Dissociating Decodability and Causal Use in Bracket-Sequence Transformers

Aryan Sharma [1] **Cutter Dawes** [2] **Shivam Raval** [3]

Abstract

When trained on tasks requiring an understanding of hierarchical structure, transformers have been found to represent this hierarchy in distinct ways: in the geometry of the residual stream, and in stack-like attention patterns maintaining a last-in, first-out ordering. However, it remains unclear whether these representations are causally used or merely decodable. We examine this gap in transformers trained on the Dyck language (a formal language of balanced bracket sequences), where the hierarchical ground truth is explicit. By probing and intervening on the residual stream and attention patterns, we find that depth, distance, and top-of-stack signals are all decodable, yet their causal roles diverge. Specifically, masking attention to the true top-of-stack position causes a sharp drop in long-distance accuracy, while ablating the low-dimensional residual stream subspaces that encode this hierarchical information has comparatively little effect. This effect extends even to natural language settings, suggesting that even in a controlled setting where the relevant hierarchical variables are known, decodability alone does not imply causal use.

1. Introduction

Linear probes are widely used to test whether model states encode algorithmic and linguistic variables (Hewitt & Manning, 2019; Hewitt & Liang, 2019; Elazar et al., 2021). However, encoding does not directly imply use, and decodable variables have previously been seen to be behaviorally irrelevant (Hewitt & Liang, 2019; Elazar et al., 2021). This gap has become increasingly important as many empirical claims in interpretability methodology rely on decodability alone. More broadly, false positives from multiple testing and underconstrained analysis pipelines are a familiar methodological risk, meaning this concern is not unique to interpretability (Méloux et al., 2025).

We study this gap in the Dyck language setting—the formal language of balanced bracket sequences—where the relevant hierarchical variables are explicitly computable. Intuitively, a Dyck string is a sequence of nested brackets that must be closed in a last-in-first-out manner, so predicting each closing bracket requires tracking which opener is currently unmatched. This minimal, fully specified instance of hierarchical structure provides us with a controlled setting in which representational and causal claims can be separated cleanly, since the relevant variables can be probed directly and interventions can be evaluated against ground-truth structure. Specifically, we ask: when hierarchical variables are decodable within Dyck transformers, which are actually relevant to model computation?

Our main claim is that the behavior of these Dyck transformers depends more strongly on sparse top-of-stack attention routing (i.e., each closing bracket attending almost entirely to the current stack top) than on decoded residual-stream directions. We support this claim with five findings:

1. Hierarchical signals (depth, distance, and top-of-stack identity) are strongly decodable early in training.
2. Depth and distance remain decodable OOD, yet ablating their probe-aligned subspaces has near-zero behavioral effect.
3. Masking the true top-of-stack attention edge is catastrophic.
4. Activation patching localizes the critical retrieval step to a single computational block.
5. A templated natural-language setting shows the same dissociation outside of the Dyck language case.

2. Related Work

Prior work has shown that syntactic and hierarchical structure is linearly decodable from language model representations. In particular, Hewitt & Manning (2019) demonstrated that parse-tree distance and depth are linearly decodable from transformer hidden states, and we test whether they are

[1]Yale University [2]Independent [3]Harvard University. Correspondence to: Aryan Sharma <aryan.sharma@yale.edu>.

Accepted to the 2nd Workshop on Compositional Learning at ICML 2026, Seoul, South Korea. *Copyright 2026 by the author(s).*

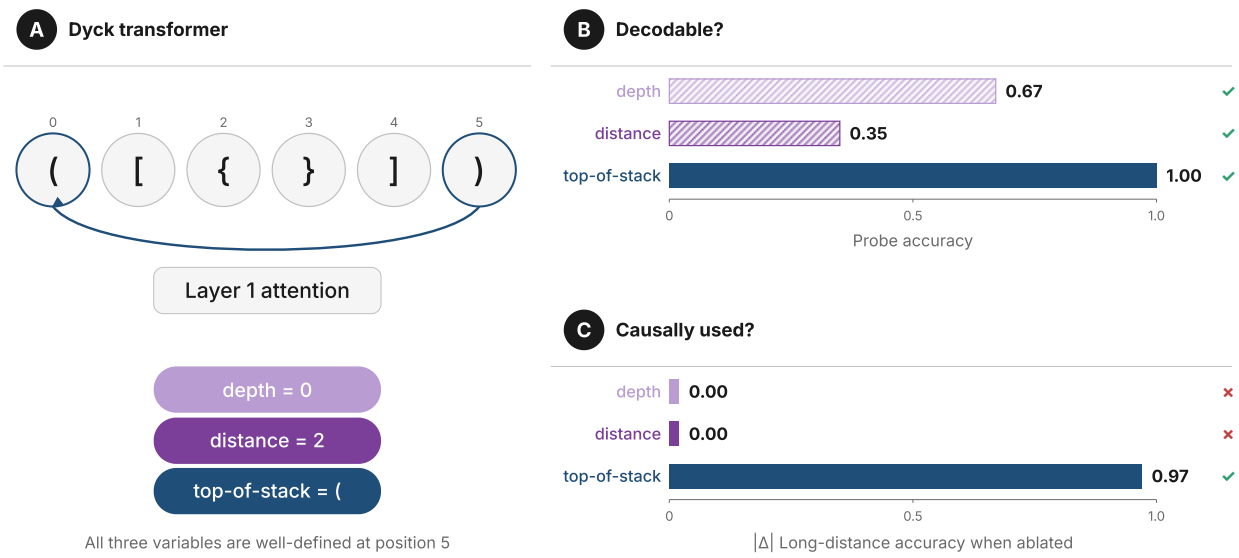

Figure 1. **Decodability $\neq$ causal use. (A)** At a closing bracket in a trained Dyck transformer, layer-1 attention routes sharply to the matching opener, while three latent variables (depth, distance, and top-of-stack) are well-defined at this position. **(B)** All three variables are decodable from the model. **(C)** Only top-of-stack is causally used: ablating the probe-aligned residual-stream subspaces for depth and distance has near-zero effect on task performance, while masking the top-of-stack attention edge collapses it.

causally used. Subspace-removal methods extend probing by ablating decoded directions to test causal use (Ravfogel et al., 2020; 2022), and have found that certain decoded features can be removed without affecting downstream behavior, which motivates our decodability-versus-causality framing in a controlled setting.

