear decodability, position-level causal dependence, and component-level function.

3. Experimental Setup

Task and metric. We study indirect object identification using 300 prompts drawn from ten BABA-style IOI template families (Wang et al., 2023), with 30 prompts per family and both assignments of each name pair to the correct and repeated roles. All names consist of a single token. We call the name mentioned once the indirect object, or IO, and the name mentioned twice the repeated subject, or S. Its second occurrence is S2. Performance is measured by the IO-minus-S logit difference, defined as the correct name's logit minus the repeated name's logit. Accuracy is the fraction of prompts with a positive logit difference, so chance on this two-name comparison is 50 percent.

Models. We analyze six Pythia scales from 160 million to 12 billion parameters (Biderman et al., 2023), nine PolyPythias 160M variants that vary random seed, data order, or initialization (van der Wal et al., 2025), and Stanford GPT-2 Small (Karamcheti et al., 2021). Most detailed analyses use Pythia-160M. Additional protocol details and the experiments performed for each model are reported in Appendices D through G and I.

Matched S2 intervention. For each prompt, we construct an otherwise identical control in which the repeated name

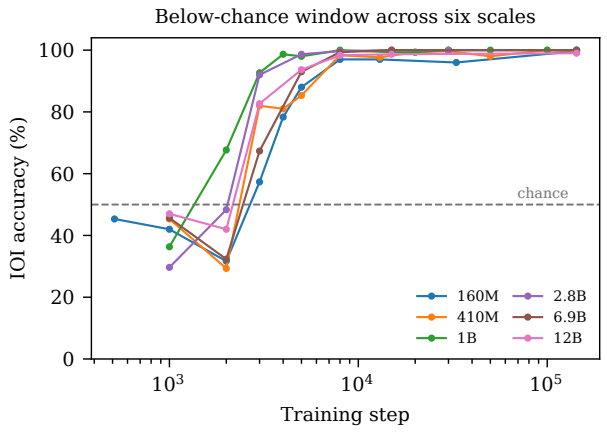

Figure 1. IOI accuracy across training for six Pythia scales on 300 prompts from ten BABA-style template families. Every model falls below the two-name chance level of 50 percent before recovering, with minimum accuracies ranging from 29.3 percent for Pythia-410M to 42.0 percent for Pythia-12B.

at S2 is replaced by a third, unrelated name. We replace the residual-stream activations at S2 with the corresponding activations from the control prompt over a fixed middle-layer window. The intervention effect is the change in IO-minus-S logit difference after replacement. Positive values move the model toward the correct name, while negative values move it toward the repeated name. The procedure is fixed across training, while each checkpoint supplies its own matched control activations. Exact layer windows are listed in Appendix E.

Component and representation analyses. We zero-ablate individual attention heads to test whether the early preference can be localized to one head. To trace suppressor function, we select heads by their mature mean-ablation effect, fix their identities, and apply the same ablation across earlier checkpoints. Held-out checks for this selection procedure are described in Appendix D. We also train logistic-regression probes to predict whether S2 contains a repeated name using disjoint names and template families. Position-shuffle and projection-removal controls are reported in Appendices C and H.

Uncertainty. Unless stated otherwise, we report 95 percent bootstrap intervals over prompts. For the primary Pythia-160M intervention, we additionally resample template families and name pairs to account for dependence among related prompts.

4. A Transient Preference for the Repeated Name

During the below-chance window (Figure 1), the model usually predicts neither candidate as its most likely next token. At the Pythia-160M floor, the highest-ranked token is neither name on 98.67 percent of prompts. When the two names are compared directly, the repeated name nevertheless scores higher, with a higher mean probability, 0.0238 against 0.0118, and a better average rank, 14.9 against 29.4, where lower rank is closer to the top of the vocabulary.

The same preference appears in the continuous IO-minus-S logit difference. It remains negative across a contiguous range of checkpoints, with bootstrap intervals excluding zero. The window therefore reflects a sustained change in a signed quantity rather than an artifact of thresholding a close score into correct or incorrect predictions. Over the same interval, the language modeling loss on the fixed Pile sample (Gao et al., 2020) decreases at every evaluated checkpoint, leaving the window absent from the sampled loss trajectory (Appendix A). We next examine how the causal effect of intervening at S2 changes as the model recovers.

5. The Effect of the S2 Replacement Reverses

At each checkpoint, we replace the residual-stream activations at S2 with those produced by a matched control prompt that differs only by replacing the repeated name at S2 with an unrelated name. This replacement helps during the low-accuracy window and harms after recovery (Figure 2, center). In Pythia-160M, it raises the IO-minus-S logit difference by 0.95 at step 2000, with a 95 percent interval of 0.68 to 1.25 when entire template families are resampled. Its estimated effect is 0.02 at step 3000, where even the prompt-level interval spans zero, and it becomes negative by step 4000. At the final checkpoint, the replacement changes the logit difference by -4.12, with a template-clustered interval of -4.54 to -3.72. Resampling by name pair gives the same positive sign during the window and negative sign at maturity. The effect of the same replacement procedure therefore changes from helping the model choose the correct name to pushing it toward the repeated name as training proceeds.

