Transformers Learn Two Competing Sorting Algorithms, and Only One Length-Generalizes

Mechanisms, initialization, and the limits of the one-head result

Khashayar Gatmiry, Nathan W. Henry, Chloe H. Su, Emmanuel Abbe,
Samy Bengio, Melanie Weber, Jonathan A. Kelner, Peter L. Bartlett

arXiv:2610.26974v1 Research group discussion, 26 October 2026

Extra capacity creates a choice of learned algorithms

Question

If one attention layer can sort, what does training do when a second layer is available?

- The same two-layer architecture reaches near-zero training loss through two distinct attention circuits.
- In the paper's *one-head-per-layer* setting, the circuit that uses both layers extrapolates much better to longer lists.
- The paper traces this split to rare small gaps, cross-layer specialization, and a query-side hint.

Paper, Secs. 1, 3–4; PDF pp. 2–10.

Sorting is a sequence of successor queries

Data and target

Sample K distinct values from $\{1, \ldots, N\}$; append SEP; output the values in ascending order.

$$x_1, \ldots, x_K, \mathrm{SEP}, x_{s(1)}, \ldots, x_{s(K)}.$$

At output value $x_{s(i)}$, predict its successor $x_{s(i+1)}$.

Paper, Sec. 2; PDF p. 4.

Main protocol

Train at $K = 16$; test from $K = 2$ to $K = N$. Causal transformer, two layers, one head per layer, no positional encoding.

Online sampling; AdamW, learning rate 0.03, batch 4096, 60,000 steps.

Ablating attention 2 separates two solution classes

Single-stage

Remove the second *attention* layer: full-sequence sorting accuracy stays near 1. The first attention selects a wider set of nearby candidates; later MLPs can still choose the output.

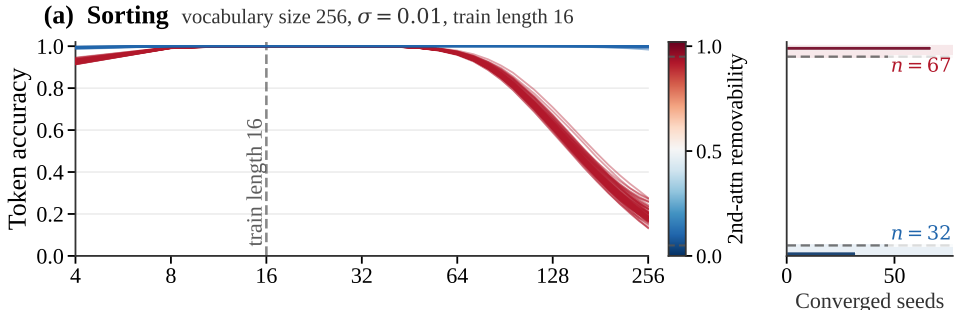
Two-stage

The same ablation collapses full-sequence accuracy toward 0. Both attention layers contribute; attention 2 is much more focused on the correct successor.

Keep the metrics apart: ablated *sequence* accuracy defines the class; unablated *token* accuracy measures length generalization. MLP2 remains in the ablation.

Paper, Sec. 3, Fig. 1; PDF pp. 2, 4–5.

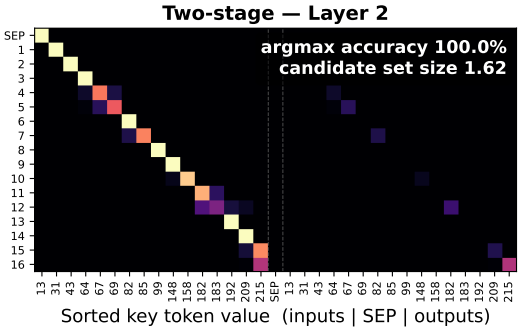
At long lengths, the two-stage one-head models win



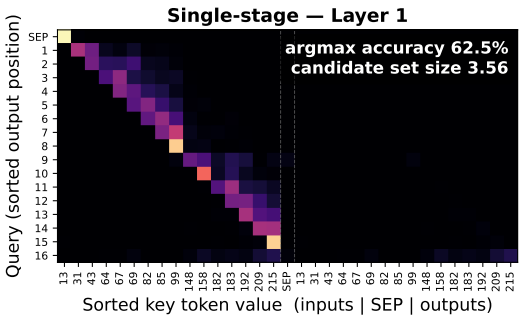
Training: $K = 16$, $N = 256$. At $K = 256$, blue two-stage curves remain near 1 token accuracy; red single-stage curves fall near 0.2.