Related work in causal abstraction (Geiger et al., 2021) and mechanistic interpretability (Elhage et al., 2021; Conmy et al., 2023; Wang et al., 2023) emphasizes specific attention pathways rather than globally decodable geometry. Notably, Wang et al. (2023) identified a small circuit of attention edges carrying the indirect-object-identification task in GPT-2. Our attention-knockout finding is a formal-language instance of this pattern, with the advantage that the ground-truth mechanism is explicitly computable.

The Dyck language is a standard testbed for hierarchical structure in transformers (Yao et al., 2021; Murty et al., 2023; Tiwari et al., 2025; Hayakawa & Sato, 2025; Wen et al., 2023). We add a causal, per-model analysis differentiating linearly decodable features from those used at inference time.

3. Experimental setup

Task and models. The Dyck-(k, m) language consists of balanced strings with k bracket types and maximum depth m, where depth counts the number of open brackets. For example, in `([{}])`, the depth after each token is $(1, 2, 3, 2, 1, 0)$. We take distance to be the parse-tree distance between token pairs, following the methodology of the structural probe in Hewitt & Manning (2019); in the example above, the closing `)` and its matching opener `(` are at parse-tree distance 2 (Figure 1). At each position, we refer to the *stack top* as the most recently opened unclosed bracket. We use the Dyck language because it exposes explicit latent hierarchical variables—including stack top, depth, and distance—in a setting where both probing and intervention can be evaluated against the ground truth.

As a baseline setting, we train on Dyck-$(20, 10)$ with 2-layer, 1-head transformers, embedding dimension $d \in \{16, 32, 64\}$, 5×10^5 optimization steps, a training set of 10^4 sequences, and a held-out test set of 10^3 sequences with no overlap. Data generation details are in Appendix A.2. On held-out IID evaluation data, these base models achieve 98.2%–100% long-distance accuracy (i.e., accuracy on matched bracket pairs whose closing position is at least 10 positions from the corresponding opening position) across the three widths. Architecture and training details are in Appendix A.1.

We additionally scale task difficulty by varying $(k, m) \in \{(20, 18), (30, 14), (40, 10), (40, 18)\}$ at $d = 64$. Training

details and full results are in Appendix A.10.[1]

4. Probing decodability of task-relevant quantities

Residual-stream and attention probes. We probe depth and distance from the layer-2 residual stream and report the Pearson correlation between predicted and true values as our primary metric. Probe architectures, training details, and R^2/MSE numbers are in Appendices A.3 and A.6. We additionally sweep over probe capacity to show that regularization does not change the qualitative OOD pattern (Appendix A.7).

When probing top-of-stack position, we read it directly from the attention pattern, since the model's prediction at a closing bracket depends on routing attention to a single source token. Specifically, at a closing-bracket query q with ground-truth stack-top source position $s(q)$, we report the *attention miss* as $1 - a^{(1)}_{q,\,s(q)}$, where $a^{(1)}_{q,\cdot}$ is the layer-1 attention row at q. This has no fitted probe parameters, so overfitting is not a concern. A logistic-regression probe on the residual stream is reported in Appendix A.8.

We note that the top-of-stack identity and distance are related but not redundant, since top-of-stack fixes which bracket type is the next closer (categorical), while distance measures how far back the corresponding opener sits (scalar). Knowing distance at a closing bracket determines identity via the sequence, but the converse does not hold. Meanwhile, depth is orthogonal to both.

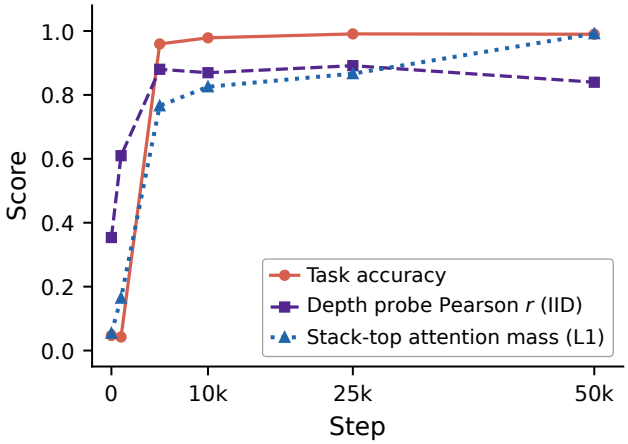

Figure 2. Training dynamics for the $d = 64$ base model on the first 50k steps. Task accuracy (red), depth probe Pearson IID (dark purple), and layer-1 stack-top attention mass (blue) all stabilize by step 5k. The full 500k-step trajectory is in Figure 7.

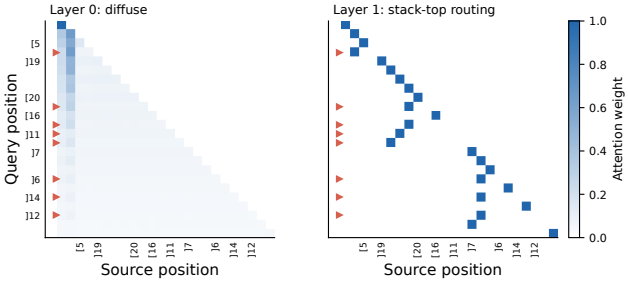

Figure 3. Attention weights for a representative length-24 sequence. Orange triangles mark closing-bracket query rows. Layer 0 (left) attention is diffuse; Layer 1 (right) collapses each marked row onto the matching opener's column.

Hierarchical signals emerge early. Figure 2 shows all three signals over training. By step 5k, top-of-stack attention mass and depth probe Pearson correlation are already strong, and task accuracy is near ceiling. This indicates that depth, distance, and top-of-stack identity are recoverable before complete task convergence. Probe accuracy as a function of sequence position is in Appendix A.16.

Figure 3 shows the corresponding picture through attention. Each row of the heatmap is an attention distribution for a single query position, and the orange triangles mark the rows corresponding to closing brackets, i.e., the positions whose next-token prediction depends on retrieving the matching opener. In layer 0, these marked rows are diffuse, meaning attention is spread broadly across source positions. In layer 1, each marked row collapses onto a single bright cell, which is consistently the column of the matching opening bracket. We show this mechanism is causally necessary in Section 5.