The same positive-to-negative pattern appears at all six Pythia scales from 160 million to 12 billion parameters, in all nine PolyPythias runs that vary random seed, data order, or initialization, and in Stanford GPT-2 Small (Appendices E, F, and G).

Control interventions show that this change in sign is specific to S2 among the positions we test. Both the matched replacement and a norm-matched random replacement at S2 help during the window and harm at maturity. Replacing the first occurrence of the repeated name helps at both stages, while replacing the correct name or a structural position harms at both stages. Some of these interventions have larger effects than the S2 intervention, so the distinctive result is the reversal across training rather than the size of

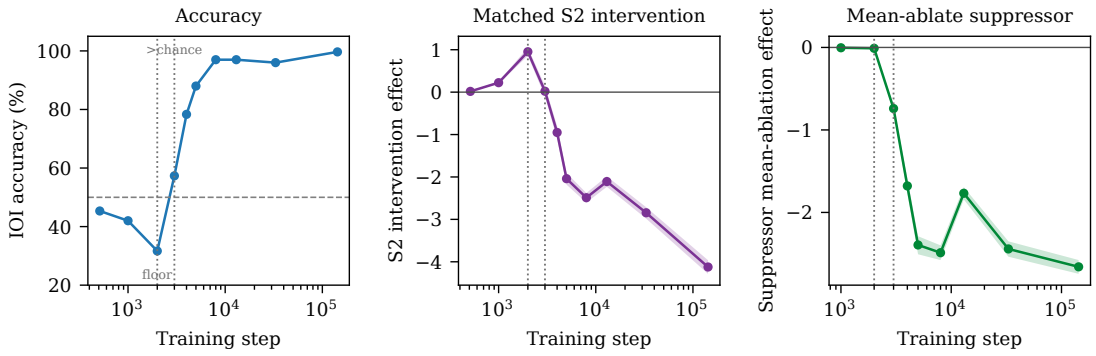

Figure 2. Three measurements on Pythia-160M across training, sharing the training step axis, with the transition window from step 2000 to 3000 shaded. Left, IOI accuracy with the chance line. Center, the effect of the matched S2 intervention on the logit difference, with a 95 percent bootstrap band; a positive value means the intervention moves the model toward the correct name. Right, the effect of mean-ablating the mature suppressor head, where a negative value means removing the head lowers the logit difference, so the head helps the correct name. The center and right panels use opposite sign conventions. At the floor the S2 intervention helps and mean-ablating the suppressor has little effect. After recovery the S2 intervention harms and mean-ablating the suppressor is increasingly damaging. These changes unfold over the broader recovery period but do not by themselves establish that suppressor formation causes the crossover.

Table 1. Effect of replacing four positions on the logit difference, during the window (step 2000) and at maturity (final checkpoint), on Pythia-160M. The random arms are norm-matched random replacements at the named position. A positive value moves the model toward the correct name. Only the two S2 replacements change sign across training. The full battery with intervals is in Appendix B.

Position	Window	Maturity	Reverses
S2 matched	$+0.95$	-4.12	yes
S2 random	$+0.92$	-4.13	yes
S1 random	$+2.15$	$+2.08$	no
Correct-name random	-3.54	-9.06	no
Structural random	-0.81	-4.45	no

the effect at one checkpoint. The random S2 result also shows that the reversal is not unique to the exact activations copied from the matched control prompt. Table 1 reports the full comparison.

A separate input-level test gives the same early direction. Replacing the repeated name at S2 with a new name raises the logit difference by 0.833, while changing an unrelated filler token changes it by only -0.082. Using a different replacement name produces nearly the same improvement. Because this test changes the sentence itself rather than the model's activations, it provides complementary evidence that repetition at S2 contributes to the early preference for the wrong name. Full details are given in Appendix B.

These results establish a reliable reversal in the effect of intervening at S2. They do not yet identify which model components produce the early preference for the repeated name. We next examine how the contribution of attention heads changes during recovery.

6. Suppressor Function Develops During Recovery

At step 1000, zeroing any one of Pythia-160M's 144 attention heads changes the mean IO-minus-S logit difference by at most 0.071. This test therefore does not attribute the early preference for the repeated name to any single attention head.

A strong suppressor is present at maturity. In Pythia-160M, L8H9 is the head whose mean ablation most harms mature IOI performance. It attends to S2 and lowers the repeated name's logit much more than the correct name's logit. Mean-ablating L8H9 at maturity lowers the IO-minus-S logit difference by 2.66, the largest mature mean-ablation effect among the model's attention heads. Stanford GPT-2 Small also contains a mature head with the same suppressor behavior.

Holding L8H9 fixed and tracing it backward through training shows that its contribution develops during recovery. Mean-ablating it changes the logit difference by only -0.01 at step 2000, but by -0.74 at step 3000 and -2.39 at step 5000. A negative value means that removing the head harms the correct-name preference. L8H9 therefore contributes almost nothing during the low-accuracy window and becomes increasingly important as performance recovers (Figure 2, right). The mature-selected suppressor in Pythia-410M follows the same pattern.