Paper, Fig. 1a, PDF p. 2. Each curve is one seed; color encodes attention-2 removability.

The two circuits allocate attention differently



Two-stage: attention 2 sharply selects successors.



Single-stage: attention 1 spreads over nearby candidates.

Paper, Fig. 2, PDF p. 5. Cropped panels from one input; bright cells have higher attention weight. Attention-argmax accuracies are 100% and 62.5% here.

Two mechanisms may explain the sorting split

1. Gap specialization

Long lists contain many more adjacent values with gap $g = 1$. In two-stage models, the first-layer path often suffices for small gaps; attention 2 handles larger gaps.

2. Query-side hint

Attention 1 supplies contextual information to attention 2. The second stage can search within a smaller value window and keep a narrower candidate set.

These are mechanistic interpretations supported by ablations, interventions, score scans, and attention statistics.

Paper, Sec. 4; PDF pp. 5–10, Figs. 2–5.

Longer lists make unit gaps common

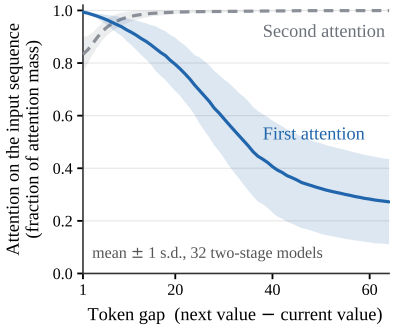
$$g_i = x_{s(i+1)} - x_{s(i)}, \qquad \mathbb{E}[g] = \frac{N+1}{K+1}.$$

$N = 256$	$K = 16$ (train)	$K = 128$	$K = 256$
Mean adjacent gap	15.12	1.99	1.00
$\Pr(g = 1) = K/N$	6.25%	50%	100%

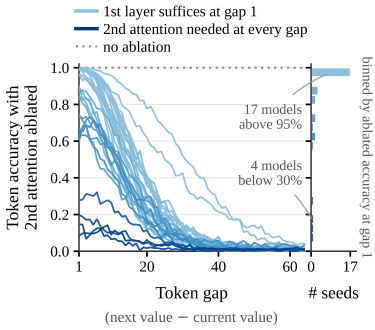
The probability row follows by counting uniform K-subsets; it is a teaching derivation from the paper's sampling scheme. A rare training case becomes the entire $K = N$ test case.

Paper, Sec. 4.1 and Appendix C; PDF pp. 6, 23.

The layers specialize across gap sizes



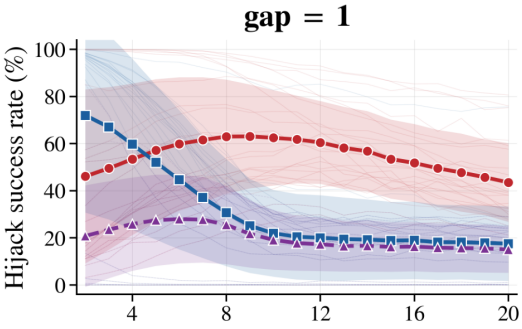
Attention 1 attends less to input tokens as g grows.



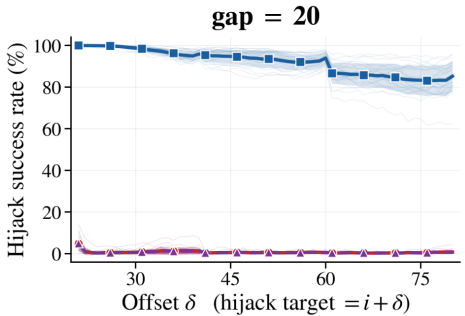
With attention 2 removed, many seeds still solve $g = 1$; accuracy collapses at larger gaps.