[1]All code is available at `https://github.com/aryans-15/bracket-transformers`.

Out-of-distribution decodability. Figure 4 reports decodability across IID, OOD-length, and OOD-depth splits (defined in Appendix A.2). The top-of-stack attention is near-perfect on all three splits, while the depth and distance probe Pearson correlations degrade but stay positive. The positive Pearson alongside negative OOD R^2 (Appendix A.6) indicates the probes preserve the rank ordering of depth and distance OOD but are poor absolute predictors, whereas top-of-stack stays robustly recoverable. We additionally sweep probe capacity to reproduce the same qualitative pattern, verifying that OOD degradation is not a probe-overfitting artifact (Appendix A.7).

5. Interventions identify causally important quantities

Causal interventions. For depth and distance, we ablate probe-identified subspaces in the residual stream at ranks $r \in \{2, 4, 8, 16, 32, 64\}$, comparing against random rank-matched baselines (further details in Appendix A.4, and intervention strength sensitivity analysis in Appendix A.14).

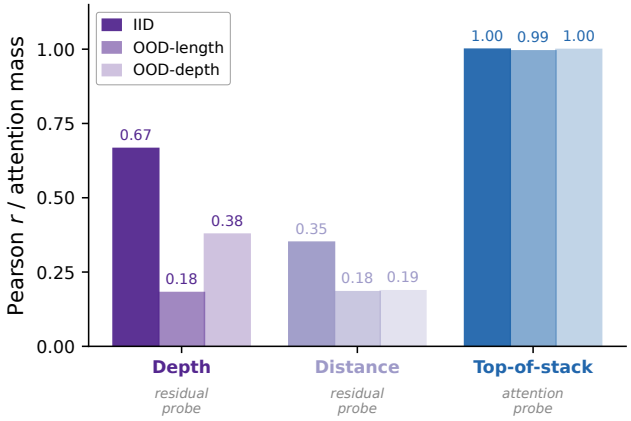

Figure 4. Decodability across splits. Depth and distance (purple) report residual-probe Pearson r; top-of-stack (blue) reports layer-1 attention mass. Top-of-stack stays near-perfect across IID, OOD-length, and OOD-depth; depth and distance degrade but stay positive. Long-distance task accuracy is ≥ 0.994 across all splits.

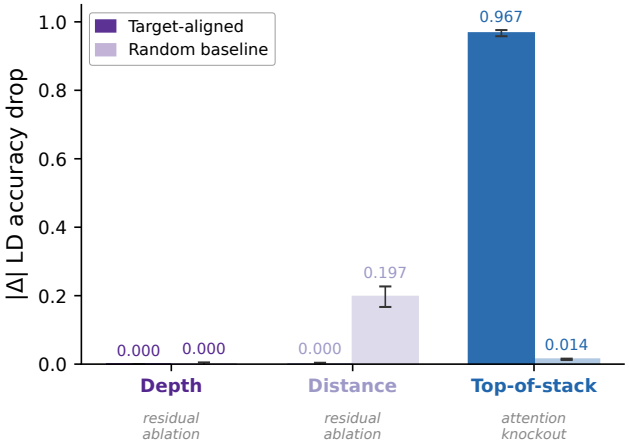

Figure 5. Causal intervention $|\Delta|$ long-distance accuracy: residual ablation of depth (left), residual ablation of rank-2 distance subspace (middle), and layer-1 attention-edge knockout of top-of-stack (right).

For top-of-stack, we perform attention knockout by zeroing the attention score to the true source position before softmax (Wang et al., 2023), against random control edges. Query/key-space interventions are in Appendix A.15.

Activation patching. To localize *where* in computation the stack-top signal becomes sufficient, we use activation patching (Meng et al., 2022): we substitute clean activations at each sublayer checkpoint of a corrupted run and measure recovery. Setup details, including corruption strategy, cache points, and offset sweep, are in Appendix A.5.

Depth and distance are not causally necessary. Ablating the decoded depth and distance directions from the residual stream has little effect. Depth ablation produces zero drop on the harder variants despite a final probe Pearson r of $+0.67$ on IID, indicating the model encodes depth but does not use it for prediction. Similarly, ablating probe-aligned distance subspaces at low ranks has a near-zero effect, while removing a random subspace of the same rank causes substantial drops (Appendix A.13). These residual directions are not behaviorally necessary, but query/key-space interventions (Appendix A.15) show they participate in computing layer-1 routing, meaning the decodable geometry is likely consumed by the attention mechanism.

Top-of-stack routing is causally necessary. In our base $d = 64$ models, layer 1 attention to the true stack-top position is $\approx$ 0.98–0.997, compared to a uniform baseline of ≈ 0.013 (Appendix A.10). This routing mechanism is stable across seeds (Appendix A.18).

To test causal necessity, we perform attention knockout. We find that masking the true stack-top edge drops long-distance accuracy by -0.967 ± 0.009 on the harder-variant test sets, while masking a randomly chosen edge only drops it by -0.014 ± 0.002. Figure 5 shows all three interventions on a common axis: residual-stream depth ablation, residual-stream distance ablation (at low rank), and attention-edge knockout of top-of-stack. Only the attention-edge knockout collapses accuracy.

Activation patching localizes the mechanism to layer 1 attention. Our attention masking result shows the stack-top edge is necessary, but does not answer at what stage that signal becomes sufficient. To bridge this gap, we turn to activation patching. Figure 6 shows that patching before layer 1 attention yields essentially no recovery, while patching at or after layer 1 attention at the closing bracket position produces 100% recovery. This indicates that the critical retrieval step is complete at layer-1 attention, and once that operation is correct, the residual state is sufficient for task success.

The dissociation persists in natural language. To test whether this dissociation extends beyond formal languages, we train a transformer with the same architecture ($d = 64$, 2 layers, 1 head) on a templated subject–verb agreement task. In the task, each input has one subject plus 0–3 prepositional-phrase attractors with a randomly chosen number; the model must match a final `<verb>` token to the given subject. We choose this setup because our template removes the redundant agreement cues of unconstrained English, forcing retrieval of the subject's number through attention rather than a local heuristic. Over four seeds, we find that our model achieves 99.2–99.4% accuracy uniformly across attractor counts, ruling out a positional shortcut. We provide the full setup and per-seed numbers in Appendix A.19.