Because L8H9 was selected using its mature effect, we repeat the timing analysis on an independently trained Pythia-160M model using held-out data. A fixed set of mature suppressors is not yet helping the correct name at step 2000 but clearly helps by step 6000. Full methods and intervals are given in Appendix D.

The head's attention to S2 and its output against the repeated name also grow during recovery. These results are consistent with the suppressor contributing to recovery, but they do not show that its development causes the S2 intervention to reverse. Harmful dependence on S2 is already present before the suppressor becomes strong, and we have not identified a single causal path connecting the two changes.

7. Reading Repetition Is Not the Same as Using It

We train a simple linear classifier on the residual-stream activations at S2 to predict whether the name at that position has appeared earlier in the prompt. To test whether this information generalizes, the classifier is trained on one set of names and templates and evaluated on a disjoint set. Its held-out accuracy is about 44 percent at initialization, 61 percent during the low-accuracy window, and 99 percent at maturity. Because the meanings of the classifier's two outputs remain fixed after training, its below-chance initialization score does not prove that no information is present. It shows that the rule learned on one split does not transfer reliably to the other. The repetition signal is therefore only modestly readable during the window and becomes nearly perfectly readable later. Selected layer results are given in Appendix C.

A separate control shows why the held-out evaluation matters. When training and evaluation use the same name and template distribution, probe accuracy at initialization is already about 88 to 90 percent. Shuffling the token positions reduces it by roughly 6 to 10 percentage points, showing that fixed positional structure accounts for part of this high initial score. This control is separate from the held-out probe above and is reported in Appendix C.

The ability to read repetition does not show that the model's preference depends on the particular direction used by the classifier. At step 2000, we remove from the layer-5 S2 activations the component along the classifier's learned direction. This does not reduce the preference for the repeated name. Instead, it lowers the IO-minus-S logit difference by approximately 0.11, slightly strengthening the preference. Removing orthogonal, shuffled-label, or random directions has little effect. The specific linear direction found by this probe is therefore not sufficient to explain the repeated-name preference. This result does not rule out other representations of repetition or other ways that repetition information may affect the output. Full controls are given in Appendix H.

The ability of a classifier to read repetition, the behavioral effect of replacing the S2 activations, and the contribution of the suppressor head therefore change on different schedules. A signal can be readable without the tested direction being responsible for the behavior, and a position can already affect the output before the mature task-solving component becomes strong.

8. Discussion

The central finding is that the effect of the same S2 replacement procedure changes sign during training. Replacing the residual-stream activations at S2 with those from a matched non-repeating prompt helps during the below-chance window and harms after recovery. A description of S2 that is accurate for the mature model therefore does not describe its earlier behavior. In this case, analyzing only the final checkpoint does not merely miss how the behavior developed. It gives the wrong sign for how the same intervention affects behavior earlier in training.

The three measurements change on different timelines and should not be treated as one event. The repetition signal becomes increasingly readable, the S2 replacement changes sign, and suppressor function grows during recovery. The suppressor may contribute to recovery, but its timing does not establish that it causes the S2 replacement effect to reverse.

Our claims are deliberately narrow. Among the positions tested, only interventions at S2 reverse sign, but a norm-matched random replacement at S2 also reverses. The result is therefore not unique to the activations copied from the matched control prompt. Removing the particular direction learned by the repetition probe also does not reduce the repeated-name preference, so that direction is not sufficient to explain the behavior. We do not identify the complete source of the early preference, and zeroing individual attention heads does not localize it to one head.

Each checkpoint supplies its own matched control activations, so we do not claim that one fixed activation vector changes its role. The position-level reversal replicates across six Pythia scales, nine PolyPythias training variants, and Stanford GPT-2 Small, while the training-time suppressor trajectory is studied only in Pythia-160M and Pythia-410M. Finally, we study one English task in decoder-only language models. Whether similar sign-changing dependencies appear in other tasks and circuits remains open.

Software and Data

The public repository at https://github.com/Tejas7007/ICML_2026_MIW_IOI_Sign_Flip contains the code, results, and scripts to reproduce and verify the analyses. The dense Pythia-160M checkpoints are at https://huggingface.co/teys7007/pythia-160m-seed42-dense.

References

Alain, G. and Bengio, Y. Understanding intermediate layers using linear classifier probes. In *ICLR Workshop Track*, 2017.

Belinkov, Y. Probing classifiers: Promises, shortcomings, and advances. *Computational Linguistics*, 48(1):207–219, 2022.

Biderman, S., Schoelkopf, H., Anthony, Q., Bradley, H., O'Brien, K., Hallahan, E., Khan, M. A., Purohit, S., Prashanth, U. S., Raff, E., Skowron, A., Sutawika, L., and van der Wal, O. Pythia: A suite for analyzing large language models across training and scaling. In *International Conference on Machine Learning (ICML)*, 2023.