Paper, Fig. 3a–b, PDF p. 7. Shading in (a): mean ± 1 s.d. across 32 two-stage models.

Interventions show a handoff to attention 2



Gap 1: either path can steer the output.



Gap 20: attention 2 dominates.

Red: redirect attention 1's direct path. Blue: redirect attention 2. Purple: both individually succeed on the same input.

Paper, Fig. 4, PDF p. 8; $K = 32$, $N = 512$, 23 two-stage seeds. Panels cropped from the original figure.

Layer 1 changes the query used by layer 2

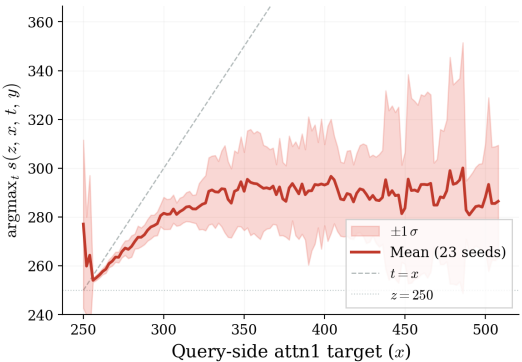
Stylized single-target intervention

$$s(z, x, t, y) = \mathbf{q}(z, x)^\top \mathbf{k}(t, y),$$
$$\mathbf{q}(z, x) = W_Q^{(2)} \, \mathrm{LN}\Big[\mathrm{MLP}_1\Big(\mathrm{LN}(e_z + V_x^{(1)})\Big)\Big].$$

z: current output value; t: candidate key; x and y: first-attention targets on the query and key sides. $V_x^{(1)}$ is the projected value vector.

Paper, Sec. 4.2, Eq. (1), PDF p. 8; Appendix E, PDF pp. 31–33.

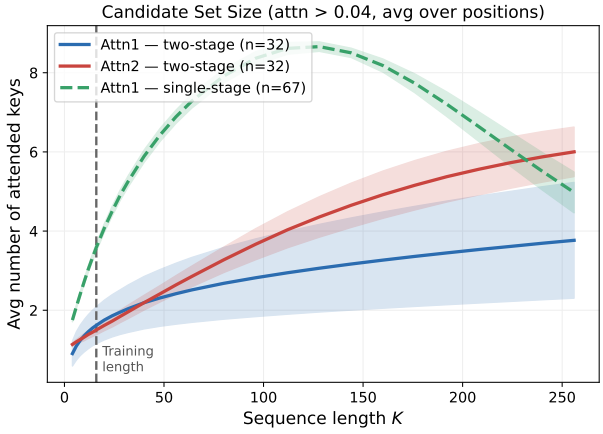
The hint creates a query-dependent search window



- The score is nearly invariant to key-side target y, but changes strongly with query-side target x.
- High scores concentrate roughly in $[z, x]$; the preferred key moves with x, then saturates near 300 for $z = 250$.
- Very small windows do not follow this pattern as cleanly.

Paper, Sec. 4.2, Fig. 5a, PDF p. 9; Appendix E, Fig. 20, PDF p. 32. Curve averages 23 seeds.

Two-stage attention keeps a smaller candidate set



Candidate = key with attention weight > 0.04. At $8\times$ training length, attention 1 averages 3.1 keys in two-stage models versus 8.7 in single-stage models.

Paper, Sec. 4.2, Fig. 5b, PDF pp. 9–10. The threshold count is a descriptive probe.

A small scale asymmetry steers algorithm choice

(σ_1, σ_2)	$\Delta = \sigma_2 - \sigma_1$	Two-stage seeds
$(0.012, 0.010)$	-0.002	$0/100$
$(0.010, 0.010)$	0	$26/100$
$(0.010, 0.011)$	$+0.001$	$82/100$
$(0.010, 0.013)$	$+0.003$	$100/100$

- Larger layer-2 initialization favors using attention 2; larger layer-1 initialization favors a single-stage solution.
- At $\Delta = +0.003$, some classified two-stage seeds generalize worse: solution class alone does not fix circuit quality.