We further find that all three findings from our Dyck experi-

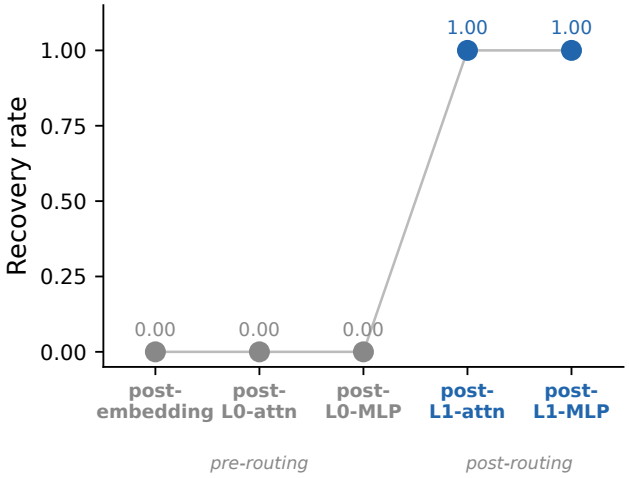

Figure 6. Activation patching recovery rate at the closing-bracket query position. Recovery jumps from 0 to 1 between post-L0-MLP and post-L1-attention. Full offset heatmap in Appendix A.12.

ments recur. First, the subject-number probe stays at ≥ 0.99 accuracy IID and OOD, while the attractor-count probe collapses OOD ($R^2 = -12.3$), just as depth and distance did. Second, masking the single subject-to-verb attention edge at layer 1 drops accuracy to chance on three of the four seeds. This aligns with the solution multiplicity in our Dyck setting (Appendix A.18). Third, activation patching localizes the computation to layer-1 attention (recovery 1.000 after layer-1 attention vs. 0.000 after the embedding), identical to the mechanism in our Dyck setting.

6. Discussion

We study the relationship between decodability and causal importance in transformer representations, within a controlled setting containing explicit hierarchy. We find depth and distance to be decodable early and positively correlated OOD, yet ablating their probe-aligned subspaces has near-zero effect. Top-of-stack identity behaves differently: its attention probe is lossless under distribution shift, and masking the single attention edge collapses task performance. Activation patching further establishes that the top-of-stack attention mechanism is not only causally necessary, but its recovery is sufficient for task success. These results make clear the importance of dissociating between decodability and causality in transformers and other neural networks. We also stress that our analysis goes beyond correlational diagnostics, since output-facing tools such as the logit lens (nostalgebraist, 2020) can show a variable is readable at a layer, but only our ablations and attention knockouts establish that it is causally necessary for the prediction.

Our study has some limitations. First, all models here are 2-layer, 1-head transformers, and whether the same dissociation holds in scaled models is untested. Second, both the Dyck strings and our natural-language pilot (Section 5) are retrieval-based tasks whose success is determined by a single matching token, which could potentially favor sparse attention routing over distributed computation. For this reason, tasks requiring the compositional integration of several latent variables could show a weaker dissociation. We leave both of these to future work.

Acknowledgments

We thank the Supervised Program for Alignment Research (SPAR) for providing compute and resources to carry out this work.

References

Conmy, A., Mavor-Parker, A., Lynch, A., Heimersheim, S., and Garriga-Alonso, A. Towards automated circuit discovery for mechanistic interpretability. In *Advances in Neural Information Processing Systems*, 2023. doi: 10.48550/arXiv.2304.14997. URL https://arxiv.org/abs/2304.14997. NeurIPS 2023 Spotlight.

Elazar, Y., Ravfogel, S., Jacovi, A., and Goldberg, Y. Amnesic probing: Behavioral explanation with amnesic counterfactuals. *Transactions of the Association for Computational Linguistics*, 9:160–175, 2021. doi: 10.1162/tacl_a_00359. URL https://arxiv.org/abs/2006.00995.

Elhage, N., Nanda, N., Olsson, C., Henighan, T., Joseph, N., Mann, B., Askell, A., Bai, Y., Dhariwal, P., Ganguli, D., et al. A mathematical framework for transformer circuits. *Transformer Circuits Thread*, 2021. URL https://transformer-circuits.pub/2021/framework/index.html.

Geiger, A., Potts, C., and Icard, T. Causal abstractions of neural networks. In *Advances in Neural Information Processing Systems*, volume 34, pp. 9574–9586, 2021. URL https://papers.neurips.cc/paper_files/paper/2021/file/4f5c422f4d49a5a807eda27434231040-Paper.pdf.

Hayakawa, D. and Sato, I. Theoretical analysis of hierarchical language recognition and generation by transformers without positional encoding. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 30777–30834, 2025. doi: 10.18653/v1/2025.acl-long.1488. URL https://aclanthology.org/2025.acl-long.1488/.

Hewitt, J. and Liang, P. Designing and interpreting probes with control tasks. In *Proceedings of EMNLP-IJCNLP*,

pp. 2733–2743, 2019. doi: 10.18653/v1/D19-1275. URL https://aclanthology.org/D19-1275/.

Hewitt, J. and Manning, C. D. A structural probe for finding syntax in word representations. In *Proceedings of NAACL-HLT*, pp. 4129–4138, 2019. doi: 10.18653/v1/N19-1419. URL https://aclanthology.org/N19-1419/.

Méloux, M., Dirupo, G., Portet, F., and Peyrard, M. The dead salmons of AI interpretability. *arXiv preprint arXiv:2512.18792*, 2025. URL https://arxiv.org/abs/2512.18792.

Meng, K., Bau, D., Andonian, A., and Belinkov, Y. Locating and editing factual associations in GPT. In *Advances in Neural Information Processing Systems*, volume 35, pp. 17359–17372, 2022. URL https://arxiv.org/abs/2202.05262.

Murty, S., Sharma, P., Andreas, J., and Manning, C. D. Grokking of hierarchical structure in vanilla transformers. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 439–448, 2023. doi: 10.18653/v1/2023.acl-short.38. URL https://aclanthology.org/2023.acl-short.38/.

nostalgebraist. Interpreting GPT: The logit lens. LessWrong, 2020. https://www.lesswrong.com/posts/AcKRB8wDpdaN6v6ru/interpreting-gpt-the-logit-lens.