Conmy, A., Mavor-Parker, A. N., Lynch, A., Heimersheim, S., and Garriga-Alonso, A. Towards automated circuit discovery for mechanistic interpretability. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Elhage, N., Nanda, N., Olsson, C., Henighan, T., Joseph, N., Mann, B., Askell, A., Bai, Y., Chen, A., Conerly, T., et al. A mathematical framework for transformer circuits. *Transformer Circuits Thread*, 2021.

Gao, L., Biderman, S., Black, S., Golding, L., Hoppe, T., Foster, C., Phang, J., He, H., Thite, A., Nabeshima, N., Presser, S., and Leahy, C. The Pile: An 800GB dataset of diverse text for language modeling. *arXiv preprint arXiv:2101.00027*, 2020.

Goldowsky-Dill, N., MacLeod, C., Sato, L., and Arora, A. Localizing model behavior with path patching. *arXiv preprint arXiv:2304.05969*, 2023.

Karamcheti, S., Orr, L., Bolton, J., Zhang, T., Goel, K., Narayan, A., Bommasani, R., Narayanan, D., Hashimoto, T., Jurafsky, D., Manning, C. D., Potts, C., Ré, C., and Liang, P. Mistral: A journey towards reproducible language model training. Technical report, Stanford CRFM, 2021.

Meng, K., Bau, D., Andonian, A., and Belinkov, Y. Locating and editing factual associations in GPT. *Advances in Neural Information Processing Systems (NeurIPS)*, 35, 2022.

Nanda, N., Chan, L., Lieberum, T., Smith, J., and Steinhardt, J. Progress measures for grokking via mechanistic interpretability. In *International Conference on Learning Representations (ICLR)*, 2023.

Olsson, C., Elhage, N., Nanda, N., Joseph, N., DasSarma, N., Henighan, T., et al. In-context learning and induction heads. *Transformer Circuits Thread*, 2022.

Power, A., Burda, Y., Edwards, H., Babuschkin, I., and Misra, V. Grokking: Generalization beyond overfitting on small algorithmic datasets. *arXiv preprint arXiv:2201.02177*, 2022.

Schaeffer, R., Miranda, B., and Koyejo, S. Are emergent abilities of large language models a mirage? *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Tigges, C., Hanna, M., Yu, Q., and Biderman, S. LLM circuit analyses are consistent across training and scale. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

van der Wal, O., Lesci, P., Müller-Eberstein, M., Saphra, N., Schoelkopf, H., Zuidema, W., and Biderman, S. PolyPythias: Stability and outliers across fifty language model pre-training runs. In *International Conference on Learning Representations (ICLR)*, 2025.

Vig, J., Gehrmann, S., Belinkov, Y., Qian, S., Nevo, D., Singer, Y., and Shieber, S. Investigating gender bias in language models using causal mediation analysis. *Advances in Neural Information Processing Systems (NeurIPS)*, 33, 2020.

Wang, K., Variengien, A., Conmy, A., Shlegeris, B., and Steinhardt, J. Interpretability in the wild: a circuit for indirect object identification in GPT-2 small. In *International Conference on Learning Representations (ICLR)*, 2023.

Wei, J., Tay, Y., Bommasani, R., Raffel, C., Zoph, B., Borgeaud, S., Yogatama, D., Bosma, M., Zhou, D., Metzler, D., Chi, E. H., Hashimoto, T., Vinyals, O., Liang, P., Dean, J., and Fedus, W. Emergent abilities of large language models. *Transactions on Machine Learning Research (TMLR)*, 2022.

Zhang, F. and Nanda, N. Towards best practices of activation patching in language models: Metrics and methods. In *International Conference on Learning Representations (ICLR)*, 2024.

A. The Below-Chance Window and the Loss

Figure 3 plots the language modeling loss of Pythia-160M on a fixed sample from the Pile training stream (Gao et al., 2020) alongside its IOI accuracy across training, using the monology Pile uncopyrighted train split. The producing script skips the first 100,000 records of the stream, reads 100 candidate texts, keeps those with at least 512 tokens, truncates each to 512 tokens, and averages the causal language modeling loss over the first 50 qualifying sequences, at steps 1000, 2000, 3000, 5000, and 10000. The loss decreases at every sampled checkpoint over the same interval in which accuracy falls below chance and recovers, so the below-chance window is not apparent in the loss trajectory. This is a fixed sample from the training stream rather than a held-out validation set, so the claim is only that loss decreases at the sampled checkpoints.

Full-vocabulary behavior at the floor. At the Pythia-160M floor, step 2000, the highest-ranked token over the full vocabulary is neither candidate name on 98.67 percent of prompts. Comparing the two names directly, the repeated name has mean probability 0.0238 and mean rank 14.9, against 0.0118 and 29.4 for the correct name.