Paper, Sec. 5.1, Fig. 6, PDF pp. 10–11; Appendix A, Fig. 8, PDF p. 20. Sorting: $K = 16$, $N = 128$.

Task difficulty and model capacity shift the choice

Larger vocabulary N

At fixed training length 16, near-all seeds are single-stage at $N = 64$; most are two-stage at $N = 512$.

Harder numerical discrimination appears to recruit the second layer.

Wider embedding E

At fixed $N = 256$, the two-stage share falls from all trained seeds at $E = 32$ to none at $E = 80$.

More representational capacity appears to make a one-attention solution easier to reach.

These comparisons concern the *probability of learning a solution class*, not the accuracy of one fixed trained model.

Paper, Sec. 5.2 and Appendix B, Figs. 9–10; PDF pp. 10, 21–22. The proposed optimization explanations are hypotheses.

Training length changes which gaps get learned

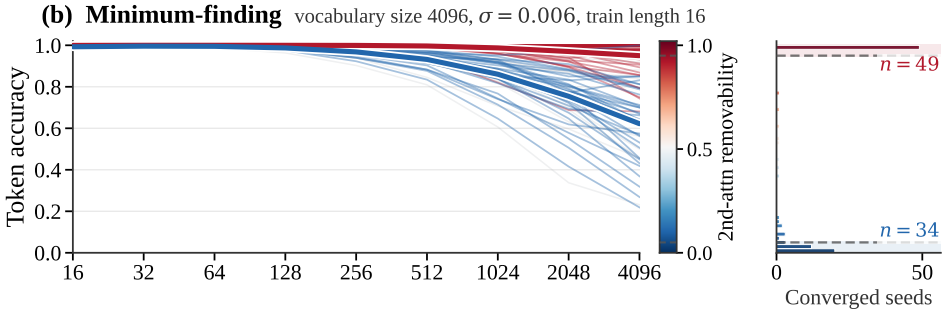
- At training lengths $K \geq 16$, two-stage seeds generalize essentially perfectly to the longest tested length; at shorter training lengths, none does so perfectly.
- At $K_{\text{train}} = 4$, first-layer attention to input values at gap 1 correlates with accuracy at $K = 256$: $r = +0.77$.
- A fully consecutive length-32 sequence has expected training frequency 10^{-39}; two-stage accuracy on its consecutive positions is 1.00, versus 0.89 for single-stage models.

Tradeoff

Longer training lists teach small gaps, while making very large gaps rarer and harder to cross.

Paper, Appendix C, Figs. 11–15; PDF pp. 23–27. Adversarial accuracy is scored on the specified consecutive positions.

Minimum-finding reverses the generalization ranking



Train at $K = 16$, $N = 4096$, sampling *with replacement*. Single-stage models (red) remain more accurate on long inputs than two-stage models (blue).

Paper, Sec. 6, Fig. 1b; PDF pp. 2, 11–13. Each curve is one seed.

The minimum task exposes different rare-input failures

Two-stage weakness

Many values clustered near the minimum become more common in longer lists; two-stage models degrade on these cases.

Single-stage weakness

Duplicates are harder for single-stage models. If the minimum has score a_1 and n copies of the runner-up have score a_2, then