Ravfogel, S., Elazar, Y., Gonen, H., Twiton, M., and Goldberg, Y. Null it out: Guarding protected attributes by iterative nullspace projection. In *Proceedings of ACL*, pp. 7237–7256, 2020. doi: 10.18653/v1/2020.acl-main.647. URL https://aclanthology.org/2020.acl-main.647/.

Ravfogel, S., Twiton, M., Goldberg, Y., and Bachrach, Y. Linear adversarial concept erasure. In *Proceedings of the 39th International Conference on Machine Learning*, pp. 18400–18421, 2022. URL https://proceedings.mlr.press/v162/ravfogel22a.html.

Tiwari, U., Gupta, A., and Hahn, M. Emergent stack representations in modeling counter languages using transformers. *arXiv preprint arXiv:2502.01432*, 2025. doi: 10.48550/arXiv.2502.01432. URL https://arxiv.org/abs/2502.01432.

Wang, K. R., 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*, 2023. URL https://openreview.net/forum?id=NpsVSN6o4ul.

Wen, K., Li, Y., Liu, B., and Risteski, A. Transformers are uninterpretable with myopic methods: a case study with bounded Dyck grammars. In *Advances in Neural Information Processing Systems*, 2023.

Yao, S., Peng, B., Papadimitriou, C., and Narasimhan, K. Self-attention networks can process bounded hierarchical languages. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (ACL-IJCNLP)*, pp. 3770–3785, 2021. doi: 10.18653/v1/2021.acl-long.292. URL https://aclanthology.org/2021.acl-long.292/.

A. Appendix

A.1. Training and architecture details

Table 1. Training hyperparameters (all models).

Parameter	Value
Optimizer	AdamW
Learning rate	3×10^{-4} (constant)
Adam β_1, β_2	$(0.9,\ 0.999)$
Weight decay	0.0
Batch size	128 sequences
Gradient clipping	1.0 (max norm)
Base model steps	500k
Harder variant steps	300k

Table 2. Model architecture (all variants).

Property	Value
Layers	2
Heads	1
MLP hidden dim	$4d$ (ReLU)
Positional encoding	Learned
Layer norm	Post-LN
Attention mask	Causal

Table 3. Parameter counts and vocabulary sizes (vocab size $= 2k + 2$). Base models use $d \in \{16, 32, 64\}$ with $k = 20$; harder variants fix $d = 64$ and vary k.

d	k	Parameters	Vocab size
16	20	15,424	42
32	20	43,136	42
64	20	135,424	42
64	30	136,704	62
64	40	137,984	82

Full training-dynamics trajectory. Figure 7 extends the main-text Figure 2 to the full 500k-step trajectory on a log x-axis. All three signals stabilize well before the end of training. The dashed vertical line marks step 50k.

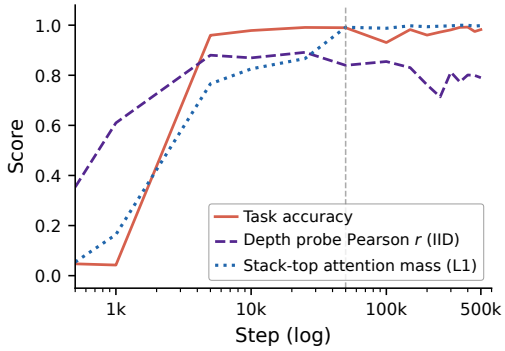

Figure 7. Full 500k-step training dynamics on a log x-axis. Task accuracy (red), depth probe Pearson IID (dark purple), and layer-1 stack-top attention mass (blue) all stabilize well before the end of training. Dashed vertical line marks step 50k.

A.2. Data generation and evaluation splits

We sample Dyck strings with a stack-based probabilistic procedure. At each step, if the stack is empty, the model samples uniformly from k opening brackets or END; if at maximum depth, it returns the closing bracket no matter what; and otherwise, it samples uniformly from k opening brackets or the one correct closing bracket.

Our base model is trained on Dyck-$(20, 10)$ with 10,000 training sequences (with length range of 22–512 and mean of 146.5) and 1,000 test sequences (with length range of 4–488 and mean of 51.9). The harder variants use 200,000 training and 20,000 test sequences each.

Out-of-distribution splits are constructed from our base set as follows. *OOD-length*: we train on sequences with length ≤ 64 (759 sequences) and evaluate on sequences with length ≥ 128 (124 sequences). *OOD-depth*: we train on sequences with max depth ≤ 4 (567 sequences) and evaluate on sequences with max depth ≥ 8 (288 sequences).

A.3. Probe training details

All our probes are trained on an 80/20 split across a base 1,000-sequence set, with up to 128 tokens subsampled per sequence, and hidden states extracted from the final transformer block without standardization. For activation patching experiments, we train probes on clean sequences only and apply them to corrupted forward passes.

Our depth probe is a ridge regressor with $\alpha = 1.0$, our top-of-stack probe is a multinomial logistic regression (`lbfgs`, $C = 1.0$), and our distance probe is a linear projection $B \in \mathbb{R}^{d \times 32}$ trained with AdamW (learning rate 3×10^{-3}, weight decay 10^{-4}, cosine schedule) for 1500 steps, fitting $\|Bh_i - Bh_j\|_2$ to the true pairwise Dyck tree distance. We additionally ablate probe-aligned distance subspaces at ranks $r \in \{2, 4, 8, 16, 32, 64\}$ for the causal-intervention experiments (Appendix A.13).

A.4. Causal intervention details

Residual-stream ablation (depth). We remove probe-identified directions from the residual stream: if $U_r \in \mathbb{R}^{d \times r}$ is an orthonormal basis for a probe-aligned rank-r subspace with projector $P_r = U_r U_r^\top$, we intervene on a hidden state h via $h' = h - \alpha P_r h$, with $\alpha = 1$ in the main text (other strengths in Appendix A.14). As a baseline, we apply the same procedure to random rank-matched orthonormal subspaces in the same layer. For completeness, we apply the same residual-stream ablation procedure to the top-of-stack direction; results are in Appendix A.9 and are consistent with our attention knockout findings.