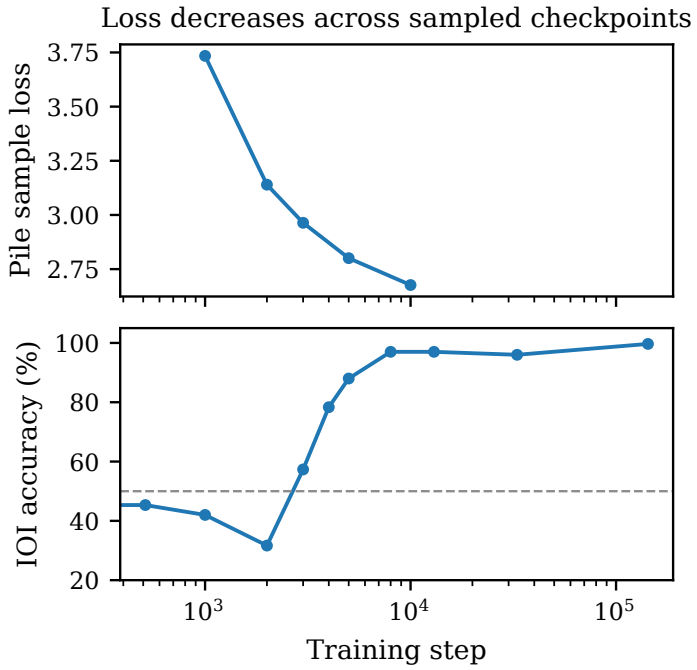

Figure 3. Pile sample loss and IOI accuracy for Pythia-160M across training, shown as two aligned panels. The loss decreases at every sampled checkpoint while accuracy dips below chance near step 2000 and recovers.

B. Full Control Battery and Input-Level Control

Table 2 reports every arm of the position battery with 95 percent bootstrap intervals, on Pythia-160M, at the window (step 2000) and at maturity. The two S2 arms are the matched name replacement and a norm-matched random replacement at the same position. The random arm replaces the residual-stream activation with an independently sampled Gaussian direction, normalized to the norm of the clean activation at the intervened position, sampled per prompt and layer with a seed tied to the checkpoint, over the same layer window as the matched replacement. The same construction is used at S1, the correct-name position, and the structural position. Among the tested positions, only the two S2 arms are significantly positive during the window and significantly negative at maturity. The other arms have effects that are often large but keep the same sign at both stages, so a change of sign is specific to the S2 position and does not follow from the norm-matched random replacements at the other tested positions.

Table 3 reports a complementary input-level control on a locked Pythia-160M benchmark of 800 prompts with a fixed prompt set, identified by a stored prompt hash, with a crossed bootstrap over template family and unordered name pair. Standard de-duplication replaces the second occurrence of the repeated name with a third, non-repeating single-token name, so the prompt no longer contains a repeat. Alternate de-duplication does the same with a different third name, which tests

Table 2. Full position battery on Pythia-160M, reporting the change in logit difference with 95 percent prompt-bootstrap intervals. Dependence-aware intervals for the primary matched-S2 arm are reported in Appendix E. A positive value moves the model toward the correct name, and only the S2 arms reverse sign between the window and maturity.

Arm	Window (step 2000)	Maturity (final)	Reverses sign
S2 matched name replacement	$+0.95$ $[+0.86, +1.05]$	-4.12 $[-4.28, -3.96]$	yes
S2 norm-matched random replacement	$+0.92$ $[+0.84, +1.00]$	-4.13 $[-4.31, -3.94]$	yes
S1 norm-matched random replacement	$+2.15$ $[+2.06, +2.24]$	$+2.08$ $[+1.88, +2.27]$	no
correct-name norm-matched random replacement	-3.54 $[-3.73, -3.34]$	-9.06 $[-9.31, -8.82]$	no
structural-position norm-matched random replacement	-0.81 $[-0.90, -0.73]$	-4.45 $[-4.67, -4.23]$	no

sensitivity to the chosen replacement. The filler placebo instead changes a neutral filler token, a place or adjective, leaving the repetition intact. De-duplicating the repeated name raises the logit difference, the alternate de-duplication gives an almost identical effect, and the placebo has an effect an order of magnitude smaller. The clean accuracy under this benchmark is 30.9 percent, with a cluster-aware interval of 23.7 to 38.3 percent.

Table 3. Input-level control at the window on the locked Pythia-160M benchmark of 800 prompts. Intervals use the crossed template-family and name-pair bootstrap where reported. The two de-duplications agree and the placebo effect is an order of magnitude smaller.

Intervention	Effect on logit difference	Cluster-aware interval or comparison
Standard de-duplication	$+0.833$	$[+0.681, +1.023]$
Alternate de-duplication	$+0.832$	agrees with standard
Filler placebo	-0.082	$[-0.159, -0.012]$

C. Probe Experiments

We report three separate probe experiments and keep them distinct.

Held-out generalization. The probe reads the residual-stream activations at S2 and predicts a binary label. The positive class is the S2 position in the original prompt, where the name is a repeat of the earlier subject. The negative class is the same position in a matched control, where that name is replaced by a third, non-repeating single-token name. The probe is trained on 600 examples drawn from one set of names and templates and evaluated on a disjoint set of names and templates, in two folds that swap the training and evaluation sets, and we report the mean over folds per layer without selecting a best layer on validation data. Table 4 reports held-out accuracy at two layers at three training steps. The held-out accuracy is below chance at initialization, about 61 percent at the window, and about 99 percent at maturity. Because the classifier's label orientation is fixed on the training split, below-chance held-out accuracy does not establish that no information is present. It shows that the learned rule does not transfer reliably in that orientation.