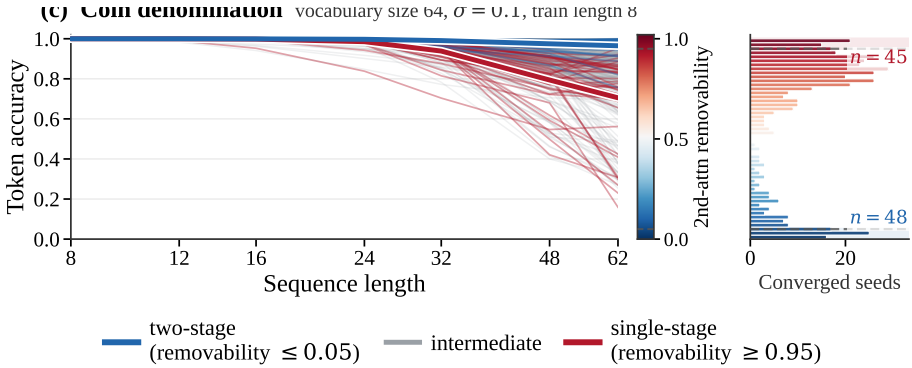
$$a_1 - a_2 > \log n$$

gives the minimum more total attention.

Which solution generalizes depends on which failure mode the length shift makes prevalent.

Paper, Sec. 6.2; PDF p. 13. The score-margin argument is an explanatory hypothesis.

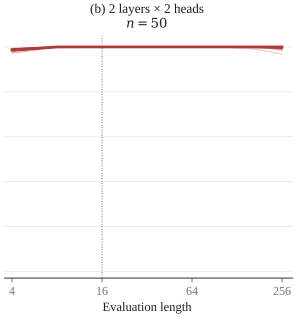
Coin denomination shows a softer split

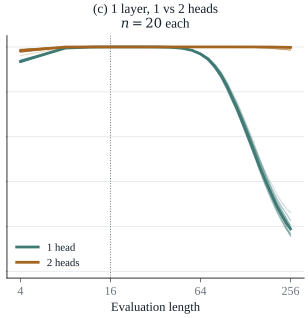


Greedy denomination is a predecessor-query task. Train with 8 distinct denominations from a vocabulary of 64; models relying more on attention 2 tend to generalize better, with more overlap and intermediate cases than sorting.

Paper, Sec. 6, Fig. 1c; PDF pp. 2, 11–12. The algorithm outputs successive remainders.

A second head rescues single-stage sorting





Two layers, two heads each: all shown models are single-stage, yet generalize.

One layer: two heads generalize; one head fails.

Token accuracy versus evaluation length; vertical line marks train length 16. The main sorting split is specific to one head per layer.

Paper, Appendix D, Fig. 19b–c, PDF p. 31. Panels cropped from the original figure; embedding dimension $E = 64$.

Takeaways: study the learned circuit and the shifted inputs

1. **Training success does not identify the algorithm.** Attention-2 ablation separates distinct functional solutions to the same sorting task.
2. **Length changes the input statistics.** Small gaps become common; gap-specialized layers and a query-dependent window help in the one-head sorting experiments.
3. **Complexity alone does not predict extrapolation.** Minimum-finding reverses the ranking; another attention head removes the sorting failure of single-stage models.

Ask which rare training cases become common at evaluation time, and which path handles them.

Paper, Secs. 3–6 and Appendix D; PDF pp. 4–13, 30–31.

Backup: MLP1 carries the second-stage QK signal

- Replace attention 2's input $e + a_1 + m_1$ with m_1 only.
- In 13 standard two-stage checkpoints, mean attention-distribution ℓ_1 distance is < 0.03; full-sequence accuracy drops by at most 0.05%.
- Two $N = 1024$ checkpoints are outliers: the mechanism is common, not required by architecture.
- Query-side target x strongly changes QK scores; key-side target y has little effect.

Paper, Appendix E, Figs. 20–22; PDF pp. 31–33. The 13 detailed checkpoints exclude two outliers from 15 two-stage checkpoints.

Backup: depth four preserves multiplicity, not the same mechanism

- Four-layer, one-head models can require one, two, three, or four leading attention layers to retain sorting accuracy.
- Lowering initialization of layers 2–4 to 0.005 steers 14 of 50 models to a one-layer solution; equal 0.01 scales yield none of those.
- At this depth, models requiring at least two layers length-generalize, but the paper does not find the same clean gap division or iterative candidate-set reduction.

Paper, Appendix D, Figs. 16–17; PDF pp. 27–29. Four-layer experiments use $K = 16$, $N = 128$, $E = 64$.