A.5. Activation patching setup

We construct (clean, corrupted) sequence pairs by replacing a matched opening bracket with a wrong bracket type and only keeping pairs where the corrupted model incorrectly predicts the closing position. For each pair, we cache the residual stream at five sublayer boundaries in the clean run (post-embedding, post-attention, and post-MLP at each layer) and substitute the clean cache into the corrupted run one position and one cache point at a time, sweeping a ± 3 token window around the closing bracket to account for variation in the opening bracket's location. Corruption strategy and probe tracking details are in Appendices A.11 and A.17.

Distance subspace ablation. For distance, we ablate the probe-aligned subspace at ranks $r \in \{2, 4, 8, 16, 32, 64\}$ and compare the behavioral effect against ablating a random rank-matched subspace. This tests whether the distance geometry encoded in the residual stream is causally used.

Attention knockout. For each query token t, we identify the source position corresponding to the true top-of-stack token and set the corresponding boolean attention-mask entry to `True` (Wang et al., 2023). This sends the relevant logit to $-\infty$ before softmax, thereby removing that edge from the attention graph. We compare masking the true top-of-stack edge against masking random control edges.

A.6. Probe metrics: Pearson vs. R^2

The main text reports Pearson correlation between predicted and true depth/distance as the probe quality metric. For completeness, we also report R^2 and MSE in Table 4. We report Pearson because of scaling differences between IID and OOD splits; specifically, the mean depth shifts from ≈ 1 IID to ≈ 5 OOD-depth. These differences inflate MSE and R^2, making them inaccurate reflections of whether the probe recovers the underlying variable, while Pearson is scale-invariant.

Table 4. Depth and distance probe metrics across splits ($d = 64$ base model). Pearson is the main-text metric; R^2 and MSE are reported here for completeness.

Probe	Split	Pearson r	R^2	MSE
Depth	IID	$+0.67$	$+0.44$	0.60
Depth	OOD-length	$+0.18$	-2.20	22.78
Depth	OOD-depth	$+0.38$	-1.00	16.14
Distance	IID	$+0.35$	-0.17	1.53
Distance	OOD-length	$+0.18$	-1.98	41.47
Distance	OOD-depth	$+0.19$	-1.18	18.76

A.7. Probe capacity sweep

As a sensitivity check on probe capacity, we sweep the top-k PC-regularized variants ($k \in \{8, 16, 32\}$) and full-rank Ridge $\alpha \in \{1, 10, 100, 1000\}$ for depth. We find that no configuration recovers positive R^2 on either OOD split, and the top-10 PC h-probe gives IID $R^2 = 0.08$, OOD-length -3.25, OOD-depth -1.88 — worse than the full-rank $\alpha = 1$ probe on IID and OOD alike, indicating the first 10 PCs of the layer-2 residual stream do not cleanly carry the depth signal. This qualitative collapse is robust across the sweep. The Pearson r sweep shows the same qualitative pattern, as every probe-capacity configuration produces an OOD Pearson r below IID, confirming that the main-text degradation is not an artifact of probe capacity.

A.8. Residual stream top-of-stack probe

As a completeness check, we fit a multinomial logistic-regression probe for top-of-stack identity directly on the residual stream. This probe achieves 1.000 accuracy across IID, OOD-length, and OOD-depth splits, consistent with the attention-based readout. The main text uses the attention pattern as the canonical stack-top readout because it has no fitted parameters and is therefore resistant to overfitting.

A.9. Residual stream top-of-stack ablation

As a completeness check, we ablate the residual-stream stack-top direction identified by the logistic-regression probe above. The ablation has a much smaller effect than the attention-edge knockout ($|\Delta|$ LD $\ll 0.967$), consistent with the finding that stack-top is causally used in the attention pattern rather than the residual stream.

A.10. Harder variant results

Table 5. Harder variant results on IID test sets. LD = long-distance accuracy, ToS = top-of-stack probe accuracy, Attn = mean layer 1 stack-top attention mass, Unif = uniform attention baseline, Edge Δ = change in LD acc. when stack-top attention edge is masked.

Variant	LD acc.	Depth R^2	ToS acc.	Attn	Unif	Edge Δ
k20_m18	0.9999	0.887	0.9999	0.986	0.013	-0.954
k30_m14	0.9981	0.846	0.9999	0.983	0.014	-0.965
k40_m10	0.9993	0.711	0.9999	0.990	0.017	-0.977
k40_m18	0.9986	0.716	0.9999	0.983	0.013	-0.972

Table 6. Harder variant training summary. All runs use $d = 64$, seed 42, AdamW lr=3×10^{-4}. Steps to 99% denotes the first checkpoint at which long-distance accuracy exceeds 0.99.

Variant	Final loss	Steps to 99%	Wall-clock (h)
k20_m18	2.145	50k	5.9
k30_m14	2.331	50k	6.3
k40_m10	2.463	50k	7.1
k40_m18	2.482	150k	7.8

We report R^2 rather than Pearson r in Table 5 because

the harder variants are only evaluated IID, where the scale concerns motivating Pearson in the main text do not apply.

All four harder variants achieve a long-distance accuracy above 99.8% and top-of-stack probe accuracy above 99.99%. The depth R^2 decreases with k, indicating that depth becomes harder to linearly extract as bracket type complexity increases, yet the task performance still remains near ceiling, further emphasizing decodability without causality. Furthermore, the layer 1 attention directs over 98% of its weight to the true stack-top position in all variants, compared to a uniform baseline of 1.3%–1.7% (i.e., what equal attention across all positions would give). Finally, causal ablations of the probe-aligned distance subspace yield $\Delta = 0.000$ for all α across all variants, consistent with our base model results.

A.11. Corruption strategy for activation patching

We construct activation patching pairs by replacing a matched opening bracket with a different bracket type and evaluating at the corresponding closing bracket prediction position, only keeping pairs where the corrupted model predicts incorrectly. This strategy directly targets the bracket-type retrieval mechanism that layer 1 attention implements.

A.12. Activation patching: full offset heatmap

The main-text patching figure reports recovery rate only at the closing-bracket query position. Figure 8 shows a larger offset sweep, illustrating that recovery is concentrated at offset 0 for post-layer-1 cache points, with negligible recovery elsewhere. This confirms that the critical retrieval step is localized to a single sublayer and position.