Table 4. Held-out probe accuracy for predicting whether the S2 position holds a repeated name or a non-repeating control name, read from the residual-stream activations at S2 on Pythia-160M. Values are the mean over the two folds. The classifier orientation is fixed on the training split, so the below chance value at initialization reflects failure to transfer in that orientation.

	Initialization	Window (step 2000)	Maturity (final)
Layer 1	0.436	0.613	0.992
Layer 5	0.443	0.616	0.994

Position shuffle. This is an in-distribution probe, trained and evaluated on the same names and templates, and is therefore not directly comparable to the held-out experiment above. It measures how much of the initial in-distribution accuracy comes from fixed architectural and positional structure. Before probing, within each template batch we apply one token-position permutation to both the repeated-name prompt and its control, updating the S2 index accordingly, then read the S2 activation at its shuffled position. Table 5 reports in-distribution probe accuracy on intact and position shuffled inputs, per layer, at initialization. Across these middle layers the intact probe reaches about 88 to 90 percent and shuffling reduces it by roughly

6 to 10 points, so a material part of the initial in-distribution accuracy is explained by fixed structure rather than by a learned signal.

Table 5. In-distribution probe accuracy on intact and position shuffled inputs, per layer, on Pythia-160M at initialization. Shuffling reduces accuracy at every layer.

Layer	Intact	Position shuffled
1	88.7%	82.3%
2	89.0%	82.7%
3	90.0%	79.7%
6	87.7%	78.7%

Projection removal. This experiment is reported in Appendix H. It shows that removing the direction a duplication probe uses does not reduce the repeated name bias.

D. Suppressor Trajectory and Split-Safe Confirmation

Table 6 traces the effect of mean-ablating the suppressor head across training, for Pythia-160M (L8H9) and Pythia-410M (L12H12). In each model the head is the one whose mean ablation most reduces mature IOI performance, and the same head is mean-ablated at every earlier checkpoint without reselection. A negative mean-ablation effect means that removing the head lowers the IO-minus-S logit difference, so the intact head supports the correct answer. The effect is near zero during the window and grows over the broader recovery period. The dominant head differs between the two models, which is expected because head positions are not shared across models.

Table 6. Change in logit difference from mean-ablating the mature suppressor head at each training step, with the accuracy at that step. The head is fixed at its maturity identity and mean-ablated at every earlier checkpoint.

	Pythia-160M (L8H9)		Pythia-410M (L12H12)	
Step	Mean-ablation effect	Accuracy	Mean-ablation effect	Accuracy
1000	−0.00	0.42	+0.00	0.45
2000	−0.01	0.32	−0.00	0.29
3000	−0.74	0.57	−0.17	0.82
5000	−2.39	0.88	−0.87	0.85
8000	−2.49	0.97	−1.62	0.98
13000	−1.77	0.97	−1.81	0.98
final	−2.66	1.00	−1.76	1.00

Suppressor characterization. Table 7 reports, for the mature-selected suppressor head, its attention to the S2 position and the projection of its output onto the IO-minus-S direction. In Pythia-160M both grow from near zero during the window to large values at maturity. For Stanford GPT-2 Small only the mature characterization is available, where zero-ablating the head reduces IOI accuracy by 0.09.

Table 7. Characterization of the mature suppressor head in Pythia-160M and Stanford GPT-2 Small. For each head the table reports its attention to the S2 position and the projection of its output onto the IO-minus-S direction. In Pythia-160M both quantities grow during recovery for the head L8H9, while for Stanford GPT-2 Small only the mature values of the head L10H10 are available.

Model and head	Step	Attention to S2	IO-minus-S output projection
Pythia-160M L8H9	1000	0.023	0.002
Pythia-160M L8H9	3000	0.869	0.986
Pythia-160M L8H9	final	0.922	5.669
Stanford GPT-2 Small L10H10	mature	0.589	6.240

Split-safe confirmation. Because the head above is chosen using its mature effect, we confirm the ordering with a stricter analysis on the independently trained Pythia-160M model. We freeze a mature-defined set of 15 suppressor heads by its functional signature on a selection split, using a separate validation split, before any developmental trajectory is evaluated. The held-out split excludes every name and template family used in discovery. Each of 192 held-out items supplies two matched same-S2 contrasts, with the S2 token identical between the target and donor runs. For each frozen head, its complete output at the final token position is copied from the matched donor run into the target run, giving 384 observations per checkpoint. We score the model by the fair three-way margin, defined as $\mathrm{logit(IO)} - \max\{\mathrm{logit(S)}, \mathrm{logit(alternate)}\}$, where the alternate is the donor's third name. The two contrasts per item are averaged before bootstrapping, so the interval resamples 192 independent item-level means rather than treating the 384 contrasts as independent. A positive replacement effect during the window means that replacing the mature-defined set improves the margin, so the native set is not yet acting suppressively; a negative effect later means that replacing it harms the margin, so the native set supports the correct answer. Table 8 reports this effect across training. At steps 1800 and 2000 the effect is significantly positive, 0.091 at step 2000 with an interval from 0.062 to 0.121. It is indistinguishable from zero at steps 3200 and 5000 and is significantly negative by step 6000, at -0.64 with an interval from -0.80 to -0.47. These are read as an observed ordering rather than exact formation times, since the checkpoint at which an interval first excludes zero depends on effect size and variance.