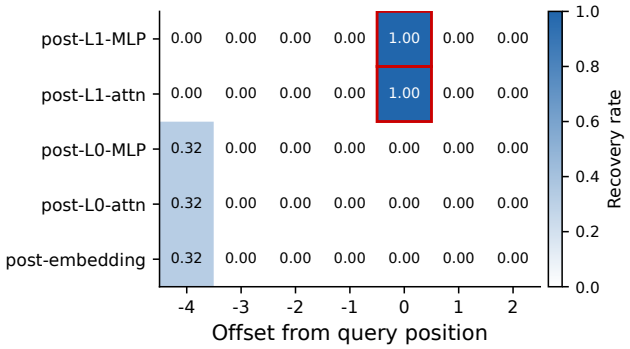

Figure 8. Full activation patching recovery heatmap across cache points (rows) and offsets from the closing bracket (columns).

A.13. Distance-sweep numeric summary

At low ranks ($r \in \{2, 4, 8\}$), the probe-aligned subspace is behaviorally inert while random rank-matched ablation causes substantial drops, showing the decoded subspace is not causally privileged. As rank grows, both ablations converge to near-total accuracy loss; at intermediate ranks ($r \leq 32$) the probe-aligned subspace remains substantially less damaging than random. This rules out the possibility that distance is used but distributed across too many dimensions for the low-rank probe to capture, further reinforcing that the decodable distance geometry does not drive the model's predictions. [2]

Table 7. Distance-subspace ablations across ranks ($d = 64$, IID). Probe Δ = accuracy change when removing the probe-aligned subspace; Rand Δ = same for a random rank-matched subspace; Gap = probe minus random. Rank 64 is aggregated across five model configurations ($d = 64$ base + four harder variants) with $\alpha = 1.0$ and five random subspace seeds per model; ranks 2–32 are from the original rank-sweep on the base model.

Rank	Probe Δ	Rand Δ	Gap
2	-0.000 ± 0.000	-0.197 ± 0.169	$+0.197$
4	-0.006 ± 0.017	-0.301 ± 0.185	$+0.295$
8	-0.015 ± 0.028	-0.485 ± 0.258	$+0.470$
16	-0.316 ± 0.447	-0.625 ± 0.262	$+0.309$
32	-0.458 ± 0.458	-0.751 ± 0.220	$+0.293$
64	-0.931 ± 0.024	-0.959 ± 0.018	$+0.028$

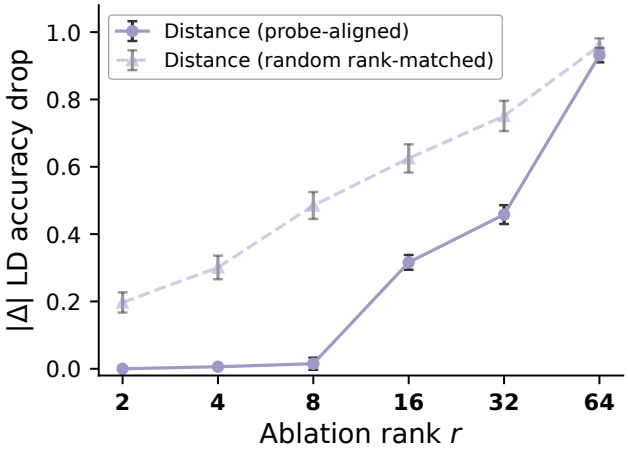

Figure 9. Accuracy drop from ablating the probe-aligned distance subspace (solid) vs. a random subspace of equal rank (dashed), across ranks $r \in \{2, 4, 8, 16, 32, 64\}$.

A.14. α-sweep (intervention strength)

All our residual-stream ablation results in the main text use $\alpha = 1$. For completeness, we also present results of ablating with $\alpha \in \{0.5, 1.5\}$ to check robustness to intervention strength. This sweep further verifies our claim that depth and distance ablation have negligible behavioral effect, and random rank-matched ablation has a much larger effect at low ranks. Figure 10 shows the full α-sweep.

[2] At rank $r = d = 64$, the ablation is degenerate, since any orthonormal basis spans the full residual space. In this case, $P_r = I$, and the probe-aligned and random ablations both zero the residual stream identically. We report this row only for completeness, and treat its values as noise.

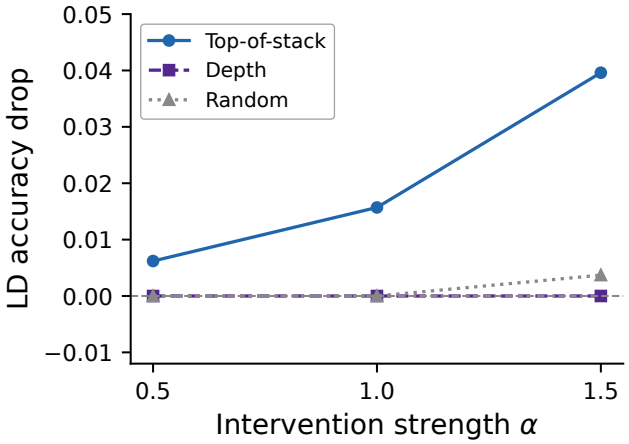

Figure 10. Residual-stream ablation at three intervention strengths.

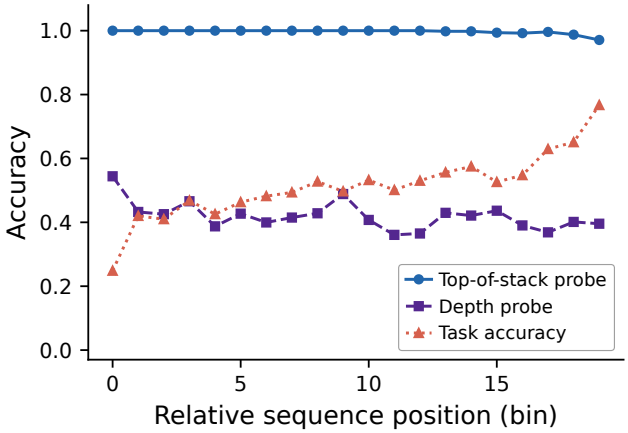

Figure 11. Probe and task accuracy vs. relative sequence position.