Table 8. Frozen suppressor set on the retrained model, evaluated on 192 held-out items. The table reports the fair three-way margin effect of replacing the set, with cluster-aware 95 percent intervals. A negative value means that replacing the native suppressor set harms the fair three-way margin, so the native set is supporting the correct answer. Replacing the set significantly improves the margin during the window, has no clearly detectable effect at the intermediate checkpoints, and significantly harms the margin by step 6000.

Step	Margin effect	Cluster-aware interval
1800	$+0.093$	$[+0.062, +0.125]$
2000	$+0.091$	$[+0.062, +0.121]$
3200	-0.051	$[-0.115, +0.014]$
5000	-0.079	$[-0.163, +0.006]$
6000	-0.636	$[-0.802, -0.468]$

E. Behavior and Intervention Reversal Across Scale

Table 9 reports, for six Pythia scales, the lowest IOI accuracy during training and the matched S2 intervention effect at the window and at maturity, all measured under the 300-prompt, ten-family protocol used in the main analyses. The floor accuracy and the intervention effect in each row are measured in the same experiment on the same prompts. Every scale falls below chance during early training, and the intervention effect is positive during the window and negative at maturity at every scale. For the 160M primary window the effect survives dependence-aware inference. Resampling by template gives 0.68 to 1.25 at the dip and -4.54 to -3.72 at maturity, and resampling by name-pair gives 0.78 to 1.15 and -4.45 to -3.82. The intervention effects at the other scales are point estimates under the prompt-level protocol, and we do not claim that a cluster-aware interval excludes zero for the small 410M dip effect. The three smallest scales use windows of layers 3 to 5, 6 to 10, and 4 to 8 respectively. The larger models use middle layer windows scaled to depth, listed in the last column.

Table 9. Behavior and intervention reversal across six Pythia scales, all under the 300-prompt, ten-family protocol. Floor accuracy and intervention effect in each row are from the same experiment. The intervention effect at the window is positive and at maturity is negative at every scale.

Model	Floor accuracy	Window step	Intervention at window	Intervention at maturity	Patch window
Pythia-160M	31.7%	2000	$+0.95$	-4.12	layers 3 to 5
Pythia-410M	29.3%	2000	$+0.11$	-3.63	layers 6 to 10
Pythia-1B	36.3%	1000	$+0.43$	-3.49	layers 4 to 8
Pythia-2.8B	29.7%	1000	$+0.33$	-4.05	layers 8 to 14
Pythia-6.9B	32.3%	2000	$+0.84$	-4.12	layers 8 to 14
Pythia-12B	42.0%	2000	$+0.43$	-3.96	layers 9 to 16

F. Robustness Across Seeds, Data Order, and Initialization

Table 10 reports, for each of the nine PolyPythias 160M variants, the behavioral floor and the matched-S2 intervention effect at the window and at maturity. The behavioral floor is the minimum IO-versus-S forced-choice accuracy in the fifteen-family behavioral sweep, and the intervention effects are measured separately with the ten-family matched-S2 protocol, so the two columns come from different runs and the intervention checkpoint is not necessarily the exact behavioral minimum. Three variants change the random seed, three change the order of the training data, and three change the initialization, each holding the other factors fixed. All nine variants show the positive-to-negative intervention reversal, positive at the window and negative at maturity, while their independently measured behavioral floors range from 18.0 to 36.7 percent. The intervention-checkpoint accuracy column reports the accuracy at the checkpoint where the intervention was measured, which is close to but not always equal to the behavioral floor. For every variant, the intervention is evaluated at step 2000 and at maturity. Step 2000 is a low-accuracy checkpoint but is not necessarily the exact minimum from the separate behavioral sweep. The intervention effects are point estimates.

Table 10. PolyPythias replication. Behavioral floor is the minimum IO-versus-S forced-choice accuracy in the fifteen-family behavioral sweep. Intervention effects are measured separately with the ten-family matched-S2 protocol. The intervention reverses sign in all nine.

Variant	Behavioral floor	Intervention-checkpoint accuracy	Intervention at window	Intervention at maturity
Seed 1	32.3%	30.7%	+0.55	−3.92
Seed 3	18.0%	14.7%	+0.99	−2.89
Seed 5	34.3%	25.3%	+1.11	−4.04
Data order 1	29.3%	30.7%	+1.20	−4.06
Data order 2	28.0%	19.3%	+1.35	−3.71
Data order 3	35.3%	36.3%	+1.01	−3.24
Initialization 1	33.3%	29.0%	+0.80	−4.56
Initialization 2	36.7%	35.7%	+0.84	−4.75
Initialization 3	31.0%	27.3%	+1.47	−3.51

G. Stanford GPT-2 Intervention Reversal

The reversal also appears in a second model family. Using the same 300-prompt, ten-family protocol with the residual-stream activations at S2 patched over layers 3 to 5, Table 11 reports the matched S2 intervention effect on Stanford GPT-2 Small across training, together with the IOI accuracy and 95 percent prompt-bootstrap interval at each checkpoint. The effect is positive at the early checkpoints, where accuracy is low, and negative at the later checkpoints, where accuracy is high, with intervals excluding zero, the same pattern as in Pythia.

Table 11. Matched S2 intervention effect on Stanford GPT-2 Small across training. The effect reverses from positive during the window to negative at maturity.

Step	1500	3000	10000	100000
IOI accuracy	12%	18%	55%	97%
Intervention effect	+1.03	+0.77	−0.23	−2.89
95% interval	[+0.95, +1.10]	[+0.69, +0.85]	[−0.33, −0.13]	[−3.03, −2.76]

H. Removing the Probe Direction

This experiment tests whether the direction a probe uses to read the repeated name is the direction the model's behavior depends on. On Pythia-160M at step 2000, we train a logistic-regression probe on the layer-5 residual-stream activations at S2. Writing $\hat{w}$ for the unit vector along the probe weight, we replace each activation a with $a - \lambda(a \cdot \hat{w})\hat{w}$ at strengths $\lambda \in \{0.5, 1, 2, 4\}$, so $\lambda = 1$ removes the probe component entirely, and measure the effect on the logit difference. The values are point estimates over the evaluation set. Table 12 also reports three control directions, a direction orthogonal to the probe direction, a direction from a probe trained on shuffled labels, and a random direction. Removing the probe direction does not reduce the preference for the repeated name at any strength. It slightly increases the preference, and the increase grows with strength, while the control directions have almost no effect. Removing the particular direction learned by this

probe therefore does not reduce the repeated-name preference. This does not rule out other linear directions, distributed representations, or nonlinear encodings of repetition.

Table 12. Effect of removing directions from the residual-stream activations at S2, measured at the window on Pythia-160M. Removing the probe direction does not reduce the preference for the repeated name at any strength. Lower accuracy means a stronger preference for the repeated name.

Direction removed	Logit difference	Accuracy
None (baseline)	-0.75	0.32
Probe direction, strength 0.5	-0.81	0.30
Probe direction, strength 1.0	-0.86	0.29
Probe direction, strength 2.0	-0.95	0.27
Probe direction, strength 4.0	-1.07	0.23
Orthogonal control, strength 1.0	-0.74	0.32
Shuffled label control, strength 1.0	-0.75	0.32
Random control, strength 1.0	-0.74	0.32

I. Protocol Summary

The analyses use released Pythia checkpoints (Biderman et al., 2023), the nine PolyPythias 160M variants (van der Wal et al., 2025), and Stanford GPT-2 Small (Karamcheti et al., 2021), together with one Pythia-160M model trained from scratch with a new seed. Protocols differ by experiment, and we list them per experiment rather than stating one protocol for all.

- **Matched S2 intervention.** First ten BABA-style template families, 30 prompts per family, 300 prompts, with symmetric exchanges of the indirect-object and repeated-subject names. The residual-stream activations at S2 are replaced over a middle layer window per scale, listed in Appendix E, with the activations from a control prompt whose second occurrence of the repeated name is a third single-token name.
- **Position battery.** The same 300 prompt set, replacing one position at a time.
- **Early single-head sweep.** At step 1000, we zero-ablate all 144 attention heads separately, using the first eight BABA-style template families and 20 prompts per family, giving 160 prompts.
- **Suppressor trajectory.** Using the 300-prompt, ten-family evaluation, we select the head whose mean ablation most reduces mature IOI performance, fix its identity, and mean-ablate that same head at every earlier checkpoint.
- **Held-out probe.** A probe trained on 600 examples from one set of names and templates and evaluated on a disjoint set, in two folds that swap the sets, predicting the repeated name position against a non-repeating control position.
- **Position-shuffle probe.** In-distribution probes on intact and position shuffled inputs.
- **Projection removal.** A probe trained on the residual-stream activations at S2, whose direction is removed at several strengths.
- **Split-safe suppressor set.** A suppressor set fixed by its functional signature using a selection split and a separate validation split, then evaluated on a disjoint held-out split.
- **Locked input control.** A locked benchmark of 800 prompts with a fixed prompt set and a stored prompt hash, comparing de-duplication, an alternate de-duplication, and a filler placebo, with cluster-aware intervals.
- **Language modeling loss.** A fixed sample from the Pile training stream on the monology Pile uncopyrighted train split (Gao et al., 2020), evaluated as described in Appendix A.

Unless otherwise stated, intervals are 95 percent bootstrap intervals over prompts. Results explicitly labeled cluster-aware resample the recorded item, template, or name-pair clusters. The held-out probe reports point estimates over the evaluation examples. Code, analysis scripts, configuration files, and released results are available in the public repository.