A.15. Query/key-space interventions

We additionally test whether probe-aligned directions are causally involved in the computation of attention scores. Let $q = W_Q h$ and $k = W_K h$ be the query and key vectors, and let U_r be the probe-aligned residual basis. We project this U_r into the query/key space via $B_Q = \mathrm{orth}(W_Q U_r)$ and $B_K = \mathrm{orth}(W_K U_r)$, then remove these components from q and k before recomputing the attention weights.

When doing this, we find that removing the probe-aligned components from query/key space causes large drops in stack-top attention at subspace dimensions 8, 16, and 32 (with mean changes of -0.778 ± 0.082, -0.816 ± 0.041, and -0.890 ± 0.059), which are much larger than the weak effects seen under residual-stream ablation. Query-only and key-only removals are also both strongly negative, meaning both sides of the attention score contribute. Furthermore, retaining only the probe-aligned component is also harmful, indicating these directions are necessary but not sufficient for the routing computation.

This is consistent with our activation patching result: probe-aligned directions participate in layer 1 attention routing, but the full mechanism requires additional structure that the low-rank probes do not capture.

A.16. Probe accuracy vs. sequence position

Figure 11 shows our probe and task accuracy as a function of relative sequence position. The top-of-stack probe is near-perfect ($\approx 100\%$) throughout, dropping only to 97.1% at sequence end. The depth probe Pearson r hovers around 0.40–0.45 with no positional trend, consistent with weak linear decodability. Next-token accuracy rises from 25% to 77% near sequence end, reflecting increasing predictability as sequences approach termination.

A.17. Probe corruption tracking

Figure 12 shows probe accuracy on corrupted inputs by offset from the corruption point. The top-of-stack probe tracks the corrupted input faithfully (100% vs. corrupt stack, 0% vs. clean stack at offset 0), while the depth probe is flat and identical for both comparisons across all offsets, confirming that our corruption procedure correctly isolates the type-tracking mechanism.

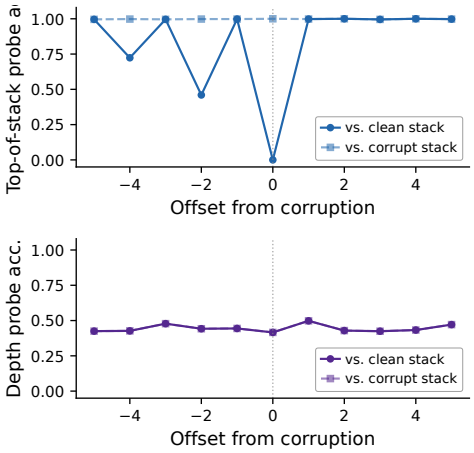

Figure 12. Probe tracking on corrupted inputs. Top: top-of-stack probe accuracy versus clean and corrupt ground truth, measured by offset from the corruption point. Bottom: depth probe accuracy is unchanged by the corruption, confirming that bracket-type corruption does not affect depth.

A.18. Multi-seed replication

We train three additional seeds (7, 11, 22) on the base Dyck-$(20, 10)$, $d = 64$ model using 150k steps, and two additional seeds (11, 22) on the Dyck-$(40, 18)$ harder variant. We find that seeds 7, 11, 22, and 42 consistently learn the same mechanism, with mean layer 1 stack-top attention mass ≈ 0.97, edge-masking yielding $\Delta \approx -0.938 \pm 0.003$, and activation patching recovery jumping from 0 to 1 at post-

layer-1-attention (logit-diff recovery 1.000 ± 0.000). We note that the 1.000 values are an architectural identity, since once our clean post-layer-1 attention state is restored at the query position, all remaining computation deterministically reproduces the clean output.

We also train four harder model variants with different language complexities — specifically with $(k, m) \in \{(20, 18), (30, 14), (40, 10), (40, 18)\}$ — all using independent random initializations. The consistency across these variants (Table 5) provides evidence that our findings are not specific to a single random seed.

We note that one additional seed converged to a qualitatively different circuit — reaching comparable long-distance accuracy with layer 0 attention specialized rather than layer 1 — which we leave as an open question about the uniqueness of the solution rather than a failure of our discovered mechanism.

A.19. Natural language pilot: subject-verb agreement

To test if our dissociation extends beyond formal languages, we train a 2-layer, 1-head transformer with the same architecture ($d = 64$) on a templated subject-verb agreement task. We evaluate the three main properties identified in the Dyck setting: decodability without causality (Q1), causal necessity for a single edge (Q2), and sublayer localization (Q3).

In our setup, each sentence follows a controlled template with a single unambiguous subject, zero or more prepositional phrase attractors with independently randomized numbers, and a dedicated `<verb>` query token at the end. This design eliminates the redundant syntactic cues present in unconstrained English, forcing our model to retrieve the subject number via attention rather than local token identity.

Across four seeds (42, 7, 11, 22), our model achieves 99.2–99.4% accuracy uniformly across attractor counts (0–3), confirming no shortcut exploitation. Furthermore, Q1, Q2, and Q3 all hold: the subject number probe accuracy remains ≥ 0.99 IID and OOD, while the attractor count probe R^2 collapses from positive IID to -12.3 OOD, and activation patching localizes the computation to layer 1 attention (logit-diff recovery 1.000 at post-layer-1 attention, 0.000 at post-embedding), replicating our Dyck localization result exactly.

For Q2, seeds 42, 7, and 11 show a catastrophic accuracy drop (≈ 0.50) when the single subject-to-verb attention edge at layer 1 is masked. Interestingly, however, seed 22 only routes subject information partially through layer 0 attention, so simply masking layer 1 is insufficient. Nevertheless, jointly masking layer 0 and layer 1 is catastrophic, confirming that our mechanism concentrates in a single attention layer. This behavior mirrors our outlier seed discussed in Appendix A.18, suggesting that solution multiplicity across layers is a general phenomenon rather than a Dyck-specific artifact.

In summary, all three findings replicate in natural language: hierarchical variables are decodable but not causally necessary (Q1), a single attention edge is causally necessary for task performance (Q2), and critical computation localizes to a single sublayer (Q